

RECOGNITION OF TREE METRICS*

HANS-JÜRGEN BANDELT†

Abstract. A tree metric on a set of n points (objects) is realized by a weighted undirected tree whose tips are labelled by the objects. It is well known that tree metrics are characterized by the so-called 4-point condition. In this note it is shown that a metric d is a tree metric if and only if the Farris transform of d qualifies as an ultrametric. Based on this, an $O(n^2 \log n)$ procedure is established for recognizing tree metrics on an n -set.

Key words. ultrametric, tree metric

AMS(MOS) subject classification. 05C05

Ultrametrics and tree metrics on a given set Ω of objects are certain real-valued matrices whose rows and columns correspond to the objects. Namely, a matrix $d = (d_{ij})$ is an *ultrametric* on Ω if and only if all diagonal elements d_{ii} are zero and

$$d_{ij} \leq \max \{d_{ik}, d_{jk}\} \quad \text{for all } i, j, k \in \Omega.$$

If this ultrametric inequality is only required for distinct objects i, j, k , then we have to assume symmetry and nonnegativity of d explicitly. Similarly, a matrix $d = (d_{ij})$ with zero diagonal is a *tree metric* if and only if

$$d_{ij} + d_{kl} \leq \max \{d_{ik} + d_{jl}, d_{il} + d_{jk}\} \quad \text{for all } i, j, k, l \in \Omega.$$

This inequality is usually referred to as the 4-point condition. It entails the triangle inequality (for $k = l$) and symmetry of d (for $i = k$ and $j = l$).

There is an extensive literature on ultrametrics; see Gordon [13] for a recent survey. Concerning tree metrics the reader can consult [2], [4], [5], [6], [8], [10], [19], [21], [22], [25], [26], [27]; see also Abdi [1] for an overview. Ultrametrics are usually represented by dendrograms, while tree metrics d are represented by unrooted trees with branch lengths so that each dissimilarity d_{ij} becomes the length of the i, j -connecting path. This representation is not immediate from the definition of a tree metric, but it follows easily from the recursive description below. Every ultrametric is, of course, a tree metric. Interestingly, a tree metric can be characterized in terms of an associated ultrametric. To this end, one selects any object $r \in \Omega$, serving as an artificial root, and then measures twice the distance from r to the i, j -connecting path; subtracting this from a sufficiently large constant gives a new dissimilarity δ_{ij} .

Transforming tree metrics to ultrametrics. Let $d = (d_{ij})$ be a metric on a set Ω . Choose any $r \in \Omega$ and any constant $c \geq 2 \cdot \max_{i,j} d_{ij}$. Define a matrix $\delta^{(r)} = (\delta_{ij}^{(r)})$ with zero diagonal by

$$\delta_{ij}^{(r)} = c + d_{ij} - d_{ir} - d_{jr} \quad \text{for } i \neq j \text{ in } \Omega.$$

Then $\delta^{(r)}$ is an ultrametric if and only if d is a tree metric.

To prove this, first observe that $\delta = \delta^{(r)}$ is symmetric and nonnegative. Further, δ is a metric because for $k \neq i, j$ we have

$$\begin{aligned} \delta_{ik} + \delta_{jk} &= d_{ik} + d_{jk} + 2c - d_{ir} - d_{jr} - 2d_{kr} \\ &\geq d_{ik} + d_{jk} + c - d_{ir} - d_{jr} \\ &\geq \delta_{ij}. \end{aligned}$$

* Received by the editors January 4, 1988; accepted for publication March 9, 1989.

† Vakgroep Kwantitatieve Economie, Faculteit der Economische Wetenschappen, Rijksuniversiteit Limburg, Postbus 616, N-6200 MD Maastricht, the Netherlands.

If i, j, k, l are all distinct, then

$$\delta_{ij} + \delta_{kl} = d_{ij} + d_{kl} + (2c - d_{ir} - d_{jr} - d_{kr} - d_{lr}).$$

Hence δ is a tree metric if and only if d is. For distinct objects i, j, k in $\Omega - \{r\}$, put

$$b = c - d_{ir} - d_{jr} - d_{kr}.$$

So, if d is a tree metric, then

$$\begin{aligned} \delta_{ij} &= d_{ij} + d_{kr} + b \\ &\leq \max \{ d_{ik} + d_{jr} + b, d_{jk} + d_{ir} + b \} \\ &= \max \{ \delta_{ik}, \delta_{jk} \}. \end{aligned}$$

This ultrametric inequality also holds in the case where r is among i, j, k since

$$\delta_{ij} \leq c = \delta_{ir} \quad \text{for all } i \neq r \quad \text{and } j \text{ in } \Omega.$$

This concludes the proof of the asserted equivalence.

The transformation $d \mapsto \delta^{(r)}$ from tree matrices to ultrametrics was already suggested by Farris [11] and Farris, Kluge, and Eckardt [12] (see also [7, p. 119] or [18, p. 302]) and was rediscovered by Klotz and Blanken [15] and Brossier [4]. It was also mentioned in Hartigan's book [14] that d is a tree metric if and only if, for every r , the derived matrix $\delta^{(r)}$ is an ultrametric. As it actually suffices (for the "if" part) that only one $\delta^{(r)}$ qualifies as an ultrametric, we can reduce the complexity of recognizing tree metrics to $O(n^3)$, where n is the number of objects in Ω . This was also observed by Simões-Pereira and Zamfirescu [24].

Another immediate consequence of the above transformation is the additive representation of an arbitrary tree metric d by an ultrametric and a bush metric (minus a constant c); cf. Carroll and Pruzansky [7] (cf. [4]). A metric $\beta = (\beta_{ij})$ is a *bush metric* on Ω if and only if

$$\beta_{ij} + \beta_{kl} = \beta_{ik} + \beta_{jl} \quad \text{for distinct } i, j, k, l \in \Omega.$$

Then, given any $r \in \Omega$, we obtain from the definition of $\delta^{(r)}$ that

$$c + d_{ij} = \beta_{ij} + \delta_{ij}^{(r)}, \quad \text{where } \beta_{ij} = d_{ir} + d_{jr}, \text{ for all } i, j \in \Omega.$$

From this we infer that the rooted tree representing $\delta^{(r)}$ has the same topology as the unrooted tree representing d .

A recursive description of ultrametrics is easily established. Using the above Farris transformation, this lifts to a recursive characterization of tree metrics.

Recursive description of ultrametrics and tree metrics. Let $d = (d_{ij})$ be a dissimilarity matrix on a set Ω with $n \geq 3$ objects.

Let i, j be distinct objects such that d_{ij} is minimal. Then d is an ultrametric if and only if

$$d_{ik} = d_{jk} \quad \text{for all } k \in \Omega - \{i, j\},$$

and the restriction of d to $\Omega - \{i\}$ is an ultrametric.

Pick any object $r \in \Omega$. Let i, j be distinct objects different from r such that $d_{ir} + d_{jr} - d_{ij}$ is maximal. Then d is a tree metric if and only if

$$d_{ik} + d_{jr} = d_{ir} + d_{jk} \quad \text{for all } k \in \Omega - \{i, j, r\},$$

d restricts to a metric on $\{i, j, r\}$, and the restriction of d to $\Omega - \{i\}$ is a tree metric.

The first assertion is quite evident. As to the second one, first observe that $d_{ir} + d_{jr} - d_{ij}$ is maximal if and only if $\delta_{ij}^{(r)}$ is minimal. Furthermore,

$$d_{ik} + d_{jr} = d_{ir} + d_{jk} \Leftrightarrow \delta_{ij}^{(r)} = \delta_{jk}^{(r)}.$$

So it remains to check the triangle inequality for the matrix d satisfying the above requirements. Suppose by way of contradiction that d violates the triangle inequality for some triple of objects. Then i must belong to this triple because the restriction of d to $\Omega - \{i\}$ is a metric. First assume that the triple consists of i, j and some object $k \neq i, j$. Then k is also different from r since d restricts to a metric on $\{i, j, r\}$. If d_{ij} exceeds $d_{ik} + d_{jk}$, then, as d restricts to a metric on $\Omega - \{i\}$, we obtain

$$\begin{aligned} d_{ir} + d_{kr} - d_{ik} &> d_{ir} + d_{kr} + d_{jk} - d_{ij} \\ &\geq d_{ir} + d_{jr} - d_{ij}, \end{aligned}$$

contrary to the choice of the pair i, j . But if d_{ik} exceeds $d_{ij} + d_{jk}$, then, as d restricts to a metric on $\{i, j, r\}$, we get

$$\begin{aligned} d_{ik} + d_{jr} &> d_{ij} + d_{jk} + d_{jr} \\ &\geq d_{ir} + d_{jk}, \end{aligned}$$

thus conflicting with the required equality in the recursion. We arrive at a similar contradiction if d_{jk} is larger than $d_{ij} + d_{ik}$. Therefore, the triangle inequality can only be violated for a triple consisting of i and some $k \neq l$ in $\Omega - \{i, j\}$. Say, $k \neq r$. Assume that $l = r$. If d_{kr} exceeds $d_{ik} + d_{ir}$, then (because d restricts to a metric on $\{i, j, r\}$)

$$\begin{aligned} d_{ir} + d_{kr} - d_{ik} &> 2 \cdot d_{ir} \\ &\geq d_{ir} + d_{jr} - d_{ij}, \end{aligned}$$

contrary to the choice of the pair i, j . If d_{ir} exceeds $d_{ik} + d_{kr}$, then (because d restricts to a metric on $\Omega - \{i\}$)

$$\begin{aligned} d_{ir} + d_{jk} &> d_{ik} + d_{kr} + d_{jk} \\ &\geq d_{ik} + d_{jr}, \end{aligned}$$

which is impossible. Similarly, d_{ik} cannot be larger than $d_{ir} + d_{kr}$. Therefore, l must be different from r . Next assume that d_{kl} exceeds $d_{ik} + d_{il}$. It follows that

$$\begin{aligned} d_{jr} + d_{kl} &> d_{jr} + d_{ik} + d_{il} \\ &= d_{jk} + d_{ir} + d_{il} \\ &\geq d_{jk} + d_{lr} \end{aligned}$$

since the triangle inequality is already established for all triples including both i and r . Then, by interchanging the role of k and l , we also get

$$d_{jr} + d_{kl} > d_{jl} + d_{kr}.$$

This, however, is impossible because d restricts to a tree metric on $\Omega - \{i\}$. Finally assume that d_{ik} exceeds $d_{il} + d_{kl}$. Then

$$\begin{aligned} d_{ik} + d_{jr} &> d_{il} + d_{kl} + d_{jr} \\ &= d_{ir} + d_{kl} + d_{jl} \\ &\geq d_{ir} + d_{jk}, \end{aligned}$$

which again is absurd. A similar contradiction is obtained when d_{il} is larger than $d_{ik} + d_{kl}$. Hence d actually satisfies the triangle inequality. This concludes the proof of the recursive characterization.

Recognition of ultrametrics and tree metrics. Let Ω be a set of $n \geq 3$ objects, and let $d = (d_{ij})$ be a real-valued matrix on Ω . First we check in $O(n^2)$ time whether d is a dissimilarity matrix.

Arrange the dissimilarities d_{ij} in linear order. This is accomplished in $O(n^2 \log n)$ time. Starting with the minimal member d_{ij} of this list, test the requirement of the recursive scheme for ultrametrics, delete the corresponding object i , and proceed in the accession list with the first member d_{jk} with $j, k \neq i$. The process terminates at latest when there are only two objects left. The whole recursion requires $O(n^2)$ time. Then d is an ultrametric if and only if d passes each test.

In order to decide whether d is a tree metric, choose any object $r \in \Omega$. Then arrange the derived similarities $d_{ir} + d_{jr} - d_{ij}$ in linear order. Starting with the maximal member of the resulting list, test successively the requirements of the recursive scheme for tree metrics.

Either algorithm for recognizing ultrametrics, respectively, tree metrics, executes in $O(n^2 \log n)$ time.

The asymmetric analogues of tree metrics can also be recognized in $O(n^2 \log n)$ time; such matrices are represented by weighted bidirectional trees (see [8], [19], [23]). Indeed, by [23, Thms. 5.3 and 6.5], a nonnegative asymmetric matrix (a_{ij}) is of this kind if and only if its symmetrization $(a_{ij} + a_{ji})$ is a tree metric, and for any point r chosen in advance, the equality

$$a_{ij} + a_{jr} + a_{ri} = a_{ir} + a_{rj} + a_{ji}$$

holds for all distinct $i, j \neq r$.

The above recognition procedure is mainly of theoretical interest, insofar as real data hardly ever matches tree metrics perfectly, let alone ultrametrics. Nevertheless, we can modify the recognition procedures in order to account for the approximation of arbitrary dissimilarity matrices d to ultrametrics or tree metrics. In either case we use the accession list provided by the corresponding recognition procedure. Say, at some step the pair i, j is processed. Then instead of deleting the object i , we merge the current clusters containing i and j into a new cluster. Thereafter, that pair k, l from the list is selected that is next to i, j but not included in any already established cluster. Still, each step of the recursion executes in linear time, so that the algorithm requires a total of $O(n^2 \log n)$ time. In the case where the accession list is given by the linear order of the original dissimilarities d_{ij} , this algorithm is, of course, nothing but the single linkage technique. This method applied to the derived dissimilarities $\delta_{ij}^{(r)}$ then yields a rooted tree, which fits the given matrix d after disregarding the orientation and estimating the branch lengths in one way or the other.

In contrast to the method of Farris [11] and Klotz and Blanken [15], the method of Klotz et al. [16] requires the addition of a hypothetical root r to the set Ω of objects, along with estimated dissimilarities between r and the original objects, but otherwise operates in the same way. For a related procedure, see Li [17]. Comparisons of these methods with other techniques can be found in [3], [20].

The above approximation procedures have the drawback of being fairly sensitive to moderate or large perturbations of just a few dissimilarities involving the distinguished object r . It is therefore advisable to use several objects in turn as temporary root r (as suggested in [15]), and then to merge the resulting unrooted trees into their strict consensus



FIG. 1

tree. If the number of trials is bounded by a (small) constant, then the total amount of time is still of order $O(n^2 \log n)$ since computing strict consensus trees is of low complexity; see Day [9].

In order to approximate a metric $d = (d_{ij})$ on Ω to a tree metric, Hartigan [14] (ascribing this to Farris) suggested that one should add up all $\delta_{ij}^{(r)}$ for $r \in \Omega$, "which will then fit the ultrametric model approximately" (see also [12, p. 186]). This, however, is very questionable since a tree metric is not necessarily recovered this way. Figure 1 displays a tree metric on $\Omega = \{i, j, k, l\}$. The matrix $(\sum_r \delta_{ij}^{(r)})$ supports the rooted tree shown in Fig. 1, when one of the single, average, or complete linkage techniques is used.

Note added in proof. In a recent paper (presented at the SIAM Conference on Discrete Mathematics in San Francisco, 1988) by Batagelj, Pisanski, and Simões-Pereira, an algorithm is described to test tree realizability in $O(n^2)$ time, and it is noted that similar results are due to Boesch (1968) and Culberson and Rudnicki. Such algorithms, however, do not extend to reliable procedures for approximating metrics by tree metrics.

REFERENCES

- [1] H. ABDI, *Additive tree representations*, Paper presented at the meeting "Trees and Hierarchical Structures" held at the University of Bielefeld, Bielefeld, Federal Republic of Germany, 1987.
- [2] H.-J. BANDELT AND A. DRESS, *Reconstructing the shape of a tree from observed dissimilarity data*, *Advances Applied Math.*, 7 (1986), pp. 309–343.
- [3] R. L. BLANKEN, L. C. KLOTZ, AND A. G. HINNEBUSH, *Computer comparison of new and existing criteria for constructing evolutionary trees from sequence data*, *J. Molecular Evol.*, 19 (1982), pp. 9–19.
- [4] G. BROSSIER, *Approximation des dissimilarités par des arbres additifs*, *Math. Sci. Humaines*, 91 (1985), pp. 5–21.
- [5] P. BUNEMAN, *The recovery of trees from measures of dissimilarity*, in *Mathematics in the archaeological and historical sciences*, F. R. Hodson, D. G. Kendall, and P. Tautu, eds., Edinburgh University Press, Edinburgh, Scotland, 1971, pp. 387–395.
- [6] ———, *A note on the metric properties of trees*, *J. Combin. Theory Ser. B*, 17 (1974), pp. 48–50.
- [7] J. D. CARROLL AND S. PRUZANSKY, *Discrete and hybrid scaling models*, in *Similarity and choice*, E. D. Lantermann and H. Feger, eds., Hans Huber Publishing, Bern, 1980, pp. 108–139.
- [8] J. P. CUNNINGHAM, *Free trees and bidirectional trees as representations of psychological distance*, *J. Math. Psych.*, 17 (1978), pp. 165–188.
- [9] W. H. E. DAY, *Optimal algorithms for comparing trees with labeled leaves*, *J. Classification*, 2 (1985), pp. 7–28.
- [10] A. J. DOBSON, *Unrooted trees for numerical taxonomy*, *J. Appl. Probab.*, 11 (1974), pp. 32–42.
- [11] J. S. FARRIS, *On the phenetic approach to vertebrate classification*, in *Major patterns in vertebrate evolution*, M. K. Hecht, P. C. Goody, and B. M. Hecht, eds., Plenum Press, New York, 1977, pp. 823–850.
- [12] J. S. FARRIS, A. G. KLUGE, AND M. J. ECKARDT, *A numerical approach to phylogenetic systematics*, *Systematic Zool.*, 19 (1970), pp. 172–189.
- [13] A. D. GORDON, *A review of hierarchical classification*, *J. Roy. Statist. Soc. A*, 150 (1987), pp. 119–137.
- [14] J. A. HARTIGAN, *Clustering algorithms*, John Wiley, New York, 1975.

- [15] L. C. KLOTZ AND R. L. BLANKEN, *A practical method for calculating evolutionary trees from sequence data*, J. Theoret. Biol., 91 (1981), pp. 261–272.
- [16] L. C. KLOTZ, N. KOMAR, R. L. BLANKEN, AND R. M. MITCHELL, *Calculation of evolutionary trees from sequence data*, Proc. Nat. Acad. Sci. U.S.A., 76 (1979), pp. 4516–4520.
- [17] W.-H. LI, *Simple method for constructing phylogenetic trees from distance matrices*, Proc. Natl. Acad. Sci. U.S.A., 78 (1981), pp. 1085–1089.
- [18] M. NEI, *Molecular evolutionary genetics*, Columbia University Press, New York, 1987.
- [19] A. N. PATRINOS AND S. L. HAKIMI, *The distance matrix of a graph and its tree realization*, Quart. Appl. Math., 30 (1972), pp. 255–269.
- [20] N. SAITOU AND M. NEI, *The number of nucleotides required to determine the branching order of three species with special reference to the human-chimpanzee-gorilla divergence*, J. Molecular Evol., to appear.
- [21] S. SATTAH AND A. TVERSKY, *Additive similarity trees*, Psychometrika, 42 (1977), pp. 319–345.
- [22] J. M. S. SIMÕES-PEREIRA, *A note on the tree realizability of a distance matrix*, J. Combin. Theory, 6 (1969), pp. 303–310.
- [23] ———, *A note on optimal and suboptimal digraph realizations of quasidistance matrices*, SIAM J. Algebraic Discrete Methods, 5 (1984), pp. 117–132.
- [24] J. M. S. SIMÕES-PEREIRA AND C. M. ZAMFIRESCU, *Submatrices of non-tree-realizable distance matrices*, Linear Algebra Appl., 44 (1982), pp. 1–17.
- [25] M. S. WATERMAN, T. F. SMITH, M. SINGH, AND W. A. BEYER, *Additive evolutionary trees*, J. Theoret. Biol., 64 (1977), pp. 199–213.
- [26] K. A. ZARECKIJ, *Postroenie dereva po naboru rasstojanij meždu vicjačimi veršinami*, Uspehi Mat. Nauk, 20 (1965), pp. 90–92.
- [27] A. A. ZYKOV, *Teorija konečnyh grafov*, Izdat., Nauka, 1969.

SORTING, MINIMAL FEEDBACK SETS, AND HAMILTON PATHS IN TOURNAMENTS*

AMOTZ BAR-NOY† AND JOSEPH NAOR‡

Abstract. A general method is presented for translating sorting by comparisons algorithms to algorithms that compute a Hamilton path in a tournament. The translation is based on the relation between minimal feedback sets and Hamilton paths in tournaments. It is proven that there is a one to one correspondence between the set of minimal feedback sets and the set of Hamilton paths. In the comparison model, all the tradeoffs for sorting between the number of processors and the number of rounds hold as well for computing Hamilton paths. For the CRCW model, with $O(n)$ processors, we show the following: (i) Two paths in a tournament can be merged in $O(\log \log n)$ time (Valiant's algorithm [*SIAM J. Comput.*, 4 (1975), pp. 348–355], (ii) a Hamilton path can be computed in $O(\log n)$ time (Cole's algorithm). This improves a previous algorithm for computing a Hamilton path whose running time was $O(\log^2 n)$ using $O(n^2)$ processors.

Key words. parallel computation, comparison model, sorting, tournaments, Hamilton paths, feedback sets

AMS(MOS) subject classifications. 68E05, 68Q10, 68R10

1. Introduction. A tournament $T = (V, D)$ ($|V| = n$) is a directed graph in which every pair of vertices is joined by a directed edge. It can be viewed as a complete graph whose edges are given an orientation. A Hamilton path in a graph is a simple path that contains all the vertices, and each vertex appears exactly once. A well-known theorem states that there is a Hamilton path in every tournament [Re], [Be].

In this paper we investigate the complexity of computing a Hamilton path in a tournament and other problems related to it. Our methodology relies on the intimate relationship that exists between Hamilton paths and minimal feedback sets in tournaments and their connection to sorting algorithms. Sorting by comparisons may be viewed as computing a Hamilton path in a *transitive (acyclic)* tournament. The purpose of this paper is to show the opposite direction, namely how sorting algorithms can be generalized to compute a Hamilton path in an arbitrary tournament.

Sorting by comparisons is a well-investigated problem, perhaps the most well investigated in computer science. We show how to exploit the wealth of results on it to design parallel and sequential algorithms for arbitrary tournaments.

Parallel algorithms to compute a Hamilton path in a tournament have appeared in [Na], [So], but were designed by ad hoc principles. The key idea in computing the Hamilton path in both papers is the following: in every tournament there exists a vertex (*separator*) whose indegree and outdegree are bounded from below by $|V|/4$; this gives a recursive formulation of the problem with only a logarithmic number of steps. The difficulty with this approach is that the best bound known for finding a separator requires $O(n^2)$ processors. Using adversary arguments, Ramachandran (private communication) showed that a lower bound on the number of edges, whose orientation must be known in order to find a separator, is $\Omega(n^2)$.

In analogy to sorting, we define two additional problems on tournaments:

- Merging two paths such that the relative order of the vertices in the original paths remains after the merge.

* Received by the editors December 19, 1988; accepted for publication (in revised form) May 1, 1989.

†‡ Computer Science Department, Stanford University, Stanford, California 94305.

† This work was supported in part by a Weizmann fellowship, by Office of Naval Research contract N00014-88-K-0166, and by a grant from the Center for Integrated Systems, Stanford University, Stanford, California 94305.

‡ This work was supported by Office of Naval Research contract N00014-88-K-0166 and by a grant from the Center for Integrated Systems, Stanford University, Stanford, California 94305.

- k -selection, e.g., finding the k th vertex in the Hamilton path in a transitive tournament, is generalized to finding a vertex that is the k th in some Hamilton path in an arbitrary tournament.

We now present a summary of our method. A *minimal edge feedback set* F in a directed graph, $G = (V, D)$ is a set of edges such that $G' = (V, D - F)$ is acyclic, and F is minimal with respect to containment. We prove that there is a one-to-one correspondence between the set of minimal feedback sets and the set of Hamilton paths in an arbitrary tournament, and show how a minimal feedback set can generate a Hamilton path and vice versa. We also show that an edge in a transitive tournament, whose orientation is known by implication, cannot appear in the Hamilton path. Assume now that the input to a sorting algorithm is not a transitive tournament but rather any arbitrary tournament. If a minimal feedback is computed for each round of comparisons and its orientation flipped, namely, the sorting algorithm is “cheated,” then the Hamilton path computed by the sorting algorithm will also be a Hamilton path for the original input. Intuitively, this happens because the edges on which we “cheated” are actually implications. These notions are formalized and proved, and they result in a general method for translating any sorting (or sorting related) algorithm to an algorithm that computes a Hamilton path.

There are two known proofs for the existence of Hamilton paths in tournaments. One proof is Redei’s proof [Re], and the other is the aforementioned separator theorem. In view of our results, the first proof corresponds to *insertion-sort*, whereas the second to *quick-sort*. In fact, our results imply that the known sequential sorting algorithms also compute a Hamilton path in an arbitrary tournament.

The equivalence between sorting and computing a Hamilton path holds both in the comparison model and in PRAMS. Valiant’s comparison model [Va] can be easily generalized to arbitrary tournaments. In this model we prove that the complexity of computing a Hamilton path in an arbitrary tournament and in a transitive one is the same. It follows that all the lower bounds, upper bounds and processor-time tradeoffs for sorting, apply also when computing a Hamilton path in an arbitrary tournament. These bounds and tradeoffs have been proved in a series of papers, [AA1], [AA2], [AAV], [Al2], [AV], [BHe], [BHo], [P]. Hell and Rosenfeld [HR] have also considered the sequential complexity of algorithms in the comparison model for computing generalized paths in tournaments.

Implementing our translation method in the PRAM model is more complex. The difficulty is that it requires the computation of a minimal feedback edge set in a directed graph. It is not known whether an NC algorithm exists for this problem. We consider two PRAM sorting algorithms, and give for these special cases a nontrivial procedure that computes a minimal feedback set in constant time. The algorithms are Cole’s *merge-sort* [Co] and Valiant’s *merging* algorithm [Va], and our results are in the CRCW (common) model. Hence, a Hamilton path can be computed in $O(\log n)$ time, and two paths of length n can be merged in $O(\log \log n)$ time; both algorithms use $O(n)$ processors.

These two algorithms achieve an almost optimal speed-up with respect to their sequential complexity. Notice that in our case the number of processors is linear in the number of vertices and not edges; this result is interesting by itself. Kruskal [Kr] showed how to implement Valiant’s merging algorithm optimally with $O(n/\log \log n)$ processors and $O(\log \log n)$ time; it seems that his methods carry over to our merging algorithm as well.

Our results also imply an $O(n \log n)$ sequential algorithm for computing a Hamilton path in an arbitrary tournament. As we already have mentioned, *merge-sort* will also output a Hamilton path for an arbitrary tournament. This bound also follows from Redei’s proof [Re], but only when appropriate data structures are used.

An interesting open problem that stems from our work is whether there exists an efficient sorting network that computes a Hamilton path in an arbitrary tournament. The sorting network implied by the insertion-sort method also computes a Hamilton path when the input is an arbitrary tournament (but the parallel running time is linear). On the other hand, Batcher's sorting network [Ba] fails to compute a Hamilton path when applied to an arbitrary tournament. Note that a sorting network for arbitrary tournaments which has the same complexity bounds as [AKS] would also imply better bounds for EREW algorithms.

As for the k -selection problem, it is easy to show a lower bound of $\Omega(\log n / \log \log n)$ time (see Theorem 3.2) for algorithms in any PRAM model even in the case of transitive tournaments. Hence, the best strategy (up to a factor of $O(\log \log n)$) would be to first compute a Hamilton path and then solve the k -selection problem. On the other hand, in the comparison model, the result of [AKSS] implies an $O(\log \log n)$ upper bound for k -selection.

2. Terminology and preliminaries. Let $T = (V, E)$ denote a tournament, that is, a complete graph whose edges are oriented. Let the cardinality of the vertex set be denoted by n . If an edge is oriented from v to w , then we say that v is *smaller* than w (w is *greater* than v) and denote it either by $v < w$, or by $v \rightarrow w$. A *path* $v_0, v_1, \dots, v_k$ is a sequence of vertices where $v_i < v_{i+1}$ and $v_i \neq v_j$ for $0 \leq i \neq j \leq k$. A *cycle* is a path where $v_0 = v_k$. If vertex v precedes vertex w in a path P , then $v < w$ with respect to the path P or v is *below* w (w is *above* v). The first and last vertices in the path are called *bottom* and *top*, respectively.

A tournament is *transitive* if and only if it is acyclic. An *implication* in a transitive tournament is an edge whose orientation is implied by the orientation of other edges in the tournament, namely to avoid cycles.

Let P and Q be two paths in an arbitrary tournament and let R be the result of merging P and Q . Then this *merge* has the following property: if $v < w$ with respect to $P(Q)$, then also $v < w$ with respect to R .

An *edge feedback set* F in a directed graph, $G = (V, D)$ is a set of edges such that $G' = (V, D - F)$ is acyclic. Computing such a set of minimum cardinality is NP-complete [GJ], whereas computing a minimal such set with respect to containment can be easily done in polynomial time by a greedy algorithm. An easy property of minimal feedback sets is that the graph resulting from inserting any edge of F in G' is cyclic.

The graph $G = (V, W, Q)$ is called a *complete path directed bipartite graph* (abbreviated CPB) if Q contains all the edges between V and W , and the graph induced by V (W) is a directed Hamilton path.

3. Hamilton paths, minimal feedback sets and the comparison model. In this section we show how Hamilton paths and minimal feedback sets are related to each other in tournaments. We first extend the parallel comparison model to tournaments. This model was first introduced by Valiant [Va] for the purpose of analyzing sorting algorithms; only comparisons are taken into account in it, whereas internal processor computation and communication are not charged for. The structure of an algorithm in this model is the following: in each round a set of element pairs are compared until the output is known. The aim of an algorithm is to minimize both the number of rounds and the total number of comparisons.

We extend this model for tournaments in the natural way. The answer to a comparison is the orientation of an edge in the tournament. Hence, in the beginning we have a tournament whose edge orientation is unknown, and at each round we ask for the orientation of a set of edges. The algorithm proceeds until the induced graph of the known edges contains the solution. Let $f(k, n)$ denote the minimum number of com-

parisons needed to compute a function f in k rounds in a tournament of order n . As we already have mentioned, sorting can be viewed as computing a Hamilton path in a transitive tournament.

The next easy lemma is used by our main theorem, which follows immediately.

LEMMA 3.1. *Let $e = u \rightarrow v$ be an edge in a transitive tournament whose direction is known by implication. Then e cannot appear on the Hamilton path.*

Proof. If the orientation of e is known by implication then there exists an element w such that $u \rightarrow w$ and $w \rightarrow v$. Hence, w must be on the Hamilton path between u and v and therefore e cannot appear on the Hamilton path. $\square$

THEOREM 3.1. *Let $\mathcal{A}$ be an algorithm that computes a Hamilton path in a transitive tournament with complexity $H(k, n)$. Then there exists an algorithm $\mathcal{B}$ that computes a Hamilton path in an arbitrary tournament T with the same complexity $H(k, n)$.*

Proof. Let $Q_1, \dots, Q_k$ be disjoint sets of directed edges and let $F = F_1 \cup F_2 \cup \dots \cup F_k$ be a set with the following properties:

1. $F_i \subseteq Q_i$.
 2. F_1 is a minimal feedback set in the graph induced by Q_1 .
 3. Inductively, F_i is a minimal feedback set in the graph induced by $(Q_1 - F_1) \cup (Q_2 - F_2) \cup \dots \cup (Q_{i-1} - F_{i-1}) \cup Q_i$.
- It is easy to verify that for every i , $F^i = F_1 \cup \dots \cup F_i$ is a minimal feedback in the graph induced by $Q_1 \cup \dots \cup Q_i$.

Denote by $\neg F_i$ the set of edges of F_i such that the orientation of each edge in F_i is flipped and let $Q'_i = Q_i - F_i \cup \neg F_i$. Define $\mathcal{A}(R_1, \dots, R_i)$ to be the set of comparisons in round $i + 1$ of algorithm $\mathcal{A}$ under the assumption that the comparisons in the first i rounds were $R_1, \dots, R_i$. With these notations, the set of comparisons of Algorithm $\mathcal{B}$, $Q_1, \dots, Q_k$, will be derived from Algorithm $\mathcal{A}$ in the following way:

1. $Q_1 \leftarrow \mathcal{A}(\emptyset)$.
2. $Q_{i+1} \leftarrow \mathcal{A}(Q'_1, \dots, Q'_i)$.

First we show that Algorithm $\mathcal{B}$ is well defined by proving that $H_i = Q'_1 \cup \dots \cup Q'_{i-1}$ is a legal input to round i of Algorithm $\mathcal{A}$. It is easy to verify that H_i is a legal input if it is acyclic. Assume to the contrary that there is a cycle C in it. The edges of H_i are of two types: edges of the minimal feedback set F^i that were flipped, or unflipped edges. By the definition of a minimal feedback set, for every edge $e = (u \rightarrow v) \in F^i$, there is a path p_e of unflipped edges from v to u . Now, exchange every flipped edge $e \in C$ by p_e and get as a result a cycle of edges that were not flipped. If that cycle is not simple, it contains a simple cycle as a subgraph, hence contradicting F^i being a minimal feedback set.

Now that Algorithm $\mathcal{B}$ is well defined, assume to the contrary that its output p is not a Hamilton path in T . The path p is not a Hamilton path only if it contains edges whose orientation was flipped, namely edges that belong to F , the minimal feedback set. This cannot happen as the edges of F are implications in the graph induced by H_k , and therefore cannot appear on the Hamilton path by Lemma 3.1. $\square$

COROLLARY 3.1. *Let $\mathcal{A}$ be an algorithm that merges two paths in a transitive tournament with complexity $M(k, n)$. Then there exists an algorithm $\mathcal{B}$ that merges two paths in a nontransitive tournament with the same complexity.*

Another problem that was extensively studied in transitive tournaments is the k -selection problem: find an element larger than $k - 1$ elements and smaller than $n - k$ elements. This problem can be generalized to nontransitive tournaments in two ways. The first one is to find an element in the tournament (if one exists) whose indegree is exactly equal to k . As previously mentioned, it can be shown that the minimum number of comparisons needed to determine such an element is $\Omega(n^2)$, and therefore the running time cannot be polylogarithmic with a linear number of processors.

A more relaxed definition is to find an element v in the tournament such that v is the k th element in some Hamilton path. We show that the complexity of this problem is equivalent to the complexity of k -selection in the transitive case.

COROLLARY 3.2. *Let $\mathcal{A}$ be an algorithm that solves the k -selection problem in transitive tournaments with complexity $S(k, n)$. Then there exists an algorithm $\mathcal{B}$ that solves the k -selection problem in arbitrary tournaments with the same complexity.*

Proof. Algorithm $\mathcal{A}$ is translated into $\mathcal{B}$ in the same way as in Theorem 3.1. The corollary follows from the observation that any k -selection algorithm can be viewed as a step in a sorting algorithm. $\square$

It follows from Corollary 3.2 that the k -selection algorithm of [AKSS] in the comparison model, whose time complexity is $O(\log \log n)$ with a linear number of processors, can be applied to arbitrary tournaments as well. On the other hand, in the PRAM model, we have the following theorem that implies that the best strategy for k -selection, (up to a factor of $O(\log \log n)$), would be to first compute a Hamilton path.

THEOREM 3.2. *There is a lower bound of $\Omega(\log n / \log \log n)$ on the complexity of the k -selection problem for transitive tournaments in the PRAM model.*

Proof. This follows easily from the lower bound of [Bea] on computing the exclusive OR of a bit vector, as it also implies a lower bound of $\Omega(\log n / \log \log n)$ on sorting. Assume there was a k -selection algorithm whose time complexity was better than $O(\log n / \log \log n)$. Invoking this algorithm n times simultaneously for $k = 1, 2, \dots, n$ would imply a better bound for sorting. Hence, a contradiction. $\square$

Another consequence of Theorem 3.1 is the following relation between Hamilton paths and minimal feedback sets.

THEOREM 3.3. *Let $\mathcal{P}$ be the set of all Hamilton paths in a tournament T and let $\mathcal{F}$ be the set of all minimal feedback sets in a tournament T . There is a one to one correspondence between $\mathcal{P}$ and $\mathcal{F}$.*

Proof. We first show how a minimal feedback set F can be computed from a given Hamilton path p . For every pair of vertices v, w such that v precedes w in the path, add the edge (v, w) to F if it is oriented from w to v . Obviously, F is a minimal feedback set and two different paths cannot generate the same minimal feedback set. We now prove the other direction. Let F be a given minimal feedback set in a tournament T and let T' be the transitive tournament in which the orientation of the edges in F were flipped. By Theorem 3.1, a Hamilton path in T' is also a Hamilton path in T . Let F_1 and F_2 be two minimal feedback sets and let $v \rightarrow w$ be an edge in F_1 and not in F_2 . F_1 and F_2 cannot generate the same path, as T'_1 will contain $v \leftarrow w$ and T'_2 will contain $v \rightarrow w$. $\square$

It follows from Theorem 3.3 that all the results on the cardinality of $\mathcal{P}$ [Mo], [Al1] apply to $\mathcal{F}$ as well. A criterion to decide whether there is a path in a tournament that starts at a given vertex v and ends at some other given vertex w also follows from Theorem 3.3. Let in_v (out_v) be the set of incoming (outgoing) edges into v (out of v).

COROLLARY 3.3. *A necessary and sufficient condition for the existence of a Hamilton path from v to w is the existence of a minimal feedback set containing both in_v and out_w .*

Proof. Assume that such a minimal feedback set exists. After its edges are flipped, v becomes a source and w a sink and hence, according to Theorem 3.1, there is a Hamilton path from v to w . In the other direction, if a Hamilton path from v to w exists, then in_v and out_w belong to its corresponding minimal feedback set. (This is the same construction as in Theorem 3.2). $\square$

4. The CRCW model. In this section we exhibit algorithms for sorting and merging in an arbitrary tournament that are based on Theorem 3.1. The complexity of these algorithms matches the complexity of the corresponding algorithms in transitive tour-

naments. Our algorithms are in the CRCW common model where in a simultaneous write processors are required to write the same value.

In the first subsection we present Algorithm MFS that in $O(1)$ time finds a minimal feedback set in a CPB graph $G = (V, W, E)$. In the other two subsections, we show how to translate certain algorithms for sorting and merging in transitive tournaments to algorithms in arbitrary tournaments by calling MFS as a subroutine. It turns out that computing a minimal feedback set in our special cases reduces to computing it in a CPB graph. We shall prove that in each case, the feedback set computed by repeated calls to Algorithm MFS is indeed the minimal feedback set required by Theorem 3.1. The algorithms chosen are (they both use $O(n)$ processors):

Merging: Valiant's merging algorithm [Va] with $O(\log \log n)$ time complexity.

Sorting: Cole's Merge-Sort [Co] with $O(\log n)$ time complexity.

4.1. Finding a minimal feedback set in CPB graphs. Algorithm MFS invokes the two procedures MAXINDEX and MAXPREFIX. We now present them and prove that each can be implemented in $O(1)$ time.

Procedure MAXINDEX (A):

input: A binary sequence $A = a_1, \dots, a_l$; l processors.

output: The maximum index k , $1 \leq k \leq l$, such that $a_k = 1$.

LEMMA 4.1. MAXINDEX can be implemented in the CRCW model in $O(1)$ time.

Proof. It is well known that computing the OR (respectively, AND) of a binary sequence of length l with l processors takes $O(1)$ time in the CRCW model.

First we show how to compute MAXINDEX in $O(1)$ time with l^2 processors. Each element of A such that $a_i = 1$ computes $b_i = \max_{i < j \leq l} a(j)$ in $O(1)$ time. (This is possible as each a_i has l processors available.) If $b_i = 1$, then there exists an index $j > i$ such that $a_j = 1$, and therefore i cannot be the desired answer. In this case we set $a_i = 0$ and clearly, only one value remains equal to 1. The index of this value is the answer.

We now show how to compute MAXINDEX in $O(1)$ time in the CRCW model with only l processors. (For clarity, assume that $\sqrt{l}$ is an integer.)

1. Divide the l elements of the sequence into $\sqrt{l}$ blocks where each contains $\sqrt{l}$ consecutive items.

2. For $1 \leq i \leq \sqrt{l}$, let c_i denote the OR of the i th block.

3. Let k be the output of Procedure MAXINDEX when the input is the vector $c_1, \dots, c_{\sqrt{l}}$. This takes $O(1)$ time as there are l processors and the vector is of length $\sqrt{l}$.

4. Let k' be the output of Procedure MAXINDEX when the input is the k th block. This takes $O(1)$ time as there are l processors and the block is of length $\sqrt{l}$.

5. The output is k' . $\square$

Procedure MAXPREFIX (A, B):

input: A sequence $A = a_1, \dots, a_l$ of integers; l^2 processors.

output: A sequence $B = b_1, \dots, b_l$ of integers such that for every i , $1 \leq i \leq l$, $b_i = \max_{1 \leq k \leq i} \{a_k\}$.

LEMMA 4.2. MAXPREFIX can be implemented in the CRCW model in $O(1)$ time.

Proof. First construct in $O(1)$ time a matrix $M = \{m_{i,j}\}_{1 \leq i,j \leq l}$ where $m_{i,j} = 1$ if and only if $a_i > a_j$, or $i = j$. Each row i calls a variation of procedure MAXINDEX and computes in $O(1)$ time: the minimum index j in the subrow $m_{i,i+1}, \dots, m_{i,l}$ such that $m_{i,j} = 0$ and $m_{i,i+1} = \dots = m_{i,j-1} = 1$. For every k , $1 \leq k < i$ or $j \leq k \leq l$ set $m_{i,k} = 0$. If now $m_{i,j} = 1$, then $a_i > a_j$ and for each k , $i < k < j$, it is also known that $a_i > a_k$.

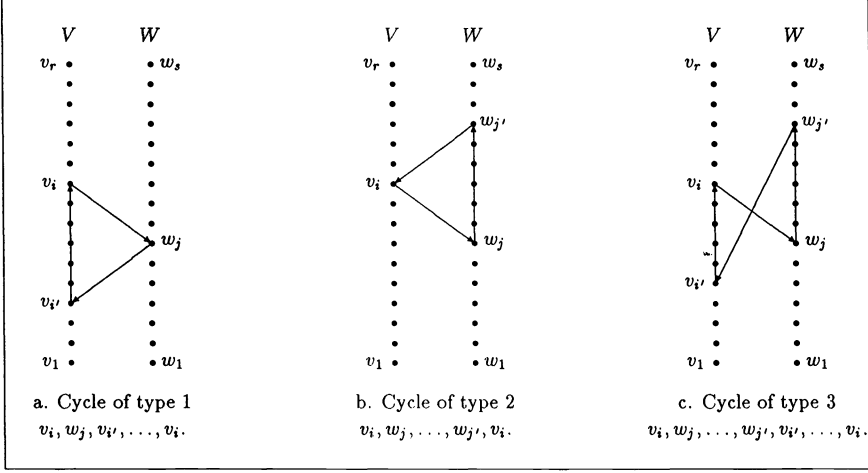


FIG. 1. The three types of cycles.

Therefore, it now remains for each a_j to find the minimal index i such that $m_{i,j} = 1$. Again, this can be done in time $O(1)$ by calling a variation of procedure MAXINDEX for each column of the matrix M . $\square$

We are now ready to present Algorithm MFS.

ALGORITHM MFS:

Input: A complete path bipartite graph $G = (V, W, Q)$; $V = v_1, \dots, v_r$; $W = w_1, \dots, w_s$; $O(r^2 + rs)$ processors.

Output: A minimal feedback set F in G .

For all i , $1 \leq i \leq r$:

1. Define $a(i)$ as follows:
 - (a) If $v_i > w_s$, then $a(i) = s$;
 - (b) else, if $v_i < w_1$, then $a(i) = 0$;
 - (c) otherwise, let $a(i)$ be the maximum index j such that $w_j < v_i < w_{j+1}$ ($1 \leq j \leq s-1$).
2. Let $b(i)$ be the maximum value among $\{a(1), \dots, a(i)\}$.
3. For all j , $1 \leq j \leq b(i)$: if $v_i < w_j$ then add the edge $v_i \rightarrow w_j$ to F .

LEMMA 4.3. The set F computed by Algorithm MFS is a minimal feedback set in Q .

Proof. F is a minimal feedback set if $Q' = Q - F$ is acyclic and if adding an edge of F to Q' generates a cycle.

Assume to the contrary that there is a cycle in Q' and suppose that the cycle is one of the following three types:

- (1) $v_i, w_j, v_{i'}, \dots, v_i$ (Fig. 1a);
- (2) $v_i, w_j, \dots, w_{j'}, v_i$ (Fig. 1b);
- (3) $v_i, w_j, \dots, w_{j'}, v_{i'}, \dots, v_i$ (Fig. 1c).

We now show that in the three above cases $b(i) \geq j$ and this leads us to a contradiction, because in Step 3 of Algorithm MFS, edge $v_i \rightarrow w_j \in F$.

1. The edge $w_j \rightarrow v_{i'}$ is in Q ; therefore the maximality of $a(i')$ implies that $a(i') \geq j$ and consequently, $b(i) \geq b(i') \geq a(i') \geq j$.
2. The edge $w_{j'} \rightarrow v_i$ is in Q ; therefore $a(i) \geq j'$ which implies that $b(i) \geq j' > j$.
3. The edge $w_{j'} \rightarrow v_{i'}$ is in Q ; therefore $b(i) \geq b(i') \geq a(i') \geq j' > j$.

Now let c be an arbitrary cycle and let v_i be its highest vertex in V . The successor of v_i in c belongs to W , and denote it by w_j . We show that there are three possible cases, and each implies the existence of one of the above three types of cycles.

If w_j is the highest vertex of c in W , then its successor in c is $v_{i'} \in V$. It follows that $i' < i$ (from v_i being the highest) and hence $v_i, w_j, v_{i'}, \dots, v_i$ is a cycle in Q' of type (1). Otherwise, let $w_{j'}$ be the highest vertex of c in W for some $j' > j$. Denote the successor of $w_{j'}$ in c by $v_{i'} \in V$ for $i' \leq i$. There are two cases. If $i' = i$ then in Q' , $v_i, w_j, \dots, w_{j'}, v_i$ is a cycle of type (2). Otherwise, $i' < i$ and $v_i, w_j, \dots, w_{j'}, v_{i'}, \dots, v_i$ is a cycle in Q' of type (3).

To complete the proof, we have to show that if an edge $v_i \rightarrow w_j \in F$ is added to Q' , then a cycle is generated. This is implied by the following two facts:

FACT 1. *The edge $w_{a(i)} \rightarrow v_i$ is always an edge in Q' as long as $a(i) > 0$.*

FACT 2. *If $v_i \rightarrow w_j \in F$ then $j \leq b(i)$.*

If $j < a(i)$, then $v_i, w_j, \dots, w_{a(i)}, v_i$ is a cycle (type (2)). Otherwise, by the above two facts, $a(i) < j \leq b(i)$. If $j = b(i)$, then there exists $i' < i$ such that $a(i') = b(i)$. By Fact 1, $w_{b(i)} \rightarrow v_{i'} \in Q'$ and $v_i, w_j, v_{i'}, \dots, v_i$ is a cycle (type (1)). Otherwise, $j < b(i)$ and again, there exists $i' < i$ such that $a(i') = b(i) > j$ and by Fact 1 $w_{b(i)} \rightarrow v_{i'} \in Q'$. It follows that $v_i, w_j, \dots, w_{b(i)}, v_{i'}, \dots, v_i$ is a cycle (type (3)). $\square$

LEMMA 4.4. *Algorithm MFS can be computed in time $O(1)$.*

Proof. It is easy to see how step 1(c) can be computed with the help of procedure MAXINDEX with rs processors. Hence, it takes time $O(1)$ by Lemma 4.2. Step 2 calls Procedure MAXPREFIX with r^2 processors and can also be computed in time $O(1)$ by Lemma 4.3. Obviously, the rest of the algorithm can be computed in time $O(1)$. $\square$

4.2. Merging. In this subsection we show how Valiant's [Va] merging algorithm can be modified to a CRCW algorithm that merges two paths in a tournament. For a detailed description and implementation of Valiant's algorithm, the reader is referred to [Kr]. Here we elaborate on its description as well to simplify the proof of Theorem 4.1 and also for the sake of completeness. The algorithm uses a linear number of processors (linear in the length of both paths) and its time complexity remains $O(\log \log n)$ (n is the length of X , the shorter path). Borodin and Hopcroft [BHo] proved a lower bound of $\Omega(\log \log n)$ on the complexity of merging two paths and hence, our results are tight. Hereafter, denote by $X = x_1, \dots, x_n$ and $Y = y_1, \dots, y_m$ the two paths to be merged. For clarity, we omit ceiling and floor notation and assume that the sorted elements are distinct.

Valiant's algorithm employs a *divide-and-conquer* method: in $O(1)$ time, merging two paths is divided into merging many pairs of subpaths from X and Y , where in each pair, the length of the subpath belonging to X is $\sqrt{n}$. Then, in $O(1)$ time, the paths produced by merging the pairs are concatenated to output the desired merge. The basic step of the recursion is when n is a constant and then, the merge is completed by performing all possible comparisons. The time complexity $O(\log \log n)$ is the solution of the recursive equation:

- $T(\alpha) = O(1); \alpha = O(1);$
- $T(n) = T(\sqrt{n}) + O(1).$

We now explain how the divide-and-conquer is achieved. Each of the two paths, X and Y , is divided into subpaths whose lengths are a square root of their original length. The top of every subpath is called its *leader*, and denote by LX and LY the set of leaders in X and Y , respectively. We may sometimes refer to LX and LY also as paths.

The first step of comparisons in the algorithm is between all the leaders of LX with all the leaders of LY . (This can be done in one step as $\sqrt{nm} < n + m$.) At this point, for each leader $y \in LY$, there are two successive leaders $x' < x \in LX$, such that $y > x'$ and

$y < x$. This pair of leaders is uniquely defined in the case of a transitive tournament and can be computed in time $O(1)$. The exact place in X where y is eventually inserted, will be called the *insertion point* of y . Now, for two consecutive leaders in LY , $y' < y$, there is a subpath $LX(y)$ in LX and a subpath $Y(y)$ in Y , such that each vertex in these subpaths is smaller than y and greater than y' (Fig. 2(a)).

The second step of comparisons in the algorithm is performed between all the vertices in $LX(y)$ and $Y(y)$ for each leader $y \in LY$. After this step, the insertion point in Y of each leader $x \in LX$ is known, namely there are two successive vertices in Y , $y' < y$, such that $x > y'$ and $x < y$. Again, this pair of vertices is uniquely defined in the case of a transitive tournament and can be computed in time $O(1)$. Now for two consecutive leaders in LX , $x' < x$, there is a subpath $X(x)$ in X and a subpath $Y(x)$ in Y , such that each vertex in these subpaths is smaller than x and greater than x' (Fig. 2(b)).

Now the subpaths can be merged simultaneously. For every leader $x \in LX$, the algorithm merges recursively the paths $X(x)$ and $Y(x)$ into the path $Z(x)$. After the recursion is completed, $x(x')$ can be inserted above (below) $Z(x)$.

Thus, by merging $\sqrt{n}$ pairs of subpaths and then concatenating them through the leaders of X , the desired merge is achieved.

The comparison steps that precede the recursive calls in Valiant's algorithm incur two problems in an arbitrary tournament. A leader $y \in LY$ ($x \in LX$) may have several insertion points in $LX(Y)$; furthermore, if $y' < y$ ($x' < x$) then the insertion point of y' (x') is not necessarily smaller than that of y (x).

To overcome these problems, we compute a minimal feedback set in the graph induced by the comparisons, as suggested by Theorem 3.1. Whenever a set of vertices is compared, a minimal feedback set is computed and the orientation of its edges is flipped. In all cases, the minimal feedback set is either the output of Algorithm MFS or the union of outputs of different calls to Algorithm MFS. There are four cases summarized below:

- 1) **The base case of the recursion.** Here, n is a constant, and the minimal feedback set is the output of Algorithm MFS computed on the whole graph.
- 2) **The first round of comparisons before the recursive calls.** Here, the leaders of LX and LY are compared and again, the minimal feedback set is the output of Algorithm MFS.

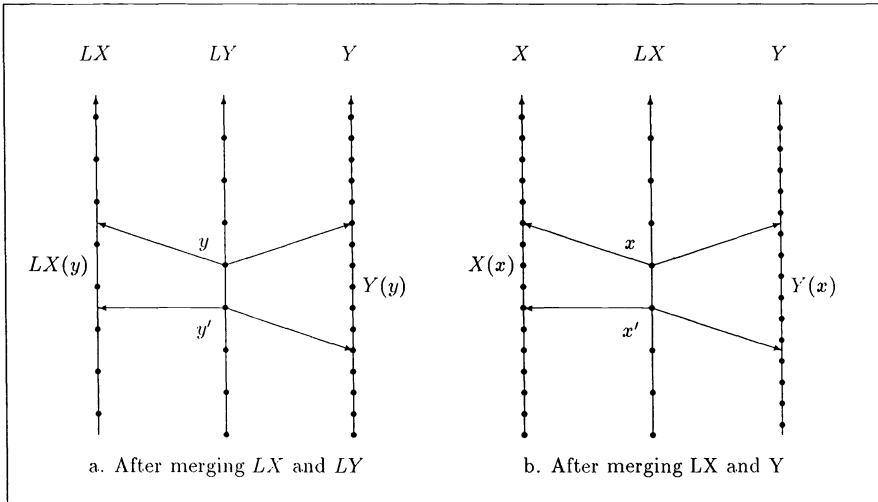


FIG. 2. Proof of Theorem 4.1.

3) The second round of comparisons before the recursive calls. The graph induced by these comparisons is a union of CPB graphs, and Algorithm MFS is invoked for each CPB graph. The minimal feedback set is the union of the minimal feedback sets computed for each graph.

4) The rounds during the recursive calls. The structure of the comparison graph is the same as in case (3) and the minimal feedback set is the union of the minimal feedback sets computed for each CPB graph.

THEOREM 4.1. *The modification of Valiant's merge algorithm according to Theorem 3.1 merges two paths in an arbitrary tournament.*

Proof. We prove it by induction on k , the depth of the recursion. When $k = 1$ (case 1), namely, the length of one of the merged paths is a constant, the correctness follows as Algorithm MFS computes a minimal feedback set (Lemma 4.3). Assume that the theorem is true when the recursion depth is less than k .

In the first step of comparisons (case 2), the leaders of X and Y are compared. It follows again from Lemma 4.3 that a minimal feedback set is indeed computed. Hence, the picture depicted by Fig. 2(a) is valid. Now assume that the second round of comparisons before the recursive calls took place (case 3) and assume to the contrary that the union of minimal feedback sets computed for edge disjoint subgraphs is not a feedback set. To disprove this last assumption, it is enough to show for every cycle that all of its edges belong to the same subgraph. We prove the following claim.

CLAIM. *For each leader $y \in LY$, if a vertex from $LX(y)$ (or $Y(y)$) takes part in a cycle, then that cycle is completely contained in $LX(y) \cup Y(y)$.*

Proof of the claim. Construct the following graph H : for each leader $y \in LY$ associate a vertex $f(y)$ in H which is the contraction of $LX(y)$ and $Y(y)$, e.g., a vertex in H can be viewed as a set of "old" vertices. Let a and b be two vertices in H ; the edge $a \rightarrow b$ exists if, before the contraction, there was an edge oriented from a vertex in the set a to a vertex in the set b .

Notice that the leader y is the only vertex in $f(y)$ that was compared to vertices in $LX(y')$ for any leader $y' < y$. The way the segments in LX were chosen implies that y is greater than all the vertices in $LX(y')$ and clearly y is greater than all the vertices in $Y(y')$. Hence, H is well defined and isomorphic to a path. The correctness of the claim follows immediately. $\square$

In the rounds during the recursion (case 4), similar arguments to the above hold. The analogous claim is that for each leader $x \in LX$, if a vertex from $X(x)$ (or $Y(x)$) takes part in a cycle, then that cycle is completely contained in $LX(y) \cup Y(y)$. Therefore, it is enough to compute a minimal feedback set in each CPB graph. $\square$

THEOREM 4.2. *The time complexity of the modified algorithm is $O(\log \log n)$ when $O(n)$ processors are available.*

Proof. Define $T(n, m)$ to be the time complexity of the modified algorithm. By Lemma 4.4, Algorithm MFS can be implemented in time $O(1)$ if the number of available processors is $O(r^2 + rs)$, where r and s denote the lengths of the input paths. We need to show that whenever Algorithm MFS is invoked there is a sufficient number of processors available.

If n is a constant, the claim follows from the fact that $O(n + m) = O(nm)$. When LX and LY are compared, the claim follows from the inequality $\sqrt{nm} < n + m$.

If $\sum \alpha_i = \sqrt{n}$, then $\sum \alpha_i^2 \leq n$. Therefore, there are enough processors for the second step of comparisons that precedes the recursive call. Each subpath $Y(y)$ from Y that is merged with α leaders from LX receives $O(\alpha^2 + \alpha\sqrt{m})$ processors.

For the simultaneous recursive calls there are enough processors, as each submerge of two paths of length n' and m' gets $O(n' + m')$ processors.

Clearly, the rest of the algorithm can be implemented in constant time the same as the transitive case. Hence the recursive equation is:

$$T(n, m) = T(\sqrt{n}, m') + O(1).$$

The solution of the above equality is $O(\log \log n)$. $\square$

4.3. Sorting. In this subsection we show how a Hamilton path in an arbitrary tournament can be computed in the CRCW model in $O(\log n)$ time using $O(n)$ processors. We rely on Cole's merge-sort [Co] algorithm and compute a minimal feedback set in $O(1)$ time for each round of comparisons. We shall give only an outline of Cole's algorithm and elaborate on the steps where comparisons are performed. We refer the reader to the description of Cole's algorithm in [GR], as our outline depends on it and uses its terminology.

In the previous subsection, the merging algorithm implies a sorting algorithm whose complexity is $O(\log n \log \log n)$. To improve the complexity to $O(\log n)$, Cole showed how to pipeline the merging steps. Let T be the complete binary tree that describes the standard *merge-sort* algorithm; a typical node v in T merges two lists X and Y when both X and Y are sorted. The novelty of Cole's algorithm is that node v starts processing the lists X and Y before they are sorted. Namely, at the i th step, two lists X_i and Y_i are merged, where X_i is a sorted sample of X , and Y_i is a sorted sample of Y . This merge can be computed in constant time if the results of the $i - 1$ st step are known.

For the sake of simplicity, assume that n is a power of 2. Let T_v be the subtree rooted at v and $list_v$ be the list of elements stored initially at the leaves of T_v . Let val_v be the current list associated with node v of the tree. The sequence val_v will always be a sorted subsequence of $list_v$ and will double its size in each round. We say that a node is *complete* if and only if $val_v = list_v$. Let us now describe what happens in a typical internal node v of the tree during the course of the algorithm (Fig. 3).

In the i th step, node v receives from its left child and right child sequences X_i and Y_i , respectively. It merges the two sequences to a sorted sequence val_v with the help of the old val_v . If v is complete, then it sends every fourth element of val_v to its parent. During the first step after v becomes complete, every second element is sent up, whereas

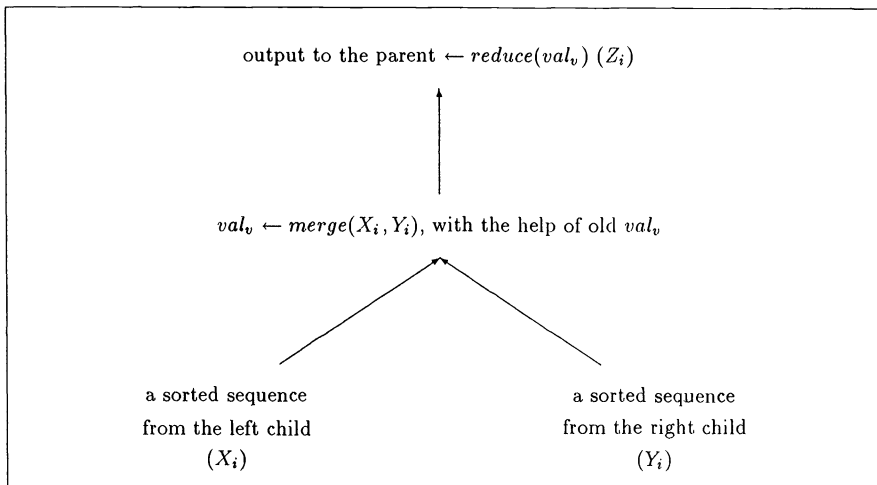


FIG. 3. The local action in an internal node v in the tree.

in the step after that, every element is sent up. The sequence Z_i sent by v to its parent is called $reduce(val_v)$.

The algorithm begins when all the leaves send their value to their parents. A node terminates two steps after it became complete, and the algorithm ends when the root becomes complete. It follows from the way $reduce(val_v)$ is defined, that two rounds after v became complete, its parent becomes complete. Therefore, the algorithm takes $O(\log n)$ rounds of internal computation in the nodes of the tree. In order to achieve the desired complexity, Cole showed how to implement each such round in constant time.

We now need some notations. The *rank* of an element x in a sequence X , $rank(x, X)$, is the number of elements in X preceding x . The *cross rank* from X to Y , denoted by $R[X, Y]$, is the function for which $R[X, Y](x) = rank(x, Y)$ for each $x \in X$. A sorted sequence X is a *good sampler* of the sorted sequence Y if and only if, between any two adjacent elements of $\{-\infty\} \cup X \cup \{\infty\}$ there are at most three elements of Y . Note that it is always true that $reduce(X)$ is a good sampler of X .

The motivation behind these notations is as follows. The cross rank $R[X, Y]$ ($R[Y, X]$) enables us to merge the sorted sequences X and Y in time $O(1)$. In this sense, the merge of two sequences and their cross ranks are equivalent. Our description relies heavily on this fact. It is also easy to merge in constant time two sorted sequences X and Y with the help of a sequence Z which is a good sampler of both (Fig. 4).

The basic property of the algorithm that entails its correctness is well demonstrated by the following invariant preserved at each step.

Main invariant. If each X_i is a good sampler of X_{i+1} , and each Y_i is a good sampler of Y_{i+1} then each Z_i is a good sampler of Z_{i+1} .

In terms of cross ranks, the merge of X_{i+1} and Y_{i+1} is performed in three steps:

1. The four cross ranks: $R[X_i, Y_{i+1}]$, $R[Y_i, X_{i+1}]$, $R[X_{i+1}, Y_i]$ and $R[Y_{i+1}, X_i]$ are computed with the help of $R[X_i, X_{i+1}]$ and $R[Y_i, Y_{i+1}]$.
2. The two cross ranks: $R[X_{i+1}, Y_{i+1}]$ and $R[Y_{i+1}, X_{i+1}]$ are computed with the help of the cross ranks of the previous step. (As stated before, this is equivalent to merging X_{i+1} and Y_{i+1}).
3. The cross rank $R[Z_i, Z_{i+1}] = R[reduce(X_i \cup Y_i), reduce(X_{i+1} \cup Y_{i+1})]$ is computed with the help of the previous calculated cross ranks.

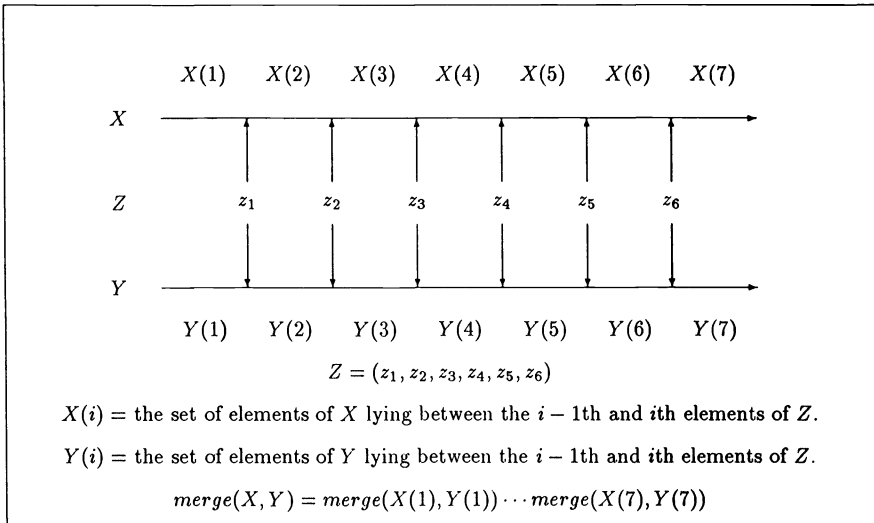


FIG. 4. Merging X and Y with the help of a good sampler Z .

The three steps can be implemented in $O(1)$ time with $O(|X_{i+1}| + |Y_{i+1}|)$ processors. The reason is that the good sampling property insures us that in each CPB graph, at least one of the paths is of length at most three.

We now explain how to modify the above algorithm according to Theorem 3.1 so that it computes a Hamilton path in an arbitrary tournament. Whenever a set of vertices is compared, a minimal feedback set is computed and the orientation of its edges is flipped. Each stage of comparisons can be decomposed to edge disjoint CPB graphs, and in each one a minimal feedback set is computed by invoking Algorithm MFS. The union of the minimal feedback sets computed in each CPB graph will be a minimal feedback set in the graph induced by the set of comparisons.

THEOREM 4.3. *The modification of Cole's sort algorithm according to Theorem 3.1 finds a Hamilton path in an arbitrary tournament.*

Proof. For arguments similar to those in the proof of the merging algorithm, it follows that the union of the outputs of the calls for algorithm MFS in each node of the tree T is a minimal feedback set in the graph induced by the comparisons performed at that node.

The theorem follows from the next claim proved by induction on $\langle k, i \rangle$, where k is the height of the tree and i is the current step.

- In the i th step, in a tree of height k , the algorithm computes a minimal feedback set.
- The sequence sent by v , the root of this tree, in the i th step is consistent with the previous sequences sent by v . Namely, if val_v contains the relation $x < y$ for $x, y \in list_v$, then in all previous steps, val_v never contained the relation $y > x$.

The induction holds clearly for $k = 1$, namely, the leaves. Now let T' be a tree of height k , let v be its root, and let i be the current step. We prove the first part of the inductive claim by showing that there cannot be a cycle after the edges of the minimal feedback set in T' , computed in the i th step, are flipped. Let X and Y be the sequences that v received from its left and right children, respectively. It follows from the induction assumption on the height of the tree that any cycle contained completely in X or in Y was handled by the children of v . On the other hand, a cycle containing vertices from both X and Y could only be generated by v . Assume there is such a cycle, and without loss of generality, it contains an edge between vertices $a, b \in X$ where a precedes b with respect to X . Because of the second part of the induction hypothesis, this edge is $a \rightarrow b$. However, such cycles are detected by Algorithm MFS.

After flipping the minimal feedback set, the graph induced by $list_v$ contains an instance of Cole's merge-sort algorithm. Therefore, the correctness of the second claim follows from the correctness of Cole's algorithm. $\square$

THEOREM 4.4. *The time complexity of the modified algorithm is $O(\log n)$.*

Proof. A more detailed inspection of the algorithm shows that in each CPB graph at least one of the paths is of length at most three. In Cole's algorithm each node in the tree gets $O(|X_{i+1}| + |Y_{i+1}|)$ processors; this number is sufficient to calculate the calls for algorithm MFS. $\square$

Acknowledgments. We would like to thank Eli Gafni, Nati Linial, and Moni Naor for valuable discussions.

REFERENCES

- [AA1] N. ALON AND Y. AZAR, *Sorting, approximate sorting, and searching in rounds*, SIAM J. Disc. Math., 1 (1988), pp. 269–280.
- [AA2] ———, *The average complexity of deterministic and randomized parallel comparison sorting algorithms*, Proceedings 28th Annual IEEE Foundations of Computer Science, Los Angeles, CA, 1987, pp. 489–498; SIAM J. Comput., 17 (1988), pp. 1178–1192.

- [AAV] N. ALON, Y. AZAR, AND U. VISHKIN, *Tight complexity bounds for parallel comparison sorting*, Proceedings 27th Annual IEEE Foundations of Computer Science, Toronto, Ontario, Canada, 1986, pp. 502–510.
- [AKS] M. AJTAI, J. KOMLÓS, AND E. SZEMERÉDI, *Sorting in $c \log n$ parallel steps*, Combinatorica, 3 (1983), pp. 1–19.
- [AKSS] M. AJTAI, J. KOMLÓS, W. STEIGER, AND E. SZEMERÉDI, *Deterministic selection in $O(\log \log n)$ parallel time*, Proceedings 18th Annual ACM Symposium on Theory of Computing, Berkeley, CA, 1986, pp. 188–195.
- [Al1] N. ALON, *The maximum number of Hamiltonian paths in a tournament*, manuscript.
- [Al2] ———, *Eigenvalues, geometric expanders, sorting in rounds and Ramsey theory*, Combinatorica, 6 (1986), pp. 207–219.
- [AV] Y. AZAR AND U. VISHKIN, *Tight comparison bounds on the complexity of parallel sorting*, SIAM J. Comput., 3 (1987), pp. 458–464.
- [Ba] K. E. BATCHER, *Sorting networks and their applications*, in Proc. AFIPS Spring Joint Comput. Conf., Vol. 32, April 1968, pp. 307–314.
- [Be] C. BERGE, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973.
- [Bea] P. BEAME, *Limits on the power of concurrent-write parallel machines*, Proceedings ACM 18th Symposium on Theory of Computing, Berkeley, CA, 1986, pp. 169–176.
- [BHe] B. BOLLOBÁS AND P. HELL, *Sorting and graphs*, in Graphs and Orders, I. Rival, ed., D. Reidel, Boston, MA, 1985, pp. 169–184.
- [BHo] A. BORODIN AND J. HOPCROFT, *Routing, merging and sorting on parallel models of computation*, J. of Comput. Systems Sci., 30 (1985), pp. 130–145.
- [Co] R. COLE, *Parallel merge sort*, SIAM J. Comput., 17 (1988), pp. 770–785.
- [GJ] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability*, W. H. Freeman, San Francisco, 1979.
- [GR] A. GIBBONS AND W. RYTTER, *Efficient Parallel Algorithms*, Cambridge University Press, Cambridge, 1988.
- [HR] P. HELL AND M. ROSENFELD, *The complexity of finding generalized paths in tournaments*, J. Algorithms, 4 (1983), pp. 303–309.
- [Kr] C. P. KRUSKAL, *Searching, merging and sorting in parallel computation*, IEEE Trans. Comput., 32 (1983), pp. 942–946.
- [Mo] J. W. MOON, *Topics on Tournaments*, Holt, Rinehart, and Winston, New York, 1968.
- [Na] JOSEPH NAOR, *Two parallel algorithms in graph theory*, Technical Report CS-86-6, Dept. of Computer Science, Hebrew University, Jerusalem, June 1986.
- [P] N. PIPPENGER, *Sorting and selecting in rounds*, SIAM J. Comput., 16 (1987), pp. 1032–1038.
- [Re] L. REDEI, *Ein kombinatorischer satz*, Acta Litt. Sci. Szeged, 7 (1934), pp. 39–43.
- [So] D. SOROKER, *Fast parallel algorithms for finding Hamilton paths and cycles in a tournament*, J. Algorithms, 9 (1988), pp. 276–286.
- [Va] L. G. VALIANT, *Parallelism in comparison problems*, SIAM J. Comput., 4 (1975), pp. 348–355.

PARALLEL SELECTION WITH HIGH PROBABILITY*

BÉLA BOLLOBÁS†‡ AND GRAHAM BRIGHTWELL†

Abstract. Given a set of n elements in some unknown order, parallel comparison algorithms to select the t th highest with probability $1 - o(1)$ as $n \rightarrow \infty$ are considered, where each order is assumed to be equally likely. Such an algorithm is given using four rounds and cn comparisons per round, and it is shown that no such algorithm exists using three rounds and cn comparisons per round.

Key words. selecting, median, parallel comparison algorithms

AMS(MOS) subject classification. 68E05

Given a set X of n elements which are in some unknown order, the *selection problem* is that of determining the t th highest element of X by making comparisons, i.e., by asking questions of the form: “Is x below y ?”. There has been a considerable amount of interest in this problem; we survey some of the known results below. The two most interesting special cases are $t = 1$ and $t = \lfloor n/2 \rfloor$, the problems of finding the top element and the median element, respectively. Of course we may assume by symmetry that $t \leq \lfloor n/2 \rfloor$. We shall always take t to be a function of n . We are interested in the problem only when n is large, and all statements we make are supposed to apply just for sufficiently large n or as $n \rightarrow \infty$.

Let us first consider the case where we make the comparisons serially, so that we know the answers to all previous questions before we have to formulate the next. It is rather well known that the minimum number of comparisons required to determine the t th highest element with probability 1, i.e., for every total order of X , is $O(n)$. For selecting the highest element, $n - 1$ comparisons are necessary and sufficient. The current best bounds for selecting the median are that $3n + o(n)$ comparisons always suffice (Schönhage, Paterson and Pippenger [12]), and that $2n + o(n)$ comparisons are necessary (Bent and John [4]). Note however that there is an algorithm due to Floyd and Rivest [8] which, with probability $1 - o(1)$, finds the median after $3n/2 + o(n)$ comparisons. This last result was proved best possible by Cunto and Munro [7]. Shortly we shall give a brief description of Floyd and Rivest’s algorithm.

Let us now turn to parallel algorithms. A parallel algorithm consists of a number k of rounds, taken in sequence, in each of which we make a number p of comparisons. The important feature is that we have to formulate all the questions in a given round simultaneously, before we know the results of any comparisons made in that round. The total number of comparisons used by such an algorithm is kp . If kp is of the same order of magnitude as the number of comparisons used in the best serial algorithm, the parallel algorithm is said to achieve *optimal speed-up*. Thus, in the case of selection, a parallel algorithm achieves optimal speed-up if $kp = O(n)$.

Valiant [14] was the first to consider parallel selection algorithms. He gave an algorithm to find the highest element using $O(\log \log n)$ rounds and performing $O(n/\log \log n)$ comparisons per round. Thus this algorithm achieves optimal speed-up. Valiant also proved that the $\log \log n$ term is best possible: if $k(n) = o(\log \log n)$, there is no algorithm which, for every n , finds the highest element with probability 1

* Received by the editors March 7, 1988; accepted for publication (in revised form) April 12, 1989.

† Department of Pure Mathematics and Mathematical Statistics, University of Cambridge, 16 Mill Lane, Cambridge CB2 1SB, United Kingdom. This work was partially supported by MCS grant 8104854.

‡ The work of this author was partially supported by National Science Foundation grant DMS-8806097.

using $k(n)$ rounds and $n/k(n)$ comparisons per round. We say that $k = O(\log \log n)$ is the *break-point* for the problem of finding the highest element with probability 1. More recently Ajtai, Komlós, Steiger, and Szemerédi [1] proved that the situation is the same for the median. They gave an algorithm which runs in $O(\log \log n)$ rounds and uses $O(n/\log \log n)$ comparisons per round, which finds the median with probability 1. This, together with Valiant's result, shows that the break-point for the problem of finding the median with probability 1 is again $k = O(\log \log n)$. For some recent bounds on the number of comparisons required to find the median with probability 1, using only a constant number of rounds, see Alon and Azar [2].

What are the corresponding results if we require only that our algorithm succeeds almost surely, i.e., for all but $o(n!)$ total orderings of X ? For this problem, Reischuk [11] gave an algorithm which uses a fixed constant number of rounds and n comparisons per round, and selects the t th highest element with probability at least $1 - 2^{-cn^{1/8}}$, for some constant c . The actual number of rounds used by Reischuk's algorithm is about 23. In the other direction, Alon and Azar [3] recently announced that at least $cn^{4/3}$ comparisons are required *on average* to find the maximum in two rounds. Of course, the problems of average case selection and almost sure selection are extremely closely related, and this result certainly implies that two rounds are insufficient to find the maximum almost surely with just $O(n)$ comparisons per round.

In this paper, we improve on Reischuk's result by giving a 4-round algorithm using cn comparisons per round which finds the median with probability at least $1 - e^{-n^{1/8-\epsilon}}$. Furthermore, we show that four rounds is the least possible, even for the simpler problem of finding the maximum: for every 3-round procedure using at most cn questions per round, the probability that the procedure fails to determine the maximum is at least $\delta = \delta(c) > 0$.

Three rounds. Let us now describe a slightly modified form of the Floyd–Rivest algorithm [8] mentioned earlier. As will be seen, this algorithm readily lends itself to at least some degree of parallelisation. To fix ideas, let us suppose we are looking for the median. The Floyd–Rivest algorithm finds the median using at most $3n/2 + o(n)$ comparisons with probability $1 - o(1)$ as $n \rightarrow \infty$. The main idea is to take a “sample” S of the elements, say $n^{2/3} \log n$ elements chosen at random from X . First we find the elements of S which are the $(|S|/2 \pm n^{1/3} \log n)$ th highest. For instance, this can be done deterministically using the Schönhage, Paterson, and Pippenger algorithm [12] with $6|S| + o(|S|)$ comparisons. The two elements we find are *markers*: with high probability the median of X lies between the markers, but the number of elements of X between them is relatively small—almost surely at most $n^{2/3} \log n$. Next we compare every element of X with the bottom marker. Those elements that are above the bottom marker we then compare with the top marker. The problem is now (almost surely) reduced to finding the t th highest element of the set T of elements of X remaining between the two markers, where t is some known integer and $|T| \leq n^{2/3} \log n$. Again, we can use the Schönhage, Paterson, and Pippenger algorithm for this.

The Floyd–Rivest algorithm thus uses $o(n)$ comparisons to find the two markers, $o(n)$ comparisons to find the t th highest element of T , and about $3n/2$ comparisons of elements with the markers: every element is compared with the bottom marker and about half survive to be compared with the top marker. In general, $n + t + o(n)$ comparisons are required by the algorithm to find the t th highest, with probability $o(1)$ of failure.

Let us consider briefly how to convert this to a parallel algorithm. As we have seen, almost all of the comparisons are used in the stage where we compare all elements of X with the two markers: this can be done in one round using $2n$ questions. However, the other two steps do not lend themselves so readily to parallelisation. In particular, to find

the median of a set T in one round using $O(n)$ comparisons, we would have to compare almost all pairs, and so we would need $|T| = O(n^{1/2})$.

Later we will show that, with four rounds at our disposal, we can surmount this difficulty. We shall give an algorithm, which is fairly similar to the Floyd–Rivest algorithm, using just four rounds and a total of $2n + o(n)$ questions. With five rounds, we can reduce this to $3n/2 + o(n)$ comparisons.

In this section, we will show that three rounds are not enough. Suppose that we are attempting to find the t th highest element of an n -set using cn comparisons in each of three rounds. We shall prove that the probability of failure must be at least $\delta = \delta(c) > 0$. Let us first show that we may as well assume that $t = 1$. Suppose that $1 < t \leq n/2$, and our task is to determine the t th highest. Equivalently, we have to find the set consisting of the $t - 1$ highest elements, and the maximum of the remaining $n - t + 1$ elements. Thus if our 3-round algorithm, using cn comparisons per round, finds the t th highest with probability $o(1)$ of failure as $n \rightarrow \infty$, this algorithm must also serve to find the maximum of an $(n - t + 1)$ -set, with probability $o(1)$ of failure and at most $2c(n - t + 1)$ comparisons per round. For the remainder of this section, we concentrate on the problem of determining the maximum.

Let us now show that three rounds are not enough to find the median almost surely using just $O(n)$ comparisons per round. How would a three-round algorithm run? If we were to follow the sampling approach, we would first take a sample of size $(2cn)^{1/2}$, and in the first round compare all pairs of elements in the sample. Thus we find the highest element x in the sample. In the second round we compare every element with x , and in the final round we compare every pair of elements which “beat” x . This algorithm is successful precisely when there is an element of the sample in the top $(2cn)^{1/2}$, and this event has probability about $1 - e^{-2c}$. It is possible to improve slightly on this algorithm, but what we shall prove is that there is always a positive probability of failure.

If we replace the constant c in the above argument with any function $\omega(n)$ tending to infinity with n , we see that $n\omega(n)$ comparisons per round suffice to find the maximum in three rounds almost surely. Thus the following result is tight.

THEOREM 1. *Let X be a set of n elements, and let $<$ be a linear order of X , chosen at random from the set of all $n!$ linear orders. Let c be a positive constant, and let $\mathcal{A}$ be a 3-round parallel algorithm, independent of $<$, which, in each round, compares at most cn pairs of elements of X . Then, for n sufficiently large, the probability that $\mathcal{A}$ fails to determine the maximum of $(X, <)$ is at least $\delta = \delta(c) > 0$.*

Before embarking on the proof itself, let us give a brief description. Throughout this discussion and proof, we shall assume without explicit reference that n is sufficiently large for various estimates to hold. We shall also omit integrality signs, where these do not affect the argument.

We assume that an adversary, called the *Questioner*, attempts, using at most cn comparisons per round, to determine the maximum of a set X of n elements in some randomly chosen linear order unknown to him. The Questioner can use comparisons in the first round in essentially one of two ways. First, he can eliminate some vertices from consideration by finding a larger element. However, whatever first-round comparisons are made, there is almost surely a constant proportion of the vertices remaining as “candidates” for the maximum. Second, the Questioner can attempt to find one or more elements that are likely to be high in the underlying order, with which he intends to compare elements in the second round. We show that success cannot be guaranteed. Suppose that, after the results of the first-round comparisons are known, the Questioner then specifies, for each candidate x , the number of comparisons involving x in the second round. For almost all candidates x , this number is bounded above by a constant d . Now suppose that the Questioner is informed of the set T of elements constituting the top

$\alpha n^{1/2}$ in the underlying order, where $\alpha = \alpha(c)$ is a large constant. This information cannot harm the Questioner's probability of success. However, with positive probability, this set has the following properties. First, there is a set U of at least $\beta n^{1/2}$ elements of the set T that are candidates to be compared with at most d elements in round two. Second, the poset induced on T by the comparisons in round one consists just of isolated vertices and components of order two (i.e., pairs of elements linked by a relation to each other but to no other vertex of T). This second condition means that we have, in the first round, failed to distinguish a very high element. The best that the Questioner can now do is to compare each element of U with at most d elements, each higher than at most one other element of T . With high probability, this process leaves us with $Cn^{1/2}$ elements of U which are still candidates, where $C > \sqrt{2}c$, and in the final round we must compare almost every pair of such elements, requiring too many comparisons.

The proof of Theorem 1 makes use of the FKG inequality of Fortuin, Kasteleyn, and Ginibre [9], which we state below.

THEOREM 2. *Let $\mathcal{L}$ be a finite distributive lattice, and let f, g, μ be functions from $\mathcal{L}$ to $\mathbb{R}^+$ such that f and g are increasing and $\mu(L)\mu(M) \leq \mu(L \vee M)\mu(L \wedge M)$ for every $L, M \in \mathcal{L}$. Then*

$$\left(\sum_{L \in \mathcal{L}} f(L)\mu(L) \right) \left(\sum_{L \in \mathcal{L}} g(L)\mu(L) \right) \leq \left(\sum_{L \in \mathcal{L}} f(L)g(L)\mu(L) \right) \left(\sum_{L \in \mathcal{L}} \mu(L) \right).$$

There have been several combinatorial applications of Theorem 2; for one example related to our use of the result below, see Shepp [13]. In many applications of Theorem 2, one sets f and g equal to indicator functions of up-sets and $\mu \equiv 1$. For our present purposes, we require at least g to be a little less trivial.

Proof of Theorem 1. As above, we shall assume that a Questioner is attempting to determine the maximum element of X using at most cn questions in each of three rounds. Let us first consider the effect of the comparisons made by the Questioner in the first round. After the first round, the information available to the Questioner is in the form of a partial order $<$ on X , with $x < y$ if he can deduce from the results of the first-round comparisons that x is below y in the underlying total order. A *candidate* is a maximal element in $(X, <)$. Thus a candidate is just an element that could be the highest, given the first-round results.

A *linear extension* of a partial order $<$ on X is a bijection λ from X to $[n] = \{1, \dots, n\}$ such that $\lambda x < \lambda y$ whenever $x < y$. Thus linear extensions are in 1-1 correspondence with linear orders on X consistent with the information $<$. Hence, from the Questioner's point of view, each linear extension of $(X, <)$ is equally likely to "be" the underlying total order. We shall work in the probability space of linear extensions of $(X, <)$, with all linear extensions having the same probability.

First we shall define an event Ω , which has positive probability, on which we shall condition various other events. We set $\alpha = 2 \times 10^5 c^{7/2}$, and pay special attention to the set T consisting of the top $\alpha n^{1/2}$ vertices of $(X, <)$. Let Y be the set of vertices that are compared with more than $\delta n^{1/2}$ elements in the first round, where $\delta = 1/(8c\alpha^3)$. Note that $|Y| \leq 2cn^{1/2}/\delta$. Let Ω be the event that the order $<$ on X is such that no element of Y is in T . We see that

$$\begin{aligned} P(\Omega) &= \frac{\binom{n - \alpha n^{1/2}}{2cn^{1/2}/\delta}}{\binom{n}{2cn^{1/2}/\delta}} \\ &\geq \left(\frac{n - \alpha n^{1/2} - 2cn^{1/2}/\delta}{n} \right)^{2cn^{1/2}/\delta} \geq \frac{1}{2} \exp(-2c\alpha/\delta - 4c^2/\delta^2), \end{aligned}$$

which is bounded away from 0 independently of n .

The next few steps of the proof consist of proving that, conditional on Ω , certain events have high probability. Let us calculate the expected number of triples $a, b, c \in T$ such that the comparisons $a : b$ and $b : c$ are both carried out in the first round, conditional on Ω . For each first-round comparison $a : b$ such that neither a nor b is in Y , there are at most $2\delta n^{1/2}$ vertices c such that either $a : c$ or $b : c$ is a first-round comparison. For each triple a, b, c of elements not in Y , the probability that all of a, b, c are in T , conditional on Ω , is at most

$$\binom{n - 2cn^{1/2}/\delta}{\alpha n^{1/2} - 3} \bigg/ \binom{n - 2cn^{1/2}/\delta}{\alpha n^{1/2}} \geq 2 \left(\frac{\alpha}{n^{1/2}} \right)^3.$$

Thus the expected number of triples a, b, c such that at least two of the three comparisons $a : b, a : c, b : c$ are carried out is at most

$$\frac{cn \cdot 2\delta n^{1/2}}{2} \cdot 2 \left(\frac{\alpha}{n^{1/2}} \right)^3 = 2c\delta\alpha^3 = \frac{1}{4}.$$

Hence the probability that there is such a triple is at most $\frac{1}{4}$.

The next step is to bound the probability, conditional on Ω , that there are few candidates after the first round. Define the random variable Z to be the number of first-round comparisons which rule out a vertex in the top $n/(8c)$. What is the probability, conditional on Ω , that a given comparison does rule out one of the top $n/(8c)$ vertices? This is just the probability that both vertices involved are in the top group, and this probability is at most

$$\left(\frac{n/(8c)}{n - 2cn^{1/2}/\delta} \right)^2 \leq \frac{1}{60c^2}.$$

Therefore $\mathbb{E}(Z|\Omega) \leq n/(60c)$, and $P(Z > n/(15c)|\Omega) \leq \frac{1}{4}$. Thus with probability at least $\frac{3}{4}$, at least $n/(20c)$ of the vertices are candidates after the first round.

Based on the first-round comparisons only, the Questioner must now specify the comparisons to be made in the second round. We relax the requirements a little, and demand only that he specifies, for each candidate x , the number of second-round comparisons involving x . If there are at least $n/(20c)$ candidates, then there are at least $n/(40c)$ which are due to be compared with at most $80c^2$ elements in the second round. Let S be the set of elements x with the following three properties.

- (i) x was involved in at most $\delta n^{1/2}$ first-round comparisons ($x \notin Y$);
- (ii) x is due to be involved in at most $80c^2$ second-round comparisons;
- (iii) x is a candidate after the first round.

Thus, conditional on there being at least $n/(20c)$ candidates after the first round, $|S| \geq n/(50c)$. What we now show is that, conditional on Ω and the results of the first-round comparisons, the intersection of S with T is, with high probability, at least half as large as it should be.

If the elements of S had not been compared with any other vertices in the first round, we could very readily calculate the probability that at least a certain number of them were in T , the set consisting of the top $\alpha n^{1/2}$ vertices. What we claim in the lemma below is that any additional information telling us that elements of S are above other vertices cannot decrease this probability. This is intuitively rather obvious, but as with many similar assertions, there does not seem to be a correspondingly trivial proof. If it were not for the conditioning on Ω , the result would follow from Shepp's Theorem [13]. However, the conditions imposed by Ω are of the wrong type to allow us to repeat the proof of that theorem, and we have to adopt a slightly different approach.

For n, s, t, r natural numbers, let $p(n, s, t, r)$ denote the probability that an s -set and a t -set, chosen independently and at random from an n -set, intersect in at least r elements.

LEMMA 3. *Let $(X, <)$ be a finite poset with $|X| = n$, and let r, s, t be natural numbers. Let S be a set of s maximal elements, and let Y be any subset of X disjoint from S . For a linear extension $<$ of $(X, <)$, let $T = T(<)$ be the set consisting of the top t elements in $<$. Then*

$$P(|S \cap T| \geq r \mid |Y \cap T| = \emptyset) \geq p(n, s, t, r).$$

Proof. Let $\mathcal{L}$ be the set of s -subsets of $[n]$, and impose a partial order on $\mathcal{L}$ by setting $L \leq M$ if, for every $k = 1, \dots, s$, the k th highest element of L is less than or equal to the k th highest element of M . This gives $\mathcal{L}$ the structure of a distributive lattice, as proved by Shepp [13] in his proof of the Graham, Yao, and Yao Inequality [10].

Let $T = [n - t + 1, n]$, the top t places in $[n]$. Now set $f(L) = 1$ if there are at least r elements of L in T , and $f(L) = 0$ otherwise. Set $g(L)$ equal to the number of linear extensions of $(X, <)$ such that:

- (i) the elements of S are in the places given by L ; and
- (ii) no element of Y is in T .

Also set $\mu(L) \equiv 1$.

The function μ certainly satisfies the condition in Theorem 2, and we claim that f and g are increasing. If L has at least the required number of elements in T , and $M > L$, then M also has the required number of elements in T . Thus f is increasing. Let λ be a linear extension satisfying (i) and (ii) above. The function λ is described uniquely by the orders it induces on S and $X \setminus S$, together with the set $\lambda S = L$. Thus if $M > L$, then λ gives rise to a unique bijection λ' from X to $[n]$ with the orders induced on S and $X \setminus S$ unchanged and $\lambda' S = M$. This bijection λ' is a linear extension, since the only changes we have made consist of putting elements of S , which are maximal in the poset, above some other elements. Also $\lambda' y \leq \lambda y$ for $y \in Y$, and so no element of Y is in T . Thus we have an injection from the set of linear extensions satisfying (i) and (ii) for L to those satisfying (i) and (ii) for M , and hence g is increasing on $\mathcal{L}$.

Thus the conditions of the FKG inequality, Theorem 2, are satisfied, and so

$$\frac{\sum f(L)g(L)}{\sum g(L)} \geq \frac{\sum f(L)}{|\mathcal{L}|}.$$

But the first ratio above is the probability that a linear extension in which no element of Y is in T also has at least r elements of S in T ; and the second ratio is the probability that a randomly chosen s -subset of $[n]$ contains at least r elements of T ; in other words, the second ratio is equal to $p(n, s, t, r)$. Hence we have the required result. $\square$

Of course Lemma 3 applies immediately to our present situation, with $t = \alpha n^{1/2}$ and $r = (ts)/(2n) \geq (\alpha n^{1/2})/(100c)$. It is apparent that, for these values of the parameters and n sufficiently large, $p(n, s, t, r) \geq \frac{3}{4}$.

Let us pause to take stock. Conditional on Ω , the event that no vertex of Y is in T , we have the following situation, with probability at least $1 - \frac{1}{4} - \frac{1}{4} - \frac{1}{4} = \frac{1}{4}$. First, T contains no element which has been compared with two other elements of T in the first round. Second, there is a set S of vertices which are candidates, due to be compared with at most $80c^2$ other elements in the second round, which intersects T in at least $(\alpha n^{1/2})/(100c)$ elements. The event Ω itself has probability bounded away from 0, and so with some probability bounded away from 0 independently of n we are in the above situation. From now on, we indeed assume that all the above conditions hold. Let $U = T \cap S$. So U is a set of at least $\alpha n^{1/2}/(100c)$ candidates in T , each due to be compared with at most $80c^2$ elements in round two.

At this point, the Questioner is informed of the set T of the top $\alpha n^{1/2}$ elements of the order. This is equivalent to giving him results of all comparisons between T and $X \setminus T$. From the Questioner's point of view, the probability of any event involving only comparisons between elements of T is independent of the order of $X \setminus T$. He must now specify exactly which comparisons are to be made in the second round. Of course, he loses nothing by comparing only pairs of elements of T , but he is constrained by having already specified the number of comparisons involving each element.

Let us fix an element x of U , and estimate the probability that it remains a candidate after the second round. The Questioner is comparing x with at most $80c^2$ other elements, each of which is greater than at most one other element of T . Thus the probability that x is found to be greater than all the elements with which it is compared is at least the probability that x is the highest in a set of at most $160c^2 + 1$ elements. Thus the probability that x is still a candidate after the results of the second-round comparisons are known is at least $1/(160c^2 + 1)$, and the expected number of elements of U which are candidates after the second round is at least $\alpha n^{1/2}/(2 \times 10^4 c^3)$. Since this number is bounded above by $\alpha n^{1/2}$, crudely, the probability that at least $\alpha n^{1/2}/(10^5 c^3)$ elements of U are candidates after round two is bounded away from 0 by a constant independent of n .

In the final round, the Questioner has to compare most pairs of elements of U which have not yet been ruled out. Indeed, if x and y are elements of U which are still maximals after the second round, then the probability that they are the top two elements is at least $1/(\frac{\alpha n}{2})^{1/2}$. Thus if, for some $\varepsilon > 0$, he fails to compare εn pairs of candidates from U , the probability that he misses the top pair and so fails to determine the maximum is at least $2\varepsilon/\alpha^2$. Hence in the final round, the Questioner is obliged to make at least

$$\frac{\alpha^2 n}{2 \times 10^{10} c^6} = 2cn$$

comparisons, which is too many. Hence, with probability $\delta = \delta(c) > 0$, the Questioner fails to determine the maximum. $\square$

Four rounds. In this section, we present a four-round algorithm, using at most cn comparisons per round, which, with probability $o(1)$ of failure, finds the t th highest element of a set X in an unknown linear order. Before proceeding with the algorithm, let us show that we can restrict attention to the case when $t = n/2$. If $t < n/2$, we simply add in $n - 2t$ elements above everything in X , and use our algorithm to find the median of our new ordered set. This algorithm will of course run in four parallel rounds using at most $2cn$ questions per round, and will produce the t th highest element of our original set, with small probability of failure.

Throughout the following discussion, we shall assume that we have a set X of size n in some unknown linear order, and that we are to determine the median (taken to mean the $\lfloor n/2 \rfloor$ th highest element). For most purposes, we may assume that X is the set $[n]$ with its standard order.

A *random sample* of size s from a larger set Y is just an s -subset of Y chosen at random, each s -subset being equally likely. We first give a standard result concerning the behavior of random samples.

LEMMA 4. *Suppose that s, r, ω are functions of n satisfying: $s = o(n)$, $s/4 \leq r \leq 3s/4$, $\omega \leq r^{1/2}$, and $\omega \rightarrow \infty$. Let S be a random sample of size s from $X = [n]$, and let x be the r th smallest element of S . Then, for n sufficiently large and with probability at least $1 - 4/(\omega\sqrt{\pi})e^{-\omega^2/4}$, we have*

$$\left| x - \frac{rn}{s} \right| \leq \omega \frac{n}{s} r^{1/2}.$$

Proof. Let E be the event that $x > (rn)/s + \omega(n/s)r^{1/2}$. By symmetry it is sufficient to show that the probability of E is at most $2/(\omega\sqrt{\pi})e^{-\omega^2/4}$. But E is equivalent to the event that S contains fewer than r elements from the bottom $(n/s)(r + \omega r^{1/2})$ of $[n]$, and the number of elements of S in this range is essentially a binomial random variable with mean $r + \omega r^{1/2}$ and variance $(r + \omega r^{1/2})(1 - r/s - \omega r^{1/2}/s) \leq 2r$. By the DeMoivre–Laplace Theorem, the probability of E is thus at most $2(1/(\omega\sqrt{\pi}))e^{-\omega^2/4}$, for sufficiently large n , as desired. $\square$

THEOREM 5. *Let X be a set of n elements, and let $\varepsilon > 0$ be given. Let $<$ be a linear order of X , chosen at random from all the $n!$ linear orders. There is a four-round parallel algorithm $\mathcal{A}$, independent of $<$, which uses at most $2n + o(n)$ comparisons per round and finds the median of $(X, <)$ with probability at least $1 - e^{-n^{1/8-\varepsilon}}$.*

Proof. Set $\alpha = \frac{3}{4} - \varepsilon$ and $\delta = \frac{1}{16} - \varepsilon/4$. Our probability of failure will be bounded above by a term of order about $e^{-n^{2\delta}}$.

The algorithm goes as follows. In the first round we take two different samples from X . First we take a sample S of size $n^{1/2}$, randomly chosen from X . Then we take another sample T of size n^α from $X \setminus S$, again chosen at random. The idea is that the smaller sample S will provide us with a rough guess at the median after one round, in the form of two markers, one of which is almost surely just above half-way, and the other just below. We then compare all elements of X with these markers. Meanwhile, using two rounds, we completely order the middle section of T , thus providing several very accurate markers with which to compare all those elements lying between the S -markers. In the final round, we compare every pair of elements which could still be the median.

In the first round, then, we compare all pairs of elements of S . Also, we take a graph G with vertex set T , just n edges and the property that every pair U, V of disjoint subsets of T of size $n^{2\alpha-1}(\log n)^2$ have an edge between them; and compare each pair of elements of T joined by an edge of G .

Using the techniques of random graphs, it is easy to show that such graphs G exist. In fact, this follows immediately from Theorem II.12 of Bollobás [5].

What we now show is that, no matter what the order $<|_T$ is, the results obtained from comparing pairs of vertices joined by an edge of G tell us a lot about the order: in particular, we can deduce, for most elements, that they are not close to the median. For this, we need a result from a previous paper [6] by the current authors. This result is not explicitly stated as a theorem but can be read immediately from the proof of Theorem 7 of [6].

LEMMA 6. *Suppose a graph G has the property that, for every pair (U, V) of subsets of $V(G)$ of size l , there is an edge from U to V . Let $<$ be any total ordering of $V(G)$ and let $P(G; <)$ be the partial order obtained by comparing each pair of vertices which are joined by an edge of G . (Thus $x < y$ in $P(G; <)$ if and only if there is a path $x = x_1 x_2 \cdots x_{r-1} x_r = y$ in G with $x_i < x_{i+1}$ for each i .) Let S be a set of $6l$ vertices consecutive in $<$. Then there are at least $5l$ elements x of S with the property that there are at most $7l \log l$ vertices y such that $y < x$ but $y \not< x$ in $P(G; <)$.*

Let us apply this result to our present situation, with G our graph on T , and $l = n^{2\alpha-1}(\log n)^2$. Let S be the set of vertices which are at heights $\frac{1}{2}n^\alpha + 8l \log l + 1$ to $\frac{1}{2}n^\alpha + 8l \log l + 6l$ inclusive in $<$. Let x be one of the at least $4l$ elements of S with the property that there are at most $7l \log l$ vertices above x in $<$ but not in $P(G; <)$ and at most $7l \log l$ vertices below x in $<$ but not in $P(G; <)$. Now consider all those vertices above x in $P(G; <)$. There are at least $\frac{1}{2}n^\alpha - 8l \log l - 6l - 7l \log l \geq \frac{1}{2}n^\alpha - 16l \log l$ of these, and they are all above every element below x in $P(G; <)$. There are at least $\frac{1}{2}n^\alpha + l \log l$ such vertices below x .

Therefore, when we compare pairs of elements of T according to our graph G , whatever results we get, we are able to deduce, for at least $\frac{1}{2}n^\alpha - 16l \log l$ elements y of

T , that y is not in the bottom $\frac{1}{2}n^\alpha + l \log l$ vertices of T . Similarly there are $\frac{1}{2}n^\alpha - 16l \log l$ elements which cannot be in the top $\frac{1}{2}n^\alpha + l \log l$. Combining these, we see that there are at most $32l \log l$ vertices which can be within $l \log l$ of the median of T .

The point of this is that, with high probability, we have caught the median, i.e., the median of $(X, <)$ lies between the $(|T|/2 - l \log l)$ th and the $(|T|/2 + l \log l)$ th elements of T . Indeed let y_1 and y_2 be the elements of T at heights $n^\alpha/2 - n^{\alpha/2+\delta}$ and $n^\alpha/2 + n^{\alpha/2+\delta}$, respectively, in T . (We are not claiming to have determined these elements as a result of first-round comparisons.) By Lemma 4, with $\omega = n^\delta$, the number of elements of X between y_1 and the $(n/2 - n^{1-\alpha/2+\delta})$ th element of X is at most

$$n^{1-\alpha+\delta} \left(\frac{n^{\alpha/2}}{\sqrt{2}} + O(n^{3\alpha/8}) \right),$$

with probability of failure at most $\frac{1}{6}e^{-n^{2\delta/4}}$. Thus almost surely y_1 is below the median of X , and similarly y_2 is above. Also the number of elements between y_1 and y_2 in $(X, <)$ is at most $4n^{1-\alpha/2+\delta}$. Of course, we must keep track of the probability of failure: here it is at most $\frac{1}{3}e^{-n^{2\delta/4}}$.

In the second round, we compare each pair of elements of T which can be within $l \log l$ of the median of T . As noted above, there are at most $32l \log l = o(n^{1/2})$ of these, so we can do this using $o(n)$ comparisons. However, since $l \log l > n^{\alpha/2+\delta}$, after we have performed these comparisons we have identified y_1 and y_2 and totally ordered the set V of all elements of T between them.

The aim is, in the third round, to compare elements of X with many of the elements between y_1 and y_2 . In order to do this, we must cut down the number of elements of X under consideration. Note that we still have $2n$ second-round questions at our disposal: we use these to compare all elements of X with suitably chosen elements of our smaller sample S .

Recall that S is a random sample of size $n^{1/2}$ chosen from X , and that in the first round we compared every pair of elements of S . Let x_1 and x_2 be the elements at height $n^{1/2}/2 - n^{1/4+\delta}$ and $n^{1/2}/2 + n^{1/4+\delta}$, respectively, in $(S, <)$. As before, we apply Lemma 4 with $\omega = n^\delta$ and obtain that the number of elements between x_1 and the $(n/2 - n^{3/4+\delta})$ th element of X is at most

$$n^{1/2+\delta} \left(\frac{n^{1/4}}{\sqrt{2}} + O(n^{3/16}) \right),$$

with probability at least $1 - \frac{1}{6}e^{-n^{2\delta/4}}$. Thus, with probability at least $1 - \frac{1}{3}e^{-n^{2\delta/4}}$, we have that x_1 is below the median, x_2 above the median, and there are at most $4n^{3/4+\delta}$ elements of X between the two.

Using $2n$ second-round questions, we compare every element of X with x_1 and x_2 . Almost certainly the median lies between the two, so we can discard all elements below x_1 or above x_2 . Let U be the set of elements of X between the two. As shown above, with high probability, $|U| \leq 4n^{3/4+\delta}$.

In the third round, we bring together the two strands of our algorithm. For each element of U , we can "afford" $O(n^{1/4-\delta})$ comparisons involving that element. We choose to compare elements of U with suitably selected elements of V .

Recall that V is a totally ordered set consisting of those elements of T between y_1 and y_2 . We now take $n^{1/4-\delta}$ markers equally spaced in V . In other words, we take every $(|V|/n^{1/4-\delta})$ th element of $(V, <)$. Then we compare the elements of U with these markers, using at most n questions. Having done this, we know which pair z_1, z_2 of consecutive markers has the median of X between them, and also which elements of X

lie between z_1 and z_2 . In the fourth and final round, we compare each pair of elements between z_1 and z_2 . This process determines the median. Furthermore, we have used at most $2n + o(n)$ comparisons in each round provided there are at most $2n^{1/2}$ elements of X between z_1 and z_2 . This is certainly the case when we have a V -marker amongst the $n^{1/2}$ elements immediately above the median and another marker amongst the $n^{1/2}$ elements just below the median. This in turn is the case whenever there are at least $|V|/n^{1/4-\delta} \leq 2n^{\alpha/2-1/4+2\delta}$ elements of T in the $n^{1/2}$ elements just above the median, and similarly below.

The probability that there are fewer than $2n^{\alpha/2-1/4+2\delta}$ elements of T in a given range of size $n^{1/2}$ is at most the probability that a binomial random variable with parameters $|T|$ and $(n^{1/2} - 2n^{\alpha/2-1/4+2\delta})/n \geq \frac{1}{2}n^{-1/2}$ takes a value smaller than $2n^{\alpha/2-1/4+2\delta} \leq \frac{1}{4}n^{\alpha-1/2}$. This probability is at most $e^{-1/25n^{\alpha-1/2}} \leq \frac{1}{6}e^{-n^{2\delta/4}}$.

Thus the probability that there are more than $n^{1/2}$ elements between z_1 and z_2 is at most $\frac{1}{3}e^{-n^{2\delta/4}}$. Hence, with probability of failure at most $e^{-n^{2\delta/4}}$, we have found the median of X using four rounds and at most $2n + o(n)$ questions per round.

Suppose we want to cut down the total number of comparisons to the value $3n/2$ given by the Floyd–Rivest algorithm [8] and proved optimal by Cunto and Munro [7]. In the above algorithm there is a fair amount to spare, and it is trivial to adapt the algorithm to use $o(n)$ comparisons at every stage except that of comparing each element of X with x_1 and x_2 . Thus this adapted algorithm uses a total of $2n + o(n)$ comparisons. This can be reduced to $3n/2 + o(n)$ comparisons by using an extra round: instead of comparing everything with both x_1 and x_2 all in the same round, we compare all elements of X with x_1 in one round, and those elements which are above x_1 we then compare with x_2 in the next round. With our usual probability of failure, there are at most $n/2 + o(n)$ elements above x_1 , so the total number of comparisons is as required. With some additional work, we can probably prove that this extra round is needed if we want to use only $3n/2 + o(n)$ comparisons.

There are several variations on the algorithm above which also satisfy our requirements. We have chosen the algorithm as it stands in an effort to minimise the probability of failure: some other variations are rather simpler to present.

There are several questions left open. As we saw earlier, there is an algorithm using $n\omega(n)$ questions in each of three rounds which finds the *maximum*, almost surely, where $\omega(n)$ is any function tending to infinity. Is there also such an algorithm to find the *median*? The problem of finding the median does seem a little harder given these requirements: to find the maximum we sort a sample and compare everything with the *largest* element: there is no precise equivalent when searching for the median. A related question is that concerning the *average* number of comparisons per round needed in a three-round algorithm: to find the maximum there is an algorithm (essentially the one mentioned above, cf. Alon and Azar [3]) which uses $O(n)$ comparisons per round on average. Is there such an algorithm to find the median?

REFERENCES

- [1] M. AJTAI, J. KOMLÓS, W. STEIGER, AND E. SZEMERÉDI, *Deterministic selection in $O(\log \log n)$ time*, in Proceedings of the 18th Annual ACM Symposium on Theory of Computing, 1986.
- [2] N. ALON AND Y. AZAR, *Sorting, approximate sorting and searching in rounds*, SIAM J. Discrete Math., 1 (1988), pp. 269–280.
- [3] ———, *The average complexity of deterministic and randomized parallel comparison-sorting algorithms*, SIAM J. Comput., 17 (1988), pp. 1178–1192.
- [4] S. W. BENT AND J. W. JOHN, *Finding the median requires $2n$ comparisons*, in Proceedings of the 17th Annual ACM Symposium on Theory of Computing, (1985), pp. 213–216.

- [5] B. BOLLOBÁS, *Random Graphs*, Academic Press, London, 1985.
- [6] B. BOLLOBÁS AND G. R. BRIGHTWELL, *Transitive orientations of graphs*, SIAM J. Comput., 17 (1988), pp. 1119–1133.
- [7] W. CUNTO AND J. I. MUNRO, *Average case selection*, in Proceedings of the 16th Annual ACM Symposium on Theory of Computing, 1984, pp. 369–375.
- [8] R. W. FLOYD AND R. L. RIVEST, *Expected time bounds for selection*, Comm. ACM, 18 (1975), pp. 165–172.
- [9] C. M. FORTUIN, J. GINIBRE, AND P. N. KASTELEYN, *Correlation inequalities on some partially ordered sets*, Comm. Math. Phys., 22 (1971), pp. 89–103.
- [10] R. L. GRAHAM, A. C. YAO, AND F. F. YAO, *Some monotonicity properties of partial orders*, SIAM J. Algebraic Discrete Methods, 1 (1980), pp. 251–258.
- [11] R. REISCHUK, *A fast probabilistic sorting algorithm*, in Proceedings of the 22nd IEEE Symposium on Foundations of Computer Science, 1981, pp. 212–219.
- [12] A. SCHÖNHAGE, M. PATERSON, AND N. PIPPENGER, *Finding the median*, J. Comp. Sys. Sci., 13 (1976), pp. 184–199.
- [13] L. A. SHEPP, *The FKG inequality and some monotonicity properties of partial orders*, SIAM J. Algebraic Discrete Methods, 1 (1980), pp. 295–299.
- [14] L. G. VALIANT, *Parallelism in comparison problems*, SIAM J. Comput., 4 (1975), pp. 348–355.

AN ISOPERIMETRIC INEQUALITY ON THE DISCRETE TORUS*

BÉLA BOLLOBÁS†§ AND IMRE LEADER‡†

Abstract. The discrete torus is the graph on $\mathbb{Z}_k^n = (\mathbb{Z}/k\mathbb{Z})^n$ in which $x = (x_i)_1^n$ is joined to $y = (y_i)_1^n$ if for some i there is $x_i = y_i \pm 1$ and $x_j = y_j$ for all $j \neq i$. For a set $A \subset \mathbb{Z}_k^n$ and a natural number t , let $A_{(t)}$ be the set of vertices of $\mathbb{Z}_k^n$ within distance t of A . The main aim of this paper is to give a best possible lower bound for $|A_{(t)}|$ in terms of $|A|$, for even values of k .

Key words. isoperimetric inequalities, discrete isoperimetric inequalities, discrete tori.

AMS(MOS) subject classifications. 05A05, 05C35

For $k \geq 2$ and $n \geq 1$, consider the graph on $\mathbb{Z}_k^n = (\mathbb{Z}/k\mathbb{Z})^n$ in which $(x_i)_1^n$ and $(y_i)_1^n$ are joined if and only if for some i we have $x_i = y_i \pm 1$ and $x_j = y_j$ for $j \neq i$. Equivalently, this graph, which we call a *discrete torus* and for the sake of convenience denote by $\mathbb{Z}_k^n$, is the product of n cycles of length k if $k \geq 3$, and is the *discrete cube* $\{0, 1\}^n$ if $k = 2$.

Given a set A of vertices of a graph G and a natural number t , define the t -boundary of A as

$$A_{(t)} = \{x \in V(G) : d(x, A) \leq t\},$$

where as usual $d(x, A)$ denotes the length of a shortest path from x to A . Many authors refer to $A_{(t)}$ as the t -neighbourhood of A . We often write ∂A for $A_{(1)}$, and call it the *boundary* of A . It is easily seen that $(A_{(s)})_{(t)} = A_{(s+t)}$ for all s and t .

An inequality of the form

$$|A_{(t)}| \geq g_G(a, t) \quad \text{whenever } A \subset V(G) \quad \text{and} \quad |A| = a$$

is called an *isoperimetric inequality* on G . For a graph G , one would like to determine the best possible isoperimetric inequality for every pair (a, t) , i.e., the function

$$f_G(a, t) = \min \{|A_{(t)}| : A \subset V(G), |A| = a\}.$$

Ideally, we would even like to describe the *extremal* sets, i.e., the sets $A \subset V(G)$ such that $|A| = a$ and $|A_{(t)}| = f_G(a, t)$, for given values of a and t .

The prime example of an isoperimetric inequality is that on the discrete cube, proved by Harper [3] more than 20 years ago. Since then, isoperimetric inequalities have been studied by many authors and have found many applications in random graphs (see, e.g., [1]) and functional analysis (see, e.g., [4]).

Wang and Wang [6] posed the problem of determining which configurations in the discrete torus $\mathbb{Z}_k^n$ minimise the boundary among sets of a given size. In other words, Wang and Wang were asking for the best possible isoperimetric inequality for the discrete torus, at least for a reasonably dense sequence of values of a . Our aim in this paper is to solve this problem for *even* values of k and the sequence

$$a = a(r) = |\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r\}|, \quad r = 0, 1, \dots.$$

* Received by the editors January 4, 1988; accepted for publication (in revised form) April 12, 1989.

†† Department of Pure Mathematics and Mathematical Statistics, University of Cambridge, 16 Mill Lane, Cambridge CB2 1SB, England.

‡† Department of Mathematics, Louisiana State University, Baton Rouge, Louisiana 70803.

§ The work of this author was partially supported by National Science Foundation grant DMS-8806097.

We shall prove that if k is even then the sets $\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r\}$ are extremal for all t ; in other words, if $A \subset \mathbb{Z}_k^n$ satisfies

$$|A| = |\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r\}|$$

then

$$|A_{(t)}| \geq |\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r + t\}|.$$

It is really rather surprising that this holds for *all* values of r ; it shows that, in some sense, the “wraparound” feature of the torus does not make too much difference. In addition, we shall give fairly good estimates for $f_G(a, t)$ for all values of a .

The case when k is odd will be discussed at the end of the paper. As we shall see, the result above does not hold in this case; however, we shall give a conjecture stating that a slightly more complicated construction gives good lower bounds for the size of the boundary.

In the proof of our main result we shall make use of the notions of fractional systems and symmetrisation, to be defined below. These methods were introduced in [2]. A *fractional system*, or simply a *system*, on $\mathbb{Z}_k^n$ is a function from $\mathbb{Z}_k^n$ to the closed interval $[0, 1]$. Note that a fractional system on $\mathbb{Z}_k^n$ is a generalisation of a subset of $\mathbb{Z}_k^n$: if a function $f : \mathbb{Z}_k^n \rightarrow [0, 1]$ is such that $f(\mathbb{Z}_k^n) \subset \{0, 1\}$ then f is naturally identified with the subset $A = f^{-1}(1)$ of $\mathbb{Z}_k^n$.

The *support* of a system f is $\text{supp } f = \{x \in \mathbb{Z}_k^n : f(x) > 0\}$. We say that the systems $\{f_i : i \in I\}$ are *nested* if for any i and j we have either $f_i \leq f_j$ or $f_j \leq f_i$.

For a system f on $\mathbb{Z}_k^n$, the *weight* of f is $w(f) = \sum_{x \in \mathbb{Z}_k^n} f(x)$. The *boundary* of a system f is the system ∂f given by

$$\partial f(x) = \begin{cases} 1 & \text{if } f(x) > 0 \\ \max \{f(y) : d(y, x) = 1\} & \text{if } f(x) = 0 \end{cases}$$

Thus if $f(\mathbb{Z}_k^n) \subset \{0, 1\}$ and f is identified with $A = f^{-1}(1)$, then ∂f is identified with the usual boundary of A .

A system f which is of the form

$$f(x) = \begin{cases} 1 & \text{if } d(x, 0) < r \\ \alpha & \text{if } d(x, 0) = r \\ 0 & \text{if } d(x, 0) > r \end{cases}$$

for some $0 \leq r \leq nk/2$ and $\alpha \in [0, 1]$ is called a *fractional Hamming ball* or simply a *ball*. Note that our balls are always centred at 0. For $0 \leq v \leq k^n$ we write b^v for the (unique) ball of weight v .

Our aim is to show that, for even values of k , balls in $\mathbb{Z}_k^n$ have the smallest boundaries. In other words, we wish to show that the weight of the boundary of a system of weight v is at least $w(\partial b^v)$.

We start with the simple case $n = 1$; although in this case the assertion is easily checked, we state it as a lemma in order to emphasize it. Furthermore, we give a proof, since the corresponding assertion is false for odd values of k .

LEMMA 1. *Let $k \geq 2$ be even. Then in $\mathbb{Z}_k$ the weight of the boundary of a system f of weight v is at least the weight of the boundary of the ball b^v of weight v .*

Proof. If $|\text{supp } f| = k$ then ∂f is identically 1, so that certainly $w(\partial b^v) \leq w(\partial f)$.

If $|\text{supp } f| = k - 1$, say $\text{supp } f = \mathbb{Z}_k - \{k/2\}$, then $\partial f(x) = 1$ for $x \neq k/2$, and $\partial f(k/2) = \max(f(k/2 - 1), f(k/2 + 1))$. Since $b^v(k/2 - 1) = b^v(k/2 + 1)$ and

$b^v(k/2 - 1) + b^v(k/2 + 1) \leq f(k/2 - 1) + f(k/2 + 1)$, it follows that $\partial b^v \leq \partial f$, so that $w(\partial b^v) \leq w(\partial f)$.

If $|\text{supp } f| \leq 1$, then $w(\partial b^v) = w(\partial f)$.

Finally, if $2 \leq |\text{supp } f| \leq k - 2$ then it is easy to see that $w(\partial f) \geq w(f) + 2$. However, for all balls b we have $w(\partial b) \leq w(b) + 2$. $\square$

We remark that if $k \geq 3$ is odd then, in $\mathbb{Z}_k$, a ball of weight $k - 2$ has greater boundary than the complement of a ball of weight 2.

We now proceed to the case $n = 2$. It is somewhat misleading that this is the most cumbersome part of the proof since, in fact, it is only slightly less routine than the proof of Lemma 1.

LEMMA 2. *Let k be even. Then for a system f on $\mathbb{Z}_k^2$ we have $w(\partial f) \geq w(\partial b^v)$, where $v = w(f)$.*

Proof. Let $e_1 = (1, 0) \in \mathbb{Z}_k^2$, $e_2 = (0, 1) \in \mathbb{Z}_k^2$, and $\{i, j\} = \{1, 2\}$. For a system f on $\mathbb{Z}_k^2$ and a point $x \in \mathbb{Z}_k$, the i -section of f at x is the system $f_{i|x} : \mathbb{Z}_k \rightarrow [0, 1]$ defined by

$$f_{i|x}(y) = f(xe_j + ye_i), \quad y \in \mathbb{Z}_k.$$

Thus $w(f) = \sum_{x \in \mathbb{Z}_k} w(f_{i|x})$.

We wish to define a compression operator C_i , which acts on a system f by “compressing” each i -section of f into a (one-dimensional) ball of the same weight. Thus, for a system f on $\mathbb{Z}_k^2$, the system $C_i(f)$, the i -compression of f , is defined by

$$C_i(f)_{i|x} = b^{w(f_{i|x})}, \quad x \in \mathbb{Z}_k.$$

Thus $w(C_i(f)) = w(f)$. We say that f is i -compressed if $C_i(f) = f$.

We claim that, for any system f on $\mathbb{Z}_k^2$, we have $w(\partial C_i(f)) \leq w(\partial f)$. Indeed, writing f' for $C_i(f)$, it is sufficient to show that $w((\partial f')_{i|x}) \leq w((\partial f)_{i|x})$ for all $x \in \mathbb{Z}_k$.

Fix then an arbitrary $x \in \mathbb{Z}_k$. We have

$$(\partial f)_{i|x} = \partial(f_{i|x}) \vee \bigvee \{f_{i|y} : y \in \mathbb{Z}_k, d(y, x) = 1\}$$

where as usual $\vee$ denotes pointwise maximum, and a similar relation holds with f' in place of f . However, the system $\partial(f'_{i|x})$ and the systems $f'_{i|y}$ for $y \in \mathbb{Z}_k, d(y, x) = 1$ are nested, since they are all balls. Moreover, $w(f'_{i|y}) = w(f_{i|y})$ for all y , while Lemma 1 tells us that $w(\partial(f'_{i|x})) \leq w(\partial(f_{i|x}))$. It follows that $w((\partial f')_{i|x}) \leq w((\partial f)_{i|x})$, proving our claim.

Given a system f on $\mathbb{Z}_k^2$, let $G = \{g \in [0, 1]^{\mathbb{Z}_k^2} : w(g) = w(f) \text{ and } w(\partial g) \leq w(\partial f)\}$. We must show that there is a ball in G . This will be accomplished by a simple compactness argument. Note first that G is a (nonempty) compact subset of the product space $[0, 1]^{\mathbb{Z}_k^2}$. Furthermore, we know that if $g \in G$ then $C_i(g) \in G$ for $i = 1, 2$.

For $0 \leq r \leq k$, write $w^{(r)}(g)$ for $\sum_{x \in \mathbb{Z}_k^2, d(x, 0) = r} g(x)$, and let $H \subset G$ be the set obtained by maximising successively $w^{(0)}(g), w^{(1)}(g), \dots, w^{(k)}(g)$ on G . Note that for $g \in H$ and $i = 1, 2$ we must have $w^{(r)}(C_i(g)) = w^{(r)}(g)$, since otherwise, with $s = \min \{r : w^{(r)}(C_i(g)) \neq w^{(r)}(g)\}$, we would have $w^{(s)}(C_i(g)) > w^{(s)}(g)$, contradicting $g \in H$. Thus if $g \in H$ then also $C_i(g) \in H$ for $i = 1, 2$.

Now choose $g \in H$ with $w(g^{1/2})$ maximal. To prove the lemma, we shall show that this g is a ball. As $w^{(r)}(C_i(g)) = w^{(r)}(g)$ for all r , we have $w(C_i(g)^{1/2}) \geq w(g^{1/2})$, with equality if and only if $C_i(g) = g$. It follows from the choice of g that g is i -compressed for $i = 1, 2$. Thus, letting $r = \max \{d(x, 0) : g(x) > 0\}$, it suffices to show that g is constant on $\{x \in \mathbb{Z}_k^2 : d(x, 0) = r\}$.

So suppose, to the contrary, that g is not constant on $\{x \in \mathbb{Z}_k^2 : d(x, 0) = r\}$. Then, as g is i -compressed, we can find $x \in \mathbb{Z}_k^2$ such that $d(x, 0) = d(x - e_1 + e_2, 0) = r$ and $g(x) \neq g(x - e_1 + e_2)$: say $g(x) < g(x - e_1 + e_2)$. Then $d(x - e_1, 0), d(x + e_2, 0) =$

$r \pm 1$, with $d(x - e_1, 0) \neq d(x + e_2, 0)$: say without loss of generality $d(x - e_1, 0) = r - 1$ and $d(x + e_2, 0) = r + 1$.

Define a system g' on $\mathbb{Z}_k^2$ by

$$g'(z) = \begin{cases} c & \text{for } z = x, x - e_1 + e_2 \\ g(z) & \text{otherwise,} \end{cases}$$

where $c = \frac{1}{2}(g(x) + g(x - e_1 + e_2))$. Then $w(g') = w(g)$: we claim that in addition $w(\partial g') \leq w(\partial g)$. Indeed, it is easy to see that $\partial g'(z) \leq \partial g(z)$ for all $z \neq x + e_1$ (noting that, since g is 2-compressed, we have $g(x - e_1 - e_2) = 1$ or $g(x - e_1 + e_2) = 1$).

If $\partial g'(x + e_1) \leq \partial g(x + e_1)$ then certainly $w(\partial g') \leq w(\partial g)$. If, however, $\partial g'(x + e_1) > \partial g(x + e_1)$ then we must have $g(x + e_1 + e_2) < g(x - e_1 + e_2)$, and this implies $d(x + e_1 + e_2, 0) = r + 2$ (as if $d(x + e_1 + e_2, 0) = r$ then the fact that g is 1-compressed would give $g(x + e_1 + e_2) = g(x - e_1 + e_2)$). But then $\partial g'(x + e_2) = c = \partial g(x + e_2) - (c - g(x))$, whereas $\partial g'(x + e_1) \leq \partial g(x + e_1) + (c - g(x))$. Thus $\partial g'(x + e_1) + \partial g'(x + e_2) \leq \partial g(x + e_1) + \partial g(x + e_2)$, and so $w(\partial g') \leq w(\partial g)$, as claimed.

This shows that $g' \in G$, from which it follows that $g' \in H$. However, $w(g'^{1/2}) > w(g^{1/2})$, contradicting the choice of g . $\square$

Before embarking on the essential part of the paper, we need some more definitions. For $1 \leq i < j \leq n$ set

$$\mathbb{Z}_k^{ij} = \{(x_l)_l^n \in \mathbb{Z}_k^n : x_l = 0 \text{ for } l \neq i, j\}$$

and

$$\hat{\mathbb{Z}}_k^{ij} = \{(x_l)_l^n \in \mathbb{Z}_k^n : x_i, x_j = 0\}.$$

Thus $\mathbb{Z}_k^{ij}$ is isomorphic to $\mathbb{Z}_k^2$, and $\hat{\mathbb{Z}}_k^{ij}$ is isomorphic to $\mathbb{Z}_k^{n-2}$.

Let f be a system on $\mathbb{Z}_k^n$. For $1 \leq i < j \leq n$ and $x \in \mathbb{Z}_k^{ij}$, the ij -section of f at x is the system $f_{ij|x}$ on $\mathbb{Z}_k^{ij}$ given by

$$f_{ij|x}(y) = f(x + y), \quad y \in \mathbb{Z}_k^{ij}.$$

Note that $w(f) = \sum_{x \in \mathbb{Z}_k^n} w(f_{ij|x})$.

We wish to define a symmetrisation operator S_{ij} which acts on a system by replacing each ij -section by a ball of the same weight. More precisely, for any system f on $\mathbb{Z}_k^n$ and $1 \leq i < j \leq n$, we define a system $S_{ij}(f)$ on $\mathbb{Z}_k^n$, the ij -symmetrisation of f , by giving its ij -sections: if $x \in \mathbb{Z}_k^{ij}$ then

$$S_{ij}(f)_{ij|x} = b^{w_0},$$

where $w_0 = w(f_{ij|x})$. Thus $w(S_{ij}(f)) = w(f)$. We say that f is ij -symmetrised if $S_{ij}(f) = f$.

LEMMA 3. *Let f be a system on $\mathbb{Z}_k^n$, where k is even, and let $1 \leq i < j \leq n$. Then $w(\partial S_{ij}(f)) \leq w(\partial f)$.*

Proof. Write f' for $S_{ij}(f)$. To obtain $w(\partial f') \leq w(\partial f)$, it is sufficient to show that $w((\partial f')_{ij|x}) \leq w((\partial f)_{ij|x})$ for every $x \in \mathbb{Z}_k^{ij}$.

Then fix an arbitrary $x \in \mathbb{Z}_k^{ij}$. We have

$$(\partial f)_{ij|x} = \partial(f_{ij|x}) \vee \bigvee \{f_{ij|y} : y \in \mathbb{Z}_k^{ij}, d(y, x) = 1\},$$

and a similar relation holds with f' in place of f .

Now, the system $\partial(f'_{ij|x})$ and the systems $f'_{ij|y}$ for $y \in \mathbb{Z}_k^{ij}$, $d(y, x) = 1$ are nested. Moreover, $w(f'_{ij|y}) = w(f_{ij|y})$ for all y , while Lemma 2 tells us that $w(\partial(f'_{ij|x})) \leq w(\partial(f_{ij|x}))$. It follows that $w((\partial f')_{ij|x}) \leq w((\partial f)_{ij|x})$, as required. $\square$

We are now ready to prove the isoperimetric inequality in the discrete torus.

THEOREM 4. *Let $k \geq 2$ be even, and let f be a system on $\mathbb{Z}_k^n$ of weight v . Then $w(\partial f) \geq w(\partial b^v)$.*

Proof. For $n = 1$ and 2 the assertion was proved in Lemmas 1 and 2, so we may assume that $n \geq 3$. In this case it is easy to see that a system which is ij -symmetrised for all i and j must be a ball. Thus it is sufficient for us to find a system g on $\mathbb{Z}_k^n$ which satisfies $w(g) = w(f)$ and $w(\partial g) \leq w(\partial f)$, and which is ij -symmetrised for all i and j . This will be accomplished by a simple compactness argument.

Let $G = \{g \in [0, 1]^{\mathbb{Z}_k^n} : w(g) = w(f) \text{ and } w(\partial g) \leq w(\partial f)\}$. Then G is a (nonempty) compact subset of the product space $[0, 1]^{\mathbb{Z}_k^n}$. Furthermore, from Lemma 3 we know that if $g \in G$ then $S_{ij}(g) \in G$ for all i and j .

For $0 \leq r \leq nk/2$, write $w^{(r)}(g)$ for $\sum_{d(x,0)=r} g(x)$, and let H be the subset of G obtained by maximising successively $w^{(0)}(g)$, $w^{(1)}(g)$, $\dots$, $w^{(nk/2)}(g)$ on G . Clearly, $g \in H$ implies $S_{ij}(g) \in H$ for all i and j . Moreover, if $g \in H$ then each section $g_{ij|x}$ is such that for some r we have $g_{ij|x}(y) = 1$ if $d(y, 0) < r$ and $g_{ij|x}(y) = 0$ if $d(y, 0) > r$.

Now choose $g \in H$ with $w(g^{1/2})$ maximal. We claim that this g is ij -symmetrised for all i and j . Indeed, fix $1 \leq i < j \leq n$, and write g' for $S_{ij}(g)$.

For $x \in \mathbb{Z}_k^{\hat{ij}}$, there is an r such that $g_{ij|x}(y) = 1$ if $d(y, 0) < r$ and $g_{ij|x}(y) = 0$ if $d(y, 0) > r$. Let 1 denote the constant function 1 on $\mathbb{Z}_k^{\hat{ij}}$. Then, by the Cauchy–Schwarz inequality, we have

$$w^{(r)}(g_{ij|x}^{1/2}) \leq w^{(r)}(1)^{1/2} w^{(r)}(g_{ij|x})^{1/2},$$

with equality if and only if $g_{ij|x}$ is constant on $\{y \in \mathbb{Z}_k^{\hat{ij}} : d(y, 0) = r\}$, i.e., if and only if $g_{ij|x} = g'_{ij|x}$.

Since $w^{(r)}(g'_{ij|x}) = w^{(r)}(g_{ij|x})$, it follows that $w^{(r)}(g_{ij|x}^{1/2}) \leq w^{(r)}(g'_{ij|x}^{1/2})$, with equality if and only if $g_{ij|x} = g'_{ij|x}$. Thus $w(g_{ij|x}^{1/2}) \leq w(g'_{ij|x}^{1/2})$, with equality if and only if $g_{ij|x} = g'_{ij|x}$. So, by the choice of g , we must have $g_{ij|x} = g'_{ij|x}$. Since x was arbitrary, it follows that $g = g'$, as required. $\square$

In particular, we have the following result for (nonfractional) subsets of the torus.

COROLLARY 5. *Let k be even and let A be a subset of $\mathbb{Z}_k^n$ with*

$$|A| = |\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r\}| + \alpha |\{x \in \mathbb{Z}_k^n : d(x, 0) = r+1\}|,$$

where $0 \leq \alpha < 1$. Then

$$|\partial A| \geq |\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r+1\}| + \alpha |\{x \in \mathbb{Z}_k^n : d(x, 0) = r+2\}|.$$

From this we deduce immediately the corresponding result about t -boundaries.

COROLLARY 6. *Let k be even and let $A \subset \mathbb{Z}_k^n$ with $|A| \geq |\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r\}|$. Then for any $t = 0, 1, \dots$ we have $|A_{(t)}| \geq |\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r+t\}|$.*

It is easy to see that Corollary 6 implies the following isoperimetric inequality in the infinite grid $\mathbb{Z}^n$, first proved by Wang and Wang [5].

COROLLARY 7. *Let $A \subset \mathbb{Z}^n$ with $|A| \geq |\{x \in \mathbb{Z}^n : d(x, 0) \leq r\}|$. Then for any $t = 0, 1, \dots$, we have $|A_{(t)}| \geq |\{x \in \mathbb{Z}^n : d(x, 0) \leq r+t\}|$.*

In conclusion, let us turn to the case when k is odd. It is easy to see that Theorem 4 is not true in this case. In fact, as we mentioned earlier, even Lemma 1 is false. Furthermore, the corresponding theorem for (nonfractional) subsets of $\mathbb{Z}_k^n$, namely Corollary 6, is false for $n \geq 3$. For example, take $A = \{x \in \mathbb{Z}_3^3 : d(x, 0) \geq 2\}$. Then $|A| > |\{x \in \mathbb{Z}_3^3 : d(x, 0) \leq 2\}|$, but $|\partial A| < |\{x \in \mathbb{Z}_3^3 : d(x, 0) \leq 3\}|$.

The reason why balls are not always best appears to be that the complement of a ball, in contrast to the case k even, is not (isomorphic to) a ball. This contrasts with the

fact that one would expect the complement of a set of minimal boundary to be itself a set of minimal boundary.

Although we do not have much basis for it, we believe that if $k \geq 3$ is odd then in $\mathbb{Z}_k^n$ balls or complements of balls have smallest boundary. More precisely, we propose the following conjecture.

CONJECTURE 8. *If $k \geq 3$ is odd and A is a subset of $\mathbb{Z}_k^n$ with*

$$|A| \geq \max(|\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r\}|, |\{x \in \mathbb{Z}_k^n : d(x, 0) \geq s\}|)$$

then

$$|\partial A| \geq \min(|\{x \in \mathbb{Z}_k^n : d(x, 0) \leq r+1\}|, |\{x \in \mathbb{Z}_k^n : d(x, 0) \geq s-1\}|).$$

Needless to say, for fractional systems we conjecture the analogue of Theorem 4, namely that if $k \geq 3$ is odd then in $\mathbb{Z}_k^n$ fractional balls or complements of fractional balls have smallest boundary.

REFERENCES

- [1] B. BOLLOBÁS, *The chromatic number of random graphs*, *Combinatorica*, 8 (1988), pp. 49–55.
- [2] B. BOLLOBÁS AND I. B. LEADER, *Isoperimetric inequalities and fractional set systems*, *J. Combin. Theory Ser. A*, to appear.
- [3] L. H. HARPER, *Optimal numberings and isoperimetric problems on graphs*, *J. Combin. Theory*, 1 (1966), pp. 385–393.
- [4] V. D. MILMAN AND G. SCHECHTMAN, *Asymptotic Theory of Finite Dimensional Normed Spaces*, *Lecture Notes in Mathematics*, 1200, Springer-Verlag, Berlin, New York, 1986.
- [5] D.-L. WANG AND P. WANG, *Discrete isoperimetric problems*, *SIAM J. Appl. Math.*, 32 (1977), pp. 860–870.
- [6] ———, *Extremal configurations on a discrete torus and a generalization of the generalized Macaulay theorem*, *SIAM J. Appl. Math.*, 33 (1977), pp. 55–59.

ON THE DISTRIBUTION OF LENGTHS OF EVOLUTIONARY TREES*

M. CARTER†, M. HENDY‡, D. PENNY‡, L. A. SZÉKELY§, AND N. C. WORMALD¶

Abstract. This paper presents the results of the authors' investigation of a combinatorial problem arising from the study of evolutionary trees. In graph theoretic terms it can be expressed as a problem of colouring vertices of a binary tree. For a given colouring of the pendant vertices of a binary tree there is a simple algorithm for assigning colours to internal vertices minimising the number of edges of the tree whose end vertices have differing colours. This minimal number is called the *length* of the tree. The question posed is: For given numbers of pendant vertices of assigned colours, how many trees of a particular length can be constructed on those vertices? This question is answered in two special cases. Answers to this problem are needed to establish the distribution of lengths of evolutionary trees, by which the significance of the maximum parsimony principle for selecting evolutionary trees can be judged.

Key words. evolutionary trees, rooted binary trees, maximum parsimony, generating functions, Lagrange inversion formula, minimal colouring

AMS(MOS) subject classifications. 05C30, 05C05, 92A05

Introduction. Biologists frequently express the outcome of comparative studies of sets of taxa or species in the form of a tree indicating a possible evolutionary relationship. The most discriminating trees are rooted binary trees with each internal vertex representing a hypothetical ancestor of the taxa beyond it. For a set of n taxa there are a large number

$$\left(\sim \sqrt{2} \left(\frac{2n-4}{e} \right)^{n-2} \text{ as } n \rightarrow \infty \right)$$

of possible trees from which one is selected. One common criterion for this selection is the maximum parsimony principle, which favours the tree representing the minimum number of evolutionary events (mutations, sequence changes, etc.). This number is referred to as the length of the tree. Protein or nucleotide sequence data have a sequence of characters representing each taxon. These sequences are aligned in rows. Within any column the differences between the characters of different sequences are counted as the evolutionary events. In particular, when these sequences are associated with the pendant vertices of a tree, the algorithm of Fitch [1] gives a means of deriving sequences for internal vertices, so that the number of changes between sequences of adjacent vertices over the entire tree is minimised. For small values of n (≤ 16), the branch and bound algorithm [4] can then be used to determine all the trees of minimal and near minimal length. The significance of the minimal length tree must be gauged against the distribution of lengths of all competing trees.

In an attempt to understand the distribution of the lengths of evolutionary trees derived from sequence data, we have attempted to find the numbers of trees that require a given number of mutations of the characters in a single column of the table of aligned sequences. In two special cases, we succeed in deriving these numbers as a function of the number of sequences n and the number of occurrences of each character. This paper presents the resulting formulae. The main tool used is Lagrange inversion of generating functions. Some of the computations were so complicated that we found it helpful to

* Received by the editors April 11, 1988; accepted for publication (in revised form) March 31, 1989.

† Department of Mathematics and Statistics, Massey University, Palmerston North, New Zealand.

‡ Department of Botany and Zoology, Massey University, Palmerston North, New Zealand.

§ Eötvös Loránd Tudomány Egyetem Matematikai Intézet, Museum krt 6-8, Budapest, 1088 Hungary.

¶ Department of Mathematics and Statistics, University of Auckland, Private Bag, Auckland, New Zealand.

use the algebraic manipulation package REDUCE on the IBM 4341 at the University of Auckland.

The distribution of lengths of two distinct columns of data, however, are not independent, as can be seen in small examples. Thus, these formulae do not lead directly to answers for the broader questions, although they give some insight into the distribution of tree lengths over the whole data set.

Binary trees with labeled vertices. A *pendant* vertex is a vertex of degree 1. A *binary* tree with $n \geq 2$ labelled pendant vertices is a tree with exactly n pendant vertices, which are labelled $1, 2, \dots, n$, at most one vertex of degree 2, and the remaining vertices of degree 3. For completeness, when $n = 1$ we define an isolated vertex to be a binary tree and regard it as a pendant vertex. If the tree has a vertex of degree 2 (or 0, when $n = 1$) this vertex is called the *root* and the tree is *rooted*. Let $\mathbf{R}_n$ be the set of all such rooted trees and $\mathbf{U}_n$ be the set of all such unrooted trees. The following well-known results are easily derived.

LEMMA 1.

- (a) $|\mathbf{R}_n| = (2n - 3)!!$;
- (b) $|\mathbf{U}_n| = (2n - 5)!!$;
- (c) $T \in \mathbf{R}_n \cup \mathbf{U}_n$ has $n - 2$ vertices of degree 3;
- (d) $T \in \mathbf{R}_n$ has $2n - 2$ edges;
- (e) $T \in \mathbf{U}_n$ has $2n - 3$ edges.

Recall that for any positive integer m the double factorial can be defined recursively as $m!! = m \times (m - 2)!!$, with $0!! = (-1)!! = 1$.

Colouring the vertices. A convenient way of associating the nucleotide values to the vertices of a tree is to consider them as vertex colourings. Let $\mathbf{C} = \{A, B, C, \dots\}$ be a set of r colours. For $T \in \mathbf{U}_n$ a *colouring* of T is an assignment of colours to the vertices of T . If T is coloured then those edges whose end vertices have different colours are called *changes*. For a given assignment of colours to the pendant vertices of T , a *minimal colouring* of T is a colouring in which the number of changes is minimal of those colourings that agree with that assignment on the pendant vertices.

Let $f_m(a, b, c, \dots)$ be the number of trees in $\mathbf{U}_n$ that have a pendant vertices coloured A , b pendant vertices coloured B , and so on ($n = a + b + c + \dots$), and that have a minimal colouring with m changes. Also, let $N(n, m)$ denote $(2n - m - 1)! / (n - m)!(m - 1)!2^{n-m}$ if $n \geq m$ and 0 if $n < m$. We give a combinatorial meaning for $N(n, m)$ later.

We begin with two preliminary results on coefficients of power series. Here we use square brackets $[]$ to denote the extraction of coefficients. Thus $[x^i]g(x)$ is the coefficient of x^i in $g(x)$.

LEMMA 2.

$$[x^i] \left(1 - \frac{x}{2}\right)^{-(i+1-p)} (1-x)^{-j} = [t^i] \left(\frac{1 + \sqrt{1-2t}}{2}\right)^p (1-2t)^{-(j+1)/2}.$$

Proof. Let

$$\varphi(\lambda) = (1 - \tfrac{1}{2}\lambda)^{-1}, \quad f(\lambda) = (1 - \tfrac{1}{2}\lambda)^p (1 - \lambda)^{-(j+1)}$$

and

$$w(t) = t(1 - \tfrac{1}{2}w(t))^{-1}.$$

Then we have

$$w(t) = 1 - \sqrt{1-2t}$$

and

$$f(w(t)) = \left(\frac{1 + \sqrt{1-2t}}{2} \right)^p (1-2t)^{-(j+1)/2}.$$

By Lagrange's inversion formula (see Goulden and Jackson [2, Thm. 1.2.9, Part 1]) we have

$$[t^i] f(w(t)) = [\lambda^i] f(\lambda) \varphi^i(\lambda) (1 - \lambda \varphi'(\lambda) / \varphi(\lambda))$$

from which the lemma follows.

LEMMA 3. *Let*

$$h(x, y, z) = \sum_{p,q,r} h_{pqr} x^p y^q z^r$$

be a polynomial,

$$\varphi_1 = (1 - \lambda_1 \lambda_2 \lambda_3 - \tfrac{1}{2} \lambda_1)^{-1}, \quad \varphi_2 = (1 - \lambda_1 \lambda_2 \lambda_3 - \tfrac{1}{2} \lambda_2)^{-1}, \quad \varphi_3 = (1 - \tfrac{1}{2} \lambda_1 \lambda_2 \lambda_3)^{-1},$$

and $\delta = 0$ or 1 . Then

$$\begin{aligned} & [\lambda_1^{k_1 - \delta} \lambda_2^{k_2} \lambda_3^{k_3}] h(\varphi_1^{-1}, \varphi_2^{-1}, \varphi_3^{-1}) \varphi_1^{k_1+1} \varphi_2^{k_2+1} \varphi_3^{k_3+1} \\ &= 2^{-k_1 - k_2 + 3k_3 + \delta} \sum_{p,q,r} g_{pqr} \binom{2k_1 - k_3 - \delta - p}{k_1 - k_3 - \delta} \binom{2k_2 - k_3 - q}{k_2 - k_3} \binom{k_1 + k_2 + \frac{1}{2} - (p+q+r+\delta)/2}{k_3} \end{aligned}$$

where

$$\sum_{p,q,r} g_{pqr} x^p y^q z^r = h\left(x, y, \frac{1+z}{2}\right).$$

Proof. The coefficient desired is

$$[\lambda_1^{k_1 - k_3 - \delta} \lambda_2^{k_2 - k_3} \rho^{k_3}] \sum h_{pqr} \hat{\varphi}_1^{k_1+1-p} \hat{\varphi}_2^{k_2+1-q} \hat{\varphi}_3^{k_3+1-r}$$

where

$$\hat{\varphi}_1 = (1 - \rho - \tfrac{1}{2} \lambda_1)^{-1}, \quad \hat{\varphi}_2 = (1 - \rho - \tfrac{1}{2} \lambda_2)^{-1}, \quad \hat{\varphi}_3 = (1 - \tfrac{1}{2} \rho)^{-1}.$$

As

$$\hat{\varphi}_i = (1 - \rho) \left(1 - \frac{\frac{1}{2} \lambda_i}{1 - \rho} \right)$$

for $i = 1$ and 2 , this is

$$\begin{aligned} & [\lambda_1^{k_1 - k_3 - \delta} \lambda_2^{k_2 - k_3} \rho^{k_3}] \sum_{p,q,r} h_{pqr} (1 - \tfrac{1}{2} \rho)^{-(k_3+1-r)} \\ & \quad \times (1 - \rho)^{-k_1 - k_2 - 2 + p + q} \left(\frac{\frac{1}{2}}{1 - \rho} \right)^{k_1 + k_2 - 2k_3 - \delta} |B_1 B_2| \end{aligned}$$

where

$$\begin{aligned} B_1 &= \binom{-(k_1+1-p)}{k_1 - k_3 - \delta} \text{ and } B_2 = \binom{-(k_2+1-q)}{k_2 - k_3}, \\ &= \sum_{p,q} |B_1 B_2| 2^{-(k_1 + k_2 - 2k_3 - \delta)} \sum_r h_{pqr} [\rho^{k_3}] (1 - \tfrac{1}{2} \rho)^{-(k_3+1-r)} (1 - \rho)^{-v} \end{aligned}$$

where $v = 2(k_1 + k_2 - k_3 + 1) - p - q - \delta$.

The sum over r here is, by Lemma 2,

$$\begin{aligned} \sum_r h_{pqr}[t^{k_3}] \left(\frac{1 + \sqrt{1-2t}}{2} \right)^r (1-2t)^{-(r+1)/2} &= \sum_r g_{pqr}[t^{k_3}] (1-2t)^{-(v+1-r)/2} \\ &= \sum_r g_{pqr} \binom{-(v+1-r)/2}{k_3} (-2)^{k_3}. \end{aligned}$$

The lemma follows.

We now give the first of our two main results.

THEOREM 1. *For all $m \geq 1$,*

$$f_m(a, b) = (m-1)!(2n-3m)(2n-5)!! N(a, m)N(b, m)/(2n-2m-1)!!.$$

Proof. We consider three classes of rooted binary trees whose vertices of degree 1 (or 0) have been 2-coloured using colours A and B :

- T_1 : those trees in which every minimal colouring assigns the root vertex the colour A ;
- T_2 : those trees in which every minimal colouring assigns the root vertex the colour B ;
- T_3 : all remaining trees.

We define the generating functions T_1 , T_2 , and T_3 for T_1 , T_2 , and T_3 by

$$T_i = \sum_{k_1, k_2, k_3 \geq 0} n_{i, k_1, k_2, k_3} x^{k_1} y^{k_2} z^{k_3} / k_1! k_2!$$

where n_{i, k_1, k_2, k_3} is the number of trees in T_i with k_1 pendant vertices coloured A , k_2 coloured B and with k_3 changes in a minimal colouring. In this count, vertices of degree 0 are regarded as pendant vertices. The generating functions are exponential in x and y because the pendant vertices are labeled and ordinary in z because the edges are not.

Now let G be any tree in $T_1 \cup T_2 \cup T_3$ with more than one vertex. Assume that all trees are embedded in the plane, so that when the root vertex r is deleted from G there remain a *left branch* G_1 and a *right branch* G_2 (with roots r_1, r_2 , respectively), which are both in $T_1 \cup T_2 \cup T_3$. Denote the number of changes in minimal colourings of G, G_1, G_2 , by p, p_1, p_2 , respectively. There are six cases to consider.

(i) $G_1, G_2 \in T_1$. Let C be any colouring of G . If any of r, r_1 , or r_2 are coloured B by C , then replacing the colouring of the affected branches by a minimal colouring (in which r_1 or r_2 , as the case may be, must be coloured A), and (if necessary) changing the colour of r to A , gives a colouring of G with fewer changes than C . Thus any minimal colouring of G must assign the colour A to r, r_1 , and r_2 and must be minimal for G_1 and G_2 . Hence $G \in T_1$ and $p = p_1 + p_2$.

(ii) $G_1, G_2 \in T_2$. A similar argument shows that $G \in T_2$ and $p = p_1 + p_2$.

(iii) $G_1 \in T_1, G_2 \in T_3$ or vice versa. The argument is similar to (i), because the branch belonging to T_3 has at least one minimal colouring in which its root is coloured A . Again $G \in T_1$ and $p = p_1 + p_2$.

(iv) $G_1 \in T_2, G_2 \in T_3$ or vice versa. The argument is likewise similar to (ii); again $G \in T_2$ and $p = p_1 + p_2$.

(v) $G_1, G_2 \in T_3$. If C is a colouring of G that assigns different colours to r_1 and r_2 , C can be replaced by a colouring that is minimal for G_1 and G_2 and that assigns the same colour to r_1 and r_2 ; giving r this colour as well yields a colouring with fewer changes than C . Hence, any minimal colouring of G assigns r_1, r_2 , and r the same colour; evidently, there exist minimal colourings in which this colour is A and in which it is B . Thus $G \in T_3$ and $p = p_1 + p_2$.

(vi) $G_1 \in \mathbf{T}_1$, $G_2 \in \mathbf{T}_2$ or vice versa. We may assume $G_1 \in \mathbf{T}_1$. Suppose C is a minimal colouring of G that assigns the same colour to both r_1 and r_2 ; without loss of generality, let this colour be A . Since C is minimal, r must also have colour A and C must be minimal for G_1 . Replacing C by a colouring that is minimal on G_2 (so that r_2 is coloured B) reduces the number of changes in G_2 by at least one and introduces exactly one new change on an edge incident with r . But C is minimal, so the new colouring must in fact be minimal and reduce the number of changes in G_2 by exactly one. Thus $p = p_1 + p_2 + 1$ in this case, and obviously the same holds if all minimal colourings of G assign different colours to r_1 and r_2 . It follows that G must have a minimal colouring in which r_1 and r_2 are assigned different colours, and then r can be colored either A or B while preserving minimality. Thus $G \in \mathbf{T}_3$ and $p = p_1 + p_2 + 1$.

Hence, according to whether G is in $\mathbf{T}_1$, $\mathbf{T}_2$, or $\mathbf{T}_3$, either cases (i) and (iii), cases (ii) and (iv), or cases (v) and (vi) must apply. Since the distinction between left and right branches of labelled rooted trees with more than one vertex counts each such tree twice, we obtain the following relations among the generating functions (remembering that x and y are the generating functions for the trees with just one vertex in $\mathbf{T}_1$ and $\mathbf{T}_2$, respectively):

$$2(T_1 - x) = T_1^2 + T_1 T_3 + T_3 T_1$$

$$2(T_2 - y) = T_2^2 + T_2 T_3 + T_3 T_2$$

$$2T_3 = T_3^2 + zT_1 T_2 + zT_2 T_1$$

where the multiplier z in the last relation reflects the fact that in case (vi) the minimum number of changes in G is one more than the sum of the minimum number of changes in G_1 and G_2 . Hence

$$(1) \quad T_1 = T_1 T_3 + \frac{1}{2} T_1^2 + x$$

$$(2) \quad T_2 = T_2 T_3 + \frac{1}{2} T_2^2 + y$$

$$(3) \quad T_3 = \frac{1}{2} T_3^2 + zT_1 T_2.$$

Put

$$(4) \quad V = T_3 / T_1 T_2.$$

Then with $\lambda_1 = T_1$, $\lambda_2 = T_2$, $\lambda_3 = V$ we have

$$(5) \quad \lambda_1 = x\varphi_1$$

$$(6) \quad \lambda_2 = y\varphi_2$$

$$(7) \quad \lambda_3 = z\varphi_3$$

where

$$(8) \quad \begin{aligned} \varphi_1 &= (1 - \tau - \frac{1}{2}\lambda_1)^{-1} \\ \varphi_2 &= (1 - \tau - \frac{1}{2}\lambda_2)^{-1}, \\ \varphi_3 &= (1 - \frac{1}{2}\tau)^{-1} \end{aligned}$$

and τ denotes $\lambda_1 \lambda_2 \lambda_3$.

We will be interested in the expansion of $F(T_1, T_2, V)$, for various functions F (in particular, when $F(T_1, T_2, V)$ is $T_3 = VT_1 T_2$, or T_1 , or $zT_2 = VT_2(1 - \frac{1}{2}T_1 T_2 V)$) in

powers of x, y, z . By the multivariate Lagrange theorem (see [2, Thm. 1.2.9, Part 1]),

$$(9) \quad [x^{k_1} y^{k_2} z^{k_3}] F(T_1, T_2, V) = [\lambda_1^{k_1} \lambda_2^{k_2} \lambda_3^{k_3}] F(\underline{\lambda}) \varphi_1^{k_1} \varphi_2^{k_2} \varphi_3^{k_3} \left\| \delta_{ij} - \frac{\lambda_j}{\varphi_i} \left(\frac{\partial \varphi_i}{\partial \lambda_j} \right) \right\|.$$

As is usual, we denote $(\lambda_1, \lambda_2, \lambda_3)$ by $\underline{\lambda}$. The determinant here is seen to be

$$\varphi_1 \varphi_2 \varphi_3 \begin{vmatrix} 1 - 2\tau - \lambda_1 & -\tau - \frac{1}{2}\lambda_1 & -\tau \\ -\tau - \frac{1}{2}\lambda_2 & 1 - 2\tau - \lambda_2 & -\tau \\ -\frac{1}{2}\tau & -\frac{1}{2}\tau & 1 - \tau \end{vmatrix} = \varphi_1 \varphi_2 \varphi_3 D$$

where

$$D = \begin{vmatrix} \frac{2}{\varphi_1} - 1 & \frac{1}{\varphi_1} - 1 & \frac{2}{\varphi_3} - 2 \\ \frac{1}{\varphi_2} - 1 & \frac{2}{\varphi_2} - 1 & \frac{2}{\varphi_3} - 2 \\ \frac{1}{\varphi_3} - 1 & \frac{1}{\varphi_3} - 1 & \frac{2}{\varphi_3} - 1 \end{vmatrix}.$$

Thus (9) becomes

$$(10) \quad [x^{k_1} y^{k_2} z^{k_3}] F(T_1, T_2, V) = [\lambda_1^{k_1} \lambda_2^{k_2} \lambda_3^{k_3}] F(\underline{\lambda}) D \varphi_1^{k_1+1} \varphi_2^{k_2+1} \varphi_3^{k_3+1}.$$

To complete the proof, we note that $af_m(a, b)$ is the number of unrooted binary trees whose vertices of degree 1 have been coloured, a of colour A and b of colour B , with m colour changes, and with one of the vertices coloured A distinguished. The generating function for these numbers will be

$$x(T_1 + zT_2 + T_3),$$

as shown by counting the result of removing the distinguished vertex. That is,

$$(11) \quad af_m(a, b) = a!b! [x^{a-1} y^b z^m] (T_1 + zT_2 + T_3).$$

Hence, we only need to find the coefficients of T_1 , zT_2 , and T_3 . For this, we use (10) and Lemma 3. The rest of the proof is merely a matter of straightforward algebraic manipulations, but their complexity is such that the provision of a description seems advisable.

We first treat T_1 . To obtain its coefficients, set $F(T_1, T_2, V) = T_1$ in (10). Thus $F(\underline{\lambda}) = \lambda_1$, and so in Lemma 3 we wish to set $\delta = 1$ and

$$h(\varphi_1^{-1}, \varphi_2^{-1}, \varphi_3^{-1}) = D.$$

Let $\hat{D}$ denote the result of replacing φ_1^{-1} by x , φ_2^{-1} by y and φ_3^{-1} by $\frac{1}{2}(1+z)$ in D . Then

$$h(x, y, (1+z)/2) = \hat{D} = z(4xy + 1) - (x+y)(1+z^2).$$

We find g_{pqr} by setting

$$\sum_{p,q,r} g_{pqr} x^p y^q z^r = \hat{D}.$$

By Lemma 3,

$$(12) \quad [x^{k_1} y^{k_2} z^{k_3}] T_1 = 2^{-k_1 - k_2 + 3k_3 + 1} \sum_{p,q,r} g_{pqr} \binom{C_1 - p}{C_2} \binom{C_3 - q}{C_4} \binom{C_5 - s}{C_6}$$

where

$$s = \frac{1}{2}(p + q + r - 1),$$

and

$$C_1 = 2k_1 - k_3 - 1, \quad C_2 = k_1 - k_3 - 1, \quad C_3 = 2k_2 - k_3,$$

$$C_4 = k_2 - k_3, \quad C_5 = \frac{1}{2}(2k_1 + 2k_2 - 1), \quad C_6 = k_3.$$

We now assume $C_1, C_3, C_5 > 0$ and $C_2 \geq 0$. The other cases are very special and will be treated later, since the algebraic manipulations used to simplify the general case do not apply to them.

Note that

$$\binom{t_1 - p}{t_2} = \frac{1}{t_1} \binom{t_1}{t_2} \times \begin{cases} t_1 & p = 0 \\ t_1 - t_2 & p = 1 \end{cases},$$

provided $p = 0$ or 1 and $t_1 > 0$. Hence (12) gives

$$(13) \quad [x^{k_1} y^{k_2} z^{k_3}] T_1 = 2Q(C_1, C_2, C_3, C_4, C_5, C_6)$$

where $Q(t_1, t_2, t_3, t_4, t_5, t_6)$ denotes

$$(14) \quad \frac{2^{-k_1 - k_2 + 3k_3}}{t_1 t_3 t_5} \binom{t_1}{t_2} \binom{t_3}{t_4} \binom{t_5}{t_6} \sum_{p,q,r} g_{pqr} H(t_1, t_2, p) H(t_3, t_4, q) H(t_5, t_6, s)$$

and where

$$H(i, j, k) = \begin{cases} i & k = 0 \\ i - j & k = 1 \end{cases}.$$

Calculations (here we began to use REDUCE) show that

$$(15) \quad \sum_{p,q,r} g_{pqr} H(C_1, C_2, p) H(C_3, C_4, q) H(C_5, C_6, s) = (2k_1 - k_3 - 1)/2.$$

This gives us a simple formula for the coefficients of T_1 .

Next, consider $F(T_1, T_2, V) = zT_2 = VT_2(1 - \frac{1}{2}T_1T_2V)$. Instead of working with (10), we observe, as interchanging the colours A and B shows,

$$n_{2,k_1,k_2,k_3} = n_{1,k_2,k_1,k_3}.$$

Hence, we obtain a formula for

$$[x^{k_1} y^{k_2} z^{k_3}] zT_2$$

by taking the right-hand side of (13) and replacing k_3 by $k_3 - 1$ and interchanging k_1 and k_2 . Thus

$$(16) \quad [x^{k_1} y^{k_2} z^{k_3}] zT_2 = \frac{1}{4}Q(C_3, C_4, C_1 + 2, C_2 + 2, C_5, C_6 - 1),$$

and we find that

$$(17) \quad \sum_{p,q,r} g_{pqr} H(C_3, C_4, p) H(C_1 + 2, C_2 + 2, q) H(C_5, C_6 - 1, s) = (k_1 - 1)(2k_3 - k_1)/2.$$

Finally, consider $F(T_1, T_2, V) = T_3 = VT_1T_2$, to be used in (10). Then $F(\lambda) = \lambda_1\lambda_2\lambda_3 = 2(1 - \varphi_3^{-1})$, and so in Lemma 3 we set $\delta = 0$ and $h(\varphi_1^{-1}, \varphi_2^{-1}, \varphi_3^{-1}) = 2(1 - \varphi_3^{-1})D$. Then following the same calculations as for T_1 , we get

$$h(x, y, (1 + z)/2) = (1 - z)\hat{D},$$

with $\tilde{D}$ defined as before. With g , s , Q and H defined as before, Lemma 3 gives

$$(18) \quad [x^{k_1} y^{k_2} z^{k_3}] T_3 = Q(C_1 + 1, C_2 + 1, C_3, C_4, C_5 + \tfrac{1}{2}, C_6) \\ - Q(C_1 + 1, C_2 + 1, C_3, C_4, C_5, C_6).$$

Calculations show that

$$(19) \quad \sum_{p,q,r} g_{pqr} H(C_1 + 1, C_2 + 1, p) H(C_3, C_4, q) H(C_5 + \tfrac{1}{2}, C_6, s) = 0,$$

and an alternative way to see this is to consider $F = 1$ in (10) and compute the coefficients of F . Also

$$(20) \quad \sum_{p,q,r} g_{pqr} H(C_1 + 1, C_2 + 1, p) H(C_3, C_4, q) H(C_5, C_6, s) = -\tfrac{1}{2} k_3^2.$$

From (13) to (20), after more calculations, we find

$$(21) \quad [x^{k_1} y^{k_2} z^{k_3}] (T_1 + zT_2 + T_3) \\ = 2^{-k_1 - k_2 + 3k_3} \binom{C_1}{C_2} \binom{C_3}{C_4} \binom{C_5}{C_6} \frac{k_3(2k_1 - k_3)(2k_1 - k_3 + 1)(2k_1 + 2k_2 - 3k_3 + 2)C_1}{C_3 C_5 (C_2 + 1)(C_2 + 2)(C_5 - C_6 + 1)}.$$

The theorem follows from (11) and (21), after a little algebra, provided the values of m , a and b are such that (21) is valid; i.e., $C_1, C_3, C_5 > 0$ and $C_2 \geq 0$, where $k_1 = a - 1$, $k_2 = b$, and $k_3 = m$.

But since $m \geq 1$, we must have $a \geq 1$ and $b \geq 1$, and so $C_5 > 0$. It can also be seen that $C_3 > 0$ whenever the coefficient being computed is nonzero, since $k_3 \leq k_2$. For a similar reason, $C_2 \geq 0$ unless $k_1 = k_3$, but in this case we re-express

$$\binom{C_1}{C_2} \text{ as } \binom{C_1 + 1}{C_2 + 1} \frac{C_2 + 1}{C_1 + 1},$$

and then our algebraic computations become valid, with the same final result. Finally, if $C_1 \leq 0$ then again, since $k_1 \geq k_3$, we have $k_1 = k_3 = 1$. In this case there appears to be no convenient way to reexpress the expressions affected, but as there is no contribution from T_1 , it is a simple matter to work with (16) and (18) to verify the theorem. The proof is now complete.

We observe that equations (13)–(20) give quite simple formulae for the coefficients of T_1 , T_2 , and T_3 . Also, it is feasible that the result can be derived more directly due to the combinatorial significance of the numbers $N(a, m)$ and $N(b, m)$, so it is appropriate to mention the following.

LEMMA 4. $N(n, m)$ is the number of forests of exactly m binary rooted trees with exactly n pendant vertices in total, labeled $1, 2, \dots, n$.

Proof. For this lemma, we define the exponential generating function for a class of forests (or trees) to be $\sum g_n x^n / n!$, where g_n is the number of forests in the class with n pendant vertices. The exponential generating function for binary rooted trees with labelled pendant vertices is $1 - \sqrt{1 - 2x}$ by Lemma 1(a) and the binomial theorem. Hence the exponential generating function for the forests in question is $(1 - \sqrt{1 - 2x})^m$ if the m component trees are ordered (see Harary and Palmer [3, p. 8]). To unordered the m trees, we must divide by $m!$ (this is valid since each component has at least one label). Hence

$$N(n, m) = n! [x^n] (1 - \sqrt{1 - 2x})^m / m! \\ = (n! / m!) [x^n] w^m$$

where $w = 2x / (2 - w)$. The result now follows from Lagrange's theorem.

For $r > 2$ colours little progress has been made, except for one general case. This is for the number of trees $f_{r-1}(a_1, a_2, \dots, a_r)$ with vertices colored from the set $C = \{A_1, A_2, \dots, A_r\}$ of r colours in which the number of minimal changes is $r - 1$. This number of changes is the least possible for any tree on r colours.

THEOREM 2. For all $r \geq 2$,

$$f_{r-1}(a_1, \dots, a_r) = (2n-5)!! N(a_1, 1) \cdots N(a_r, 1) / (2n-2r-1)!!.$$

Proof. Begin with a labelled tree T with $r+k$ vertices, $1, \dots, r$ of degrees $d_1, \dots, d_r$ and $r+1, \dots, r+k$ of degree 3. The number of such T is (see Moon [5])

$$\frac{(r+k-2)!}{\prod (d_i-1)! 2^k}.$$

Replace each vertex d_i of T by a binary tree T_i with a_i labelled pendant vertices, coloured A_i , and choose d_i positions of attachment for the edges meeting T_i in T . Finally, “unlabel” the vertices $r+1, \dots, r+k$ in T . All this produces one of the trees counted by $f_{r-1}(a_1, \dots, a_r)$, and so, by Lemma 1(b),

$$f_{r-1}(a_1, \dots, a_r) = \sum_{k \geq 0} \frac{1}{k!} \sum_{\substack{d_1, \dots, d_r \\ \sum (d_i-1) = r-2-k}} \frac{(r+k-2)!}{\prod (d_i-1)! 2^k} \prod_{i=1}^r (2a_i-5)!! \frac{(2a_i-4+d_i)!}{(2a_i-4)!}$$

provided $a_i \geq 2$ for each i . Note that $k = r-2 - \sum (d_i-1)$ is a necessary restriction due to the degrees of the vertices in T .

The expression above is

$$(\prod N(a_i, 1))(r-2)! [x^{r-2}] \left(\sum_{k=0}^{\infty} \binom{r-2+k}{k} \frac{1}{2^k} x^k \prod_{i=1}^r \sum_{c=0}^{\infty} \frac{(2a_i-3+c)!}{c!(2a_i-3)!} x^c \right)$$

(and now the expression is valid even if some of the a_i are 1)

$$\begin{aligned} &= (\prod N(a_i, 1))(r-2)! [x^{r-2}] \left(\frac{1}{1-\frac{1}{2}x} \right)^{r-1} \prod_{i=1}^r (1-x)^{-2(a_i-1)} \\ &= (\prod N(a_i, 1))(r-2)! [x^{r-2}] (1-\frac{1}{2}x)^{-(r-1)} (1-x)^{-(2n-2r)} \\ &= (\prod N(a_i, 1))(r-2)! [t^{r-2}] (1-2t)^{-(2n-2r+1)/2} \end{aligned}$$

by Lemma 2, from which the theorem follows.

Discussion. For small values of n , ($n < 16$) it is possible by generating all $(2n-5)!!$ trees to actually compute $f_m(a_1, a_2, \dots, a_r)$ for all values of m , and for larger n , estimates of $f_m(a_1, a_2, \dots, a_r)$ can be obtained using Monte-Carlo sampling of a suitable population of binary trees. For applications we have needed to use these approximations to estimate values outside those derived above. The simple extension

$$f_m(1, a_2, a_3, \dots, a_r) = (2n-5) f_{m-1}(a_2, a_3, \dots, a_r)$$

has been useful in many smaller cases.

Open problem. Is there a more direct (structural?) proof of Theorem 1? Such a proof would hopefully make direct use of the occurrence of the factors $N(a, m)$ and $N(b, m)$ and possibly lead to more general results such as an explicit expression for

$f_m(a_1, a_2, \dots, a_r)$. It is known that $f_m(a_1, a_2, \dots, a_r)$ is not expressible as a simple product in the form of Theorems 1 and 2.

REFERENCES

- [1] W. M. FITCH, *Toward defining the course of evolution: minimum change on a specific tree topology*, Syst. Zool., 20 (1971), pp. 406–416.
- [2] I. P. GOULDEN AND D. M. JACKSON, *Combinatorial Enumeration*, John Wiley, New York, 1983.
- [3] F. HARARY AND E. M. PALMER, *Graphical Enumeration*, Academic Press, New York, 1973.
- [4] M. D. HENDY AND D. PENNY, *Branch and bound algorithms to determine evolutionary trees*, Math. Biosci., 59 (1982), pp. 277–290.
- [5] J. W. MOON, *Counting labelled trees*, Canadian Mathematical Monographs, No. 1, Canad. Math. Congress, Montreal, 1970.

A FAST PERFECT-MATCHING ALGORITHM IN RANDOM GRAPHS*

OLIVIER GOLDSCHMIDT† AND DORIT S. HOCHBAUM‡

Abstract. The matching problem is to find a maximum collection of mutually nonadjacent edges in a graph. An algorithm is presented that delivers a perfect matching in a random graph almost surely. The expected running time of this algorithm is $O(n \log_e(1/p) + n)$, where n is the number of vertices in the graph and where p is the probability of an edge. This running time is faster than $O(n \log_e n)$, the expected running time of the best randomized algorithm of Angluin and Valiant.

Key words. matching, random graphs

AMS(MOS) subject classification. 05C80

1. Introduction. Given a simple undirected graph, $G = (V, E)$, $|V| = n$, the matching problem is to find a maximum number of edges, no two of which share a node. The matching is perfect if it contains $n/2$ edges (we assume n even in this paper). One of the fastest known deterministic algorithms for the matching problem [MV80] has running time $O(\sqrt{|V|}|E|)$.

In random graphs a perfect matching can be identified faster, taking into account the properties of a random graph. Angluin and Valiant devised an algorithm that runs in time $O(n \log_e n)$ [AV79]. The algorithm we propose is advantageous, for dense graphs, where it has linear running time, and in general it is faster and simpler than Angluin and Valiant's algorithm.

The theory of random graphs is concerned with graphs drawn from certain probability spaces. A commonly used probability space is $\mathcal{G}_{n,p}$, where n is a positive integer (n is even in this paper) and $0 \leq p \leq 1$. This probability space consists of all the graphs on vertex set $V = \{v_1, \dots, v_n\}$. In a graph drawn from $\mathcal{G}_{n,p}$, the probability that an edge is present is p independently for each edge.

When $p = c_n/n$ is a function of the number of nodes of the graph, Erdős and Rényi [ER60] showed that for $p = (\log_e n + x)/n$, the probability that a graph drawn from $\mathcal{G}_{n,p}$ has no isolated node is asymptotically (as $n \rightarrow \infty$) equal to $e^{-e^{-x}}$. Hence, if $p \geq (\log_e n + t_n)/n$, $\lim_{n \rightarrow \infty} t_n = \infty$, then almost every graph in $\mathcal{G}_{n,p}$ has no isolated vertex. No isolated vertex is an obvious necessary condition for a graph to have a perfect matching. In [ER66], Erdős and Rényi proved that for the same range of p (i.e., $p \geq (\log_e n + t_n)/n$), almost every graph in $\mathcal{G}_{n,p}$ has a perfect matching.

Angluin and Valiant's randomized algorithm almost surely delivers a perfect matching in a random graph drawn from $\mathcal{G}_{n,p}$, $p \geq c \log_e n/n$, for some constant $c > 1$. It constructs the matching by having the vertices in V pair up in stages. At each stage, the lowest-numbered free (or unmatched) vertex in V , v_* , pairs up with one of its neighbors, v , randomly chosen. This neighbor is removed from the list of neighbors of the vertex v_* . The vertex v breaks off any previous pairing with another vertex, if there are any.

* Received by the editors October 3, 1988; accepted for publication (in revised form) July 14, 1989.

† Mechanical Engineering Department, University of Texas, Austin, Texas 78712. This research was supported in part by the Intercollegiate Center for Management (Belgium) and by I.B.E.R. School of Business Administration, University of California, Berkeley.

‡ Industrial Engineering and Operations Department and School of Business Administration, University of California, Berkeley, California 94720. This research was supported in part by Office of Naval Research grant N00014-88-K-0377.

The expected running time of this algorithm is $O(n \log_e n)$ and is independent of the value of p .

In the extreme case of a complete graph, when $p = 1$, we can construct a matching in linear time by pairing nodes arbitrarily with any available neighbor (and there is always one available). Generally, for dense graphs, we can take advantage of the abundance of edges to construct a perfect matching faster. This is precisely the idea we exploit.

In this paper, we present a very simple matching algorithm that we call the greedy matching algorithm. The greedy matching attempts to match a vertex in V , v , with one of its yet unmatched neighbors. If there is such a neighbor, it deletes the two matched vertices from the graph whereas, in case of failure, the vertex v is recorded in a set of unmatched vertices. The adjective “greedy” comes from the fact that, once two vertices are matched, they remain matched in the output of the algorithm. This greedy matching algorithm has the feature that, even if the graph has a perfect matching, it does not necessarily deliver a perfect matching. Yet, a large fraction of the vertices get matched. This fraction depends on the graph density. Specifically, for a graph drawn from $\mathcal{G}_{n,p}$, the following claims are true:

1. If $\lim_{n \rightarrow \infty} p = 0$, then all but $O(1/p)$ vertices are matched almost surely.
2. If p is fixed, $0 < p < 1$, then all but t_n vertices are matched almost surely, where t_n is any function such that $\lim_{n \rightarrow \infty} t_n = \infty$.
3. If $\lim_{n \rightarrow \infty} p = 1$, then the greedy matching algorithm delivers a perfect matching almost surely.

We also show that the expected running time of this greedy matching algorithm is $O(n \log_e (1/p) + n)$.

In § 4, we devise an algorithm that delivers a perfect matching almost surely if $p \geq c' \log_e n/n$, $c' > 2c$, where c is the constant used in Angluin and Valiant’s paper. This algorithm is a combination of the greedy matching and of Angluin and Valiant’s algorithm. Its expected running time is $O(n \log_e (1/p) + n)$. This algorithm is faster than the Angluin and Valiant’s one for all values of $p \geq c' \log_e n/n$, and its speed increases with increasing values of p . In particular, it runs in linear time ($O(n)$) for all values of p larger or equal to a constant.

An adaptation of this algorithm also works for bipartite graphs. A graph drawn from probability space $\mathcal{B}_{n,n,p}$ is a bipartite graph on vertex set $S \cup T$, $|S| = |T| = n$. For each pair of vertices $s \in S$ and $t \in T$, a graph drawn from $\mathcal{B}_{n,n,p}$ has edge (s, t) with probability p . In [ER64], Erdős and Rényi proved that the threshold for a random bipartite graph to have a perfect matching is the same as for general graphs, namely $p \geq (\log_e n + t_n)/n$. Applied to a graph drawn from $\mathcal{B}_{n,n,p}$, $p \geq c \log_e n/n$, Angluin and Valiant’s algorithm also delivers a perfect matching in $O(n \log_e n)$ time. In § 5, we show how to adapt our algorithm for the case where the input graph is a random bipartite graph drawn from $\mathcal{B}_{n,n,p}$. We prove that our algorithm has the same range of application and the same expected running time as for the general case.

For $(c \log_e n)/n > p \geq (\log_e n + t_n)/n$, there was, until recently, no fast algorithm known to find a maximum matching in a random graph. In [Mot88], Motwani proves that the length of augmenting paths in random graphs is bounded almost surely. This implies that a direct application of Dinic’s matching algorithm runs in expected running time $O(|E| \log_e n / \log_e(np))$. This running time is strictly worse than the one of Angluin and Valiant’s algorithm and our algorithm in the range where these apply. However, it can be applied in the whole range of p for which a perfect matching exists almost surely, where Angluin and Valiant’s and our algorithm leave out the range $(\log_e n + t_n)/n \leq p < (c \log_e n)/n$.

Summing up the results of this paper and of previous ones on finding maximum matching in random graphs, the choice of an algorithm will depend on the value of p . For increasing values of p , the proposed algorithm is not only faster but also much easier to implement.

1. $(\log_e n + t_n)/n \leq p < (c \log_e n)/n$: use Motwani's algorithm with expected running time of $O(|E| \log_e n / \log_e(np)) = O(n \log_e^2 n / \log_e \log_e n)$.
2. $(c \log_e n)/n \leq p < (c' \log_e n)/n$, $c' > 2c$: use Angluin and Valiant's algorithm with expected running time $O(n \log_e n)$.
3. $(c' \log_e n)/n \leq p < 1$: use our combined algorithm with expected running time $O(n \log_e(1/p) + n)$.
4. $\lim_{n \rightarrow \infty} p = 1$: use our greedy algorithm with expected running time $O(n)$.

2. Notations and inequalities. A *slot* is defined by a pair of vertices $\{u, v\} \in V$. In a graph drawn from $\mathcal{G}_{n,p}$, a slot $\{u, v\}$ is occupied by an edge $\{u, v\}$ with probability p independently of the occupation of the other slots. It follows that the number of edges in a given set of N slots is a random variable, X , with binomial distribution $\text{bin}(N; p)$, that is

$$P[X = s] = \binom{N}{s} p^s (1-p)^{N-s}$$

and

$$P[X \leq s] = \sum_{j=0}^s \binom{N}{j} p^j (1-p)^{N-j}.$$

We use the Chernoff's bound on the tail of a binomial: for all $\beta > 0$,

$$(1) \quad \sum_{j=(1+\beta)np}^n \binom{n}{j} p^j (1-p)^{n-j} \leq e^{-\frac{\beta^2 np}{2}}.$$

Let X be the number of heads in a sequence of n independent coin flips, where the probability of heads on the i th flip is p_i . Let $p = (p_1 + p_2 + \dots + p_n)/n$ and $m \geq np + 1$. Then, by Hoeffding's lemma (in [Ren70]),

$$(2) \quad P[X \geq m] \leq \sum_{j=m}^n \binom{n}{j} p^j (1-p)^{n-j}.$$

The probability of an edge, p , will sometimes be written c_n/n where c_n is a function of the order of the graph.

3. The greedy matching algorithm.

3.1. Description. Let $G = (V, E)$, $V = \{v_1, \dots, v_n\}$, be an undirected graph. At each iteration, the greedy matching algorithm, GM , attempts to match two vertices in V . This is done by searching the neighbors of the undeleted vertex in V of smallest index, v_i , for an undeleted vertex v_j . In case there is such an undeleted neighbor, we delete the two matched vertices in V and record edge (v_i, v_j) in the matching. In case there is no such neighbor, we record the "unmatched" v_i in the "unmatched" set U and delete it from V . If there is more than one v_j candidate, we choose the first one found by the algorithm. The algorithm delivers a set of edges in the matching, M , and a set of unmatched nodes, U .

3.2. Probabilistic analysis. To perform a probabilistic analysis of the greedy algorithm, we assume that its input, $G = (V, E)$, is drawn from the space $\mathcal{G}_{n,p}$.

THEOREM 1. *Run the greedy matching algorithm on a graph $G = (V, E)$ drawn from $\mathcal{G}_{n,p}$.*

1. *If $\lim_{n \rightarrow \infty} p = 0$, then the number of unmatched vertices, $|U|$, is less than $(1 + \beta)(1/2p)$, for any real $\beta > 0$, with asymptotic probability 1.*

2. *If p is fixed, then $|U|$ is less than t_n with asymptotic probability 1, where t_n is any function such that $\lim_{n \rightarrow \infty} t_n = \infty$.*

3. *If $\lim_{n \rightarrow \infty} p = 1$, then the number of unmatched vertices, $|U|$, is equal to zero with asymptotic probability 1.*

Proof. The difficulty in analyzing the distribution of the number of unmatched vertices, $|U|$, is that the probability of success at iteration k , $k > 1$, depends on the number of successes in the previous iterations, since a success results in the deletion of the two matched vertices, whereas a failure results in the deletion of the one unmatched vertex. Therefore, we are going to perform the analysis on a variant of the greedy matching algorithm, GM' , for which the number of unmatched vertices $|U'|$ is greater or equal to $|U|$ (see proof in the appendix). The difference between the variant GM' and the original algorithm is that in case of failure, we delete two vertices from V , v_i , the one of smallest index we tried to match without success, and v_l , the one of smallest index among the undeleted vertices in V , after v_i has been deleted.

Iteration k , $k = 1, \dots, n/2$, of GM' fails with probability $q_k = (1 - p)^{n+1-2k}$, where $n + 1 - 2k$ is the number of undeleted vertices at iteration k . Let

$$\begin{aligned} q &= \frac{1}{n/2} \sum_{k=1}^{n/2} q_k \\ &= \frac{(1-p)^{n+1} - (1-p)}{(n/2)((1-p)^2 - 1)} \\ &= \frac{1-p - (1-p)^{n+1}}{(n/2)p(2-p)}. \end{aligned}$$

We can now apply Hoeffding's lemma (2) to the number of unmatched vertices $|U'|$. Let $m \geq (n/2)q + 1$, then,

$$(3) \quad P[|U'| \geq m] \leq \sum_{i=m}^{n/2} \binom{n/2}{i} q^i (1-q)^{n/2-i}.$$

Note that the mean of the binomial distribution,

$$(n/2)q = \frac{1-p - (1-p)^{n+1}}{p(2-p)} \leq \frac{1}{2p}.$$

1. If $\lim_{n \rightarrow \infty} p = 0$, we take $m = (1 + \beta)(1/2p)$ and apply Chernoff bound (1):

$$P\left[|U'| \geq (1 + \beta)\frac{1}{2p}\right] \leq \exp\left(-\frac{\beta^2}{2} \frac{1}{2p}\right) \rightarrow 0 \text{ if } p \rightarrow 0.$$

2. For p fixed, we choose $m = t_n$, $\lim_{n \rightarrow \infty} t_n = \infty$ and apply Markov inequality, using the fact that the mean $(n/2)q \leq 1/(2p)$:

$$P[|U'| \geq t_n] \leq \frac{1}{2pt_n} \rightarrow 0.$$

3. To prove the third part of Theorem 1, for $\lim_{n \rightarrow \infty} p = 1$, we use the special case of Markov inequality, $P[X > 0] \leq E[X]$, valid if X is a nonnegative integer valued random variable:

$$\begin{aligned}
 P[|U'| > 0] &\leq E[|U'|] \\
 &= (1-p)^{n-1} + (1-p)^{n-3} + \dots + (1-p) \\
 &= \frac{(1-p) - (1-p)^{n+1}}{p(2-p)} \\
 &\leq \frac{1-p}{p} \rightarrow 0 \text{ if } p \rightarrow 1.
 \end{aligned}$$

This concludes the proof of Theorem 1. $\square$

3.3. Implementation of the greedy matching and its expected running time.

3.3.1. Data structures. We assume that the input to the problem, i.e., the graph $G = (V, E)$, is given under the following data structure:

1. Adjacency list: for each vertex $v_i \in V$, its adjacency list $ADJ_i = \{v_j \in V \mid (v_i, v_j) \in E\}$ is in random order.
2. Full $n \times n$ adjacency matrix: given indices i and j , we can check the existence of edge (v_i, v_j) in unit time.

3. An array of n entries $V = \{v_1, v_2, \dots, v_n\}$. This array is arranged so that v_j is in the j th location so that, given an index j , we can access the corresponding entry, v_j , in unit time. In each entry there is a flag that indicates whether or not the vertex belongs to the set A of the remaining vertices. Specifically, the set A contains the vertices in $V - (M \cup U)$ where M is the set of matched vertices and U is the set of “unmatched” vertices (i.e., vertices which, when examined by the algorithm, had all their neighbors in M).

4. A word, smi , in which the smallest index among available vertices in A is recorded. Initially, it contains 1.

The random order of the adjacency lists allows the algorithm to run in expected $O(n \log_e(1/p))$ time. If the adjacency lists are not random, then we assume the capability to generate a random word in $O(1)$ time.

If the graph is sparse, $p \leq O(1/\sqrt{n})$, then we do not need the adjacency matrix. Explanation is provided in § 3.3.3.

3.3.2. Description of the algorithm. The greedy matching algorithm runs in three phases: matching a subset of the vertices, updating, and matching the remaining vertices. The first phase differs from the third in that it uses the adjacency list in order to find a mate, whereas in the third phase, the search is performed by inspecting edge slots. The purpose of this use of different data structures in the first and the third phases is to cut down the expected running time.

Define A_k to be the set A at the beginning of iteration k , $k \geq 1$.

PHASE I:

The first phase runs until $|A_k| \leq \lceil np \rceil$. At iteration k , $k \geq 1$, the algorithm attempts to match v_{smi} , the vertex of smallest index in A_k , with one of its neighbors in ADJ_{smi} . If ADJ_{smi} is in random order, it picks up the first $v_j \in ADJ_{smi}$ such that $v_j \in A_k$, whereas, if ADJ_{smi} is not random, it picks up members of ADJ_{smi} randomly (each $v_j \in ADJ_{smi}$ is equiprobably chosen) and without replacement until it finds $v_j \in A_k$. In both cases, it adds (v_{smi}, v_j) to the matching M and updates the set A , $A_{k+1} \leftarrow A_k - v_{smi} - v_j$, by flagging the two vertices in V as deleted. If no $v_j \in ADJ_{smi}$ belongs to A_k , then

v_{smi} is added to the set U , and $A_{k+1} \leftarrow A_k - v_{smi}$. Finally, smi is updated, $smi \leftarrow \{\text{smallest index in } A_{k+1}\}$.

Remark. If $p \leq O(1/\sqrt{n})$, then phase I is continued until $A = \emptyset$.

PHASE II:

Phase II contracts the array V by deleting all flagged vertices. The last position in V is recorded in a word $last$.

PHASE III:

Each iteration of phase III attempts to match the last vertex in V , v_{last} , with another vertex in V , v_j , by scanning V from the first vertex until $(v_{last}, v_j) \in E$ is found ((v_{last}, v_j) is added to M) or until v_{last-1} is reached (v_{last} is added to U). Finally, the array V is contracted in $O(1)$ time.

3.3.3. Expected running time. We now compute the expected running time of the greedy matching:

Since each vertex in ADJ_{smi} has probability $|A_k|/n$ to be in A_k , the expected number of vertices examined in list ADJ_{smi} is $n/|A_k|$, when vertex v_{smi} is inspected. Similarly, the expected number of vertices examined to update smi is also $n/|A_k|$. Therefore, the expected number of vertices examined in the first phase is:

$$\begin{aligned} E[\# \text{ vertices examined in phase 1}] &\leq 2 \sum_{i=np}^n \frac{n}{i} \leq 2n \int_{np}^n \frac{1}{i} \\ &= 2n(\log_e n - \log_e np) = 2n \log_e \left(\frac{1}{p} \right). \end{aligned}$$

Hence, the first phase takes $O(n \log_e (1/p))$ time.

If $p \leq O(1/\sqrt{n})$, then the expected total length of the $|A|$ remaining adjacency lists is $(np)^2 = O(n)$ and phase I is continued until $A = \emptyset$ with a total expected running time of the algorithm of $O(n \log_e (1/p))$.

The second phase can be implemented in $O(n)$ time.

Since each vertex in V has a probability p to be adjacent to v_i , the expected number of vertices examined in V for index i is $1/p$, and the expected number of vertices examined in the third phase is

$$E[\# \text{ vertices examined in phase 3}] \leq np \frac{1}{p} = n.$$

We conclude that the expected running time of the greedy algorithm is

$$O(n \log_e (1/p) + n).$$

4. A combined matching algorithm.

4.1. Description of the algorithm. From Theorem 1, it follows that, if $\lim_{n \rightarrow \infty} p = 1$, then the greedy algorithm delivers a perfect matching almost surely. However, if p is bounded by a constant strictly smaller than 1, then at most $O(1/p)$ (if $\lim_{n \rightarrow \infty} p = 0$) or less than t_n (if p constant) vertices remain unmatched at the end of the greedy matching algorithm. Let U , $|U| = m$, be the set of unmatched vertices. The idea is to match these vertices using the Angluin and Valiant's algorithm. The difficulty is that there is no edge between vertices in U , since, otherwise, the greedy matching would have used some of them. As a result the subgraph induced by U is not random. The trick is to match the vertices in U using Angluin and Valiant's algorithm on a bipartite subgraph $B = (S, T, E')$, whose source set, S , contains U , and the other vertices were set aside without looking them up in advance.

In order to construct such a bipartite subgraph $B = (S, T, E')$, we run the greedy matching algorithm not on the graph $G = (V, E)$ but on its subgraph G' on vertex set

$\{v_{s+1}, v_{s+2}, \dots, v_n\}$. Let $V' = \{v_1, \dots, v_s\}$ be the set of vertices set aside. The number s is chosen such that:

1. s is larger than m , the number of unmatched vertices, almost surely.
2. s is even.
3. $p \geq (c \log_e(s/2))/(s/2)$, where c is the constant used in Angluin and Valiant's paper.

Since m , the number of unmatched vertices at the end of the greedy algorithm, is even, the two first conditions insure that we can construct a bipartite graph $B = (S, T, E')$, using vertices from U and V' , such that S contains U . The third condition insures that Angluin and Valiant's algorithm, applied on $B = (S, T, E')$ delivers a perfect matching almost surely (i.e., $B = (S, T, E')$ contains enough vertices with respect to p). It follows from the proof of Claim 1 in 4.2, that s can be chosen to be the maximum of $(1/p) \log_e(1/p)^{c'}$, $c' > 2c$, and t_n , such that $\lim_{n \rightarrow \infty} t_n = \infty$.

We now describe the algorithm.

1. Run the greedy matching algorithm on the subgraph G' on vertex set $V - V'$. All vertices are matched except U , the unmatched vertices of the greedy matching, and V' .
2. Construct the bipartite graph $B = (S, T, E')$ as follows: S contains U and the first $(s + m)/2 - m$ vertices of V' . T contains the last $s - (s + m)/2 + m$ vertices of V' . E' is the set of edges between vertices of S and vertices of T in the original graph.
3. Run Angluin and Valiant's algorithm on $B = (S, T, E')$.

4.2. Probabilistic analysis. In order to prove that the combined algorithm described in the last section delivers a perfect matching with asymptotic probability 1, it remains to show that the Angluin and Valiant's algorithm delivers a perfect matching for the bipartite graph $B = (S, T, E')$.

CLAIM 1. *Angluin and Valiant's algorithm applied to $B = (S, T, E')$ delivers a perfect matching almost surely.*

Proof. By construction of $B = (S, T, E')$, a slot $\{s_i, t_j\}$, $s_i \in S$ and $t_j \in T$, contains an edge with probability p . Hence $B = (S, T, E')$ is equivalent to a bipartite graph drawn from $\mathcal{B}_{((s+m)/2), ((s+m)/2), p}$ where

$$(s + m)/2 \geq \frac{1}{2} \frac{1}{p} \log_e \left(\frac{1}{p} \right)^{c'}.$$

It remains to verify that $p \geq (c \log_e((s + m)/2))/((s + m)/2)$, where c is the constant used in Angluin and Valiant's paper. This is clearly the case when p is fixed and $s = t_n$, $\lim_{n \rightarrow \infty} t_n = \infty$. In the case $\lim_{n \rightarrow \infty} p = 0$, we have,

$$\begin{aligned} \frac{c \log_e((s + m)/2)}{(s + m)/2} &\leq \frac{c \log_e(s/2)}{s/2} = \frac{c \log_e \left(\frac{1}{2} \frac{1}{p} \log_e \left(\frac{1}{p} \right)^{c'} \right)}{\frac{1}{2} \frac{1}{p} \log_e \left(\frac{1}{p} \right)^{c'}} \\ &= \frac{2cp}{c'} \left[\frac{\log_e \left(\frac{1}{2p} \right) + \log_e \log_e \left(\frac{1}{p} \right)^{c'}}{\log_e \left(\frac{1}{p} \right)} \right] \leq \frac{2cp}{c'} [1 + o(1)] \leq p \text{ if } 2c < c'. \end{aligned}$$

This concludes the proof of the claim. $\square$

4.3. Range of application. The combined algorithm described in § 4.1 can be used only if n is larger than $s + m$; that is, if

$$n > \frac{1}{p} \log_e \left(\frac{1}{p} \right)^{c'} + (1 + \beta) \frac{1}{p}.$$

From Theorem 1, m is smaller or equal to $(1 + \beta)(1/p)$ with asymptotic probability 1. This condition on n is satisfied if and only if

$$p \geq \frac{c' \log_e n}{n}.$$

Hence the range of application of this combined algorithm is strictly contained in the range of application of Angluin and Valiant's algorithm: for

$$c \log_e n / n \leq p \leq c' \log_e n / n,$$

we would use the Angluin and Valiant algorithm directly on the input graph $G = (V, E)$.

4.4. Expected running time. The expected time taken by Angluin and Valiant's algorithm on the subgraph $B = (S, T, E')$ is $O((1/p) \log_e^2(1/p))$ time. This time is dominated by the time taken by the greedy matching algorithm, $O(n \log_e(1/p) + n)$, for any $p \geq c' \log_e n / n$, so the overall expected time of this combined algorithm is $O(n \log_e(1/p) + n)$.

5. An algorithm for the bipartite matching. If the input graph is a bipartite graph $B = (S, T, E)$ drawn from the probability space $\mathcal{B}_{n,n,p}$, we adapt the greedy matching of § 3.1 as follows: one by one we try to match a vertex in S with one of its neighbors in T . In case of success, we delete the matched vertex in T and record the two matched vertices, one from S and the other from T , in the matching. In case of failure, we record the unmatched s_i in the unmatched set.

The statement of Theorem 1 of § 3.2 is similar in the bipartite case, except that the average number of unmatched vertices, $E[|U|]$, is bounded by $1/p$ instead of $1/2p$. The proof is also similar and the expected running time of the greedy algorithm in the bipartite case is $O(n \log_e(1/p) + n)$.

The combined algorithm is adapted as follows: Let B'' be the subgraph of B on vertex set $(S'' \cup T'')$ where $S'' = \{s_1, \dots, s_s\}$ and $T'' = \{t_1, \dots, t_s\}$. The number s is chosen as in the nonbipartite case.

1. Run the greedy matching algorithm on the subgraph $B - B''$. All vertices are matched except S' and T' , the unmatched vertices of the greedy matching, and S'' and T'' . Recall that s is chosen such that $|S'| = |T'| = m < s = |S''| = |T''|$ almost surely.

2. Construct the bipartite graphs $B_1 = (S_1, T_1, E_1)$ and $B_2 = (S_2, T_2, E_2)$ as follows (see Fig. 1): S_1 contains S' and the first $(s + m)/2 - m$ vertices of S'' . S_2 contains the last $s - (s + m)/2 + m$ vertices of S'' . T_2 contains T' and the first $(s + m)/2 - m$ vertices of T'' . T_1 contains the last $s - (s + m)/2 + m$ vertices of T'' . $E_1(E_2)$ is the set of edges between vertices of $S_1(S_2)$ and $T_1(T_2)$ in the original graph.

3. Run Angluin and Valiant's algorithm on B_1 and B_2 .

Similarly, we can prove that Angluin and Valiant's algorithm applied to B_1 (B_2) delivers a perfect matching almost surely.

The proofs of this section are similar to those in the nonbipartite case, so they are omitted.

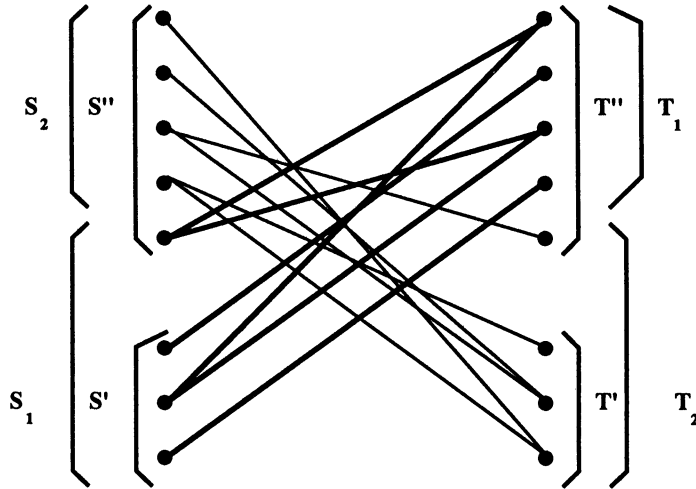


FIG. 1

6. Concluding remarks. The main advantage of our algorithm over the Angluin and Valiant algorithm is its faster expected running time.

A perfect matching algorithm is used, as a subroutine, in several algorithms, which solve problems in random graphs. In [KMN87], algorithms for the maximum-flow problem and related capacitated transportation problems are presented. These algorithms use the Angluin and Valiant perfect matching algorithm as a subroutine. Because these algorithms only apply to dense graphs (p constant), the alternative use of our algorithm would improve the running time of these algorithms (or at least the phase in which Angluin and Valiant's is used) by a factor of $O(\log_e n)$. Another example where a perfect matching algorithm is used as a subroutine is Hochbaum's algorithm to find a maximum flow in a random graph [Hoc88]. (That algorithm differs from the one in [KMN87] in that it can also be applied to sparse graphs and its running time is faster. It is only applicable though to unweighted random graphs.) [Hoc88] includes a proof that, when $p \geq \sqrt{(\log_e n)/n}$, then there exists almost surely a perfect matching between the neighbors of the source and the neighbors of the sink (after the size of the two sets has been made equal by removing vertices in the larger set). It is also proved that, for this range of p , it is faster to use our perfect matching algorithm than other proposed procedures. Hence, the greedy matching algorithm has several practical applications either when used directly, or when used as a subroutine of maximum-flow's algorithms and capacitated transportation's algorithm.

7. Appendix. It is important to note that the vertices are accessed in order of increasing value (index value). The adjacency list is random but identical for both GM and GM' .

It is proved here that given the same random graph with the same adjacency lists as input, the number of unmatched vertices in GM' , $|U'|$, is at least as large as the number of unmatched vertices in GM , $|U|$. Let the maximal matching which is the output of GM be M , and the one which is the output of GM' be M' . An equivalent statement is $|M| \geq |M'|$.

CLAIM. $|M| \geq |M'|$.

Proof. Suppose not. Consider the graph $(V, M \oplus M')$ (where $\oplus$ means the symmetric difference). Since $|M'| > |M|$, there exists an odd length alternating path in this graph

$(v_0, v_1, \dots, v_k)$ with $(v_{2i}, v_{2i+1}) \in M'$, $i = 0, 1, \dots, \lfloor k/2 \rfloor$, and $(v_{2i-1}, v_{2i}) \in M$, $i = 1, \dots, \lfloor k/2 \rfloor$, and k odd. Note that none of the vertices on the path is discarded in GM' before it is matched. The case of $k = 1$ is clearly impossible, since both v_0 and v_1 are unmatched in GM : one of them would have been accessed first and matched. So, $k > 2$ and odd. Now, $\min \{v_1, v_2\} < v_0$, since otherwise v_0 would have been accessed first by GM and would be matched in M . Also, $v_1 > v_0$ since otherwise v_1 would be accessed first in GM' and matched, in GM' , with v_2 , which appears in v_1 's adjacency list before v_0 (else, $(v_1, v_0) \in M$). Therefore $v_2 = \min \{v_0, v_1, v_2\}$.

It is now proved that $v_k = \min \{v_0, v_1, \dots, v_k\}$. The proof is by induction for even and odd cases:

1. Suppose $v_{2i} = \min \{v_0, v_1, \dots, v_{2i}\}$, for some $i \in \{1, \dots, \lfloor k/2 \rfloor\}$. If $v_{2i+1} > v_{2i}$, then v_{2i} is accessed first of $v_0, \dots, v_{2i+1}$, and v_{2i-1} appears on its adjacency list before v_{2i+1} . Therefore M' would include (v_{2i-1}, v_{2i}) , which is a contradiction. Hence, $v_{2i+1} = \min \{v_0, v_1, \dots, v_{2i+1}\}$.

2. Suppose $v_{2i-1} = \min \{v_0, v_1, \dots, v_{2i-1}\}$, for some $i \in \{2, \dots, \lfloor k/2 \rfloor\}$. If $v_{2i} > v_{2i-1}$, then v_{2i-1} is accessed first of $v_0, \dots, v_{2i}$, and v_{2i} appears on its adjacency list before v_{2i-2} . Therefore M' would include (v_{2i-1}, v_{2i}) , which is a contradiction. Hence, $v_{2i} = \min \{v_0, v_1, \dots, v_{2i}\}$.

Therefore, $v_k = \min \{v_0, v_1, \dots, v_k\}$. But then v_k is accessed first in GM and has to be matched. So v_k cannot be exposed in M , contradiction. $\square$

REFERENCES

- [AV79] D. ANGLUIN AND L. G. VALIANT, *Probabilistic algorithms for Hamiltonian circuits and matchings*, J. Comput. System Sci., 18 (1979), pp. 155–190.
- [ER60] P. ERDÖS AND A. RÉNYI, *On the evolution of random graphs*, Publ. Math. Inst. Hung. Acad. Sci., 5 (1960), pp. 17–61.
- [ER64] ———, *On random matrices I*, Publ. Math. Inst. Hung. Acad. Sci., 8 (1964).
- [ER66] ———, *On the existence of a factor of degree one of connected random graphs*, Acta Math. Acad. Sci. Hung., 17 (1966), pp. 359–379.
- [Hoc88] D. S. HOCHBAUM, *An exact sublinear algorithm for the max-flow, maximum number of vertex disjoint paths and communication problem on a random graph*, University of California, Berkeley, manuscript, 1988.
- [KMN87] R. M. KARP, R. MOTWANI, AND N. NISAN, *Probabilistic analysis of network flow algorithms*, Math. Oper. Res., (1987), submitted.
- [Mot88] R. MOTWANI, *Average-case analysis for matchings, permanents, factors and flows*, preliminary draft, 1988.
- [MV80] S. MICALI AND V. V. VAZIRANI, *An $O(\sqrt{|V|} |E|)$ algorithm for finding maximum matching in general graphs*, in Proc. of 21st Annual Symposium on the Foundation of Computer Science, 1980, pp. 17–27.
- [Ren70] A. RÉNYI, *Foundations of Probability*, Holden-Day, San Francisco, 1970.

ASYMPTOTICALLY OPTIMAL LINEAR ALGORITHM FOR THE MINIMUM k -CUT IN A RANDOM GRAPH*

OLIVIER GOLDSCHMIDT† AND DORIT S. HOCHBAUM‡

Abstract. The k -cut problem is to find a partition of a graph into k nonempty components, such that the number of edges between components is minimum. A random graph in $\mathcal{G}_{n,p}$ is a simple graph on n vertices with each pair of vertices connected by an edge with probability p . It is proved that, when k is fixed, a k -cut of almost every graph from $\mathcal{G}_{n,p}$ consists of $k - 1$ isolated vertices and one component on the remaining $n - k + 1$ vertices. An important outcome of this property is a linear algorithm that derives the minimum k -cut in such graphs, almost certainly.

Key words. k -cut, random graphs

AMS(MOS) subject classification. 05C80

1. Introduction. Given a simple and connected graph $G = (V, E)$, $|V| = n$, a k -cut in G is a set of edges, E' , whose deletion separates the graph into k nonempty components. The k -cut problem is to find such a set E' of minimum cardinality.

The k -cut problem has recently been proved polynomial for fixed k , even in edge-weighted graphs [GH88] and NP-complete if k is part of the input.

The k -cut problem was formulated in connection to generating travelling salesperson cutting planes [Pul82]. It also has numerous other applications, particularly in clustering-related setups, such as VLSI design. The k -cut problem is easily interpreted as a clustering problem. The edge weights represent the similarity between each pair of objects (represented by nodes). The k clusters that maximize the similarity between objects in the same clusters or alternatively maximize the dissimilarity between objects in separate clusters is identical to the k -cut problem. Another area of applications is network reliability where the value of the optimal k -cut is an important factor in assessing the reliability (or its lack thereof) of the network.

The purpose of this paper is to analyze the structure of the components of a minimum k -cut in random graphs. The main result of this paper is that, when k is fixed, the components of a minimum k -cut in a random graph consist of $k - 1$ isolated vertices and one big component. This structure leads to a simple heuristic for solving the k -cut problem: take the set of edges adjacent to $k - 1$ vertices of minimum degree. We prove that this heuristic yields a minimum k -cut with probability going to one asymptotically.

In terms of the applications of the k -cut problem to clustering and reliability, this outcome implies that all clusters except for one will have one vertex. That is, the clusters will not be “balanced.” The interpretation is that balanced clustering is not compatible with minimum cut clustering.

The model of random graphs, $\mathcal{G}_{n,p}$, used was first defined by Erdős and Rényi in an important paper on random graphs [ER60]: the points in the probability space $\mathcal{G}_{n,p}$

* Received by the editors June 13, 1989; accepted for publication (in revised form) July 14, 1989.

† Mechanical Engineering Department, University of Texas, Austin, Texas 78712. This research was supported in part by Institute for Business and Economic Research School of Business Administration, University of California, Berkeley.

‡ Industrial Engineering and Operations Research Department and School of Business Administration, University of California, Berkeley, California 94720. This research was supported in part by Office of Naval Research grant N00014-88-K-0377.

are simple graphs with vertex set $\{1, 2, \dots, n\}$; independently, for each pair of vertices $\{i, j\}$, the probability that a graph drawn from $\mathcal{G}_{n,p}$ has an edge is p .

Our interest is with connected random graphs. That is because for disconnected graphs the problem is trivially reduced to a $(k-1)$ -cut on a smaller graph.

When $p = c_n/n$ is a function of the order of the graph, Erdős and Rényi [ER60] showed that, if $p = (\log_e n + x)/n$, then the probability that a graph drawn from $\mathcal{G}_{n,p}$ is connected is $e^{-e^{-x}}$. Hence, if $p \geq (\log_e n + w_n)/n$, $\lim_{n \rightarrow \infty} w_n = \infty$, then almost all graphs from $\mathcal{G}_{n,p}$ are connected.

In this paper, we show that, when the graph is asymptotically connected (i.e., when $p \geq (\log_e n + w_n)/n$, $\lim_{n \rightarrow \infty} w_n = \infty$), a minimum k -cut partition of almost every graph from $\mathcal{G}_{n,p}$ consists of a subgraph on $n - k + 1$ vertices and $k - 1$ isolated vertices. In order to prove this, we consider any k -cut with a component whose size is between 2 and $n - k$, and, from it, we construct a k -cut with $k - 1$ isolated vertices which is, for each of the cuts, smaller with high probability. More precisely, we determine the probability that the constructed k -cut is not smaller than the k -cut it is constructed from. It is then proved that the sum of all these probabilities is asymptotically equal to zero.

In order to prove such results, it is shown that whenever there are components containing more than one vertex, the probability is exponentially close to 1 that there is another, smaller valued, k -cut. The cases in which this probability is the least overwhelming are the ones in which those components containing more than one vertex contain only few vertices, say two.

The greedy heuristic, which consists of taking the set of edges adjacent to $(k-1)$ vertices of minimum degree in the original graph, is proved to yield a minimum k -cut for almost every graph, when $(\log_e n + w_n)/n \leq p \leq 1 - (w_n \log_e n)/n$, $\lim_{n \rightarrow \infty} w_n = \infty$. We prove that, with asymptotic probability one, no vertex, in the subgraph on the remaining $n - k + 1$ vertices, has degree strictly less than any of the $k - 1$ vertices selected.

2. Notations and inequalities. Let $\{a_n\}$ and $\{b_n\}$ be sequences of nonnegative real numbers. The big- O and little- o notations are defined with the usual interpretation:

$$a_n = O(b_n),$$

if and only if for some positive constants K and N , $a_n \leq Kb_n$ for all $n > N$. Thus $a_n = O(1)$ means that $\{a_n\}$ is a bounded sequence.

$$a_n = o(b_n),$$

if and only if $\limsup_{n \rightarrow \infty} a_n/b_n = 0$.

It is well known (see, for instance, [Lin65]), that the probability of the union of events is not larger than the sum of their probabilities: let A_i , $i = 1, \dots, n$, be events whose probability of realization is $P[A_i]$.

$$(1) \quad P\left[\bigcup_{i=1}^n A_i\right] \leq \sum_{i=1}^n P[A_i]$$

If X is an integer-valued nonnegative random variable, then the probability that X is different from zero is smaller than or equal to the expected value of X :

$$(2) \quad P[X > 0] \leq E[X].$$

A *slot* is defined by a pair of vertices $\{i, j\}$. In a graph drawn from $\mathcal{G}_{n,p}$, a slot $\{i, j\}$ is occupied by an edge (i, j) with probability p independently of the occupation

of the other slots. It follows that the number of edges in a given set of N slots is a random variable, X , with binomial distribution $\text{Bin}(n; p)$, namely,

$$P[X = s] = \binom{N}{s} p^s (1-p)^{N-s}$$

and

$$P[X \leq s] = \sum_{j=0}^s \binom{N}{j} p^j (1-p)^{N-j}.$$

We use the following bound [Pal86] on $P[X \leq s]$ when $s \leq Np$

$$(3) \quad P[X \leq s] = \sum_{j=0}^s \binom{N}{j} p^j (1-p)^{N-j} \leq \binom{N}{s} p^s (1-p)^{N-s} \frac{p(N-s+1)}{p(N+1)-s}.$$

$(n)_k$ denotes the falling factorial:

$$(n)_k = n(n-1) \cdots (n-k+1).$$

The following bounds on factorials and on binomial coefficients can be found, for instance, in [Pal86].

$$(4) \quad n! \leq n^n$$

$$(5) \quad \binom{n}{k} \leq \left(\frac{ne}{k} \right)^k$$

$$(6) \quad (1+x) \leq e^x.$$

We also use the following Bernoulli inequality ([Mit70]): For any $x > -1$ and for any positive integer n ,

$$(7) \quad (1+x)^n \geq 1 + nx.$$

We shall say that almost every graph drawn from $\mathcal{G}_{n,p}$ has a certain property Q if $P[Q] \rightarrow 1$ as $n \rightarrow \infty$.

In order to simplify the formulae, we consider, hereafter, $(k+1)$ -cuts instead of k -cuts. A $(k+1)$ -cut is said to be *canonical* if its components are k isolated vertices and one big component on $n-k$ vertices.

The probability of an edge, p , will sometimes be written c_n/n , where c_n is a function of the number of vertices of the graph.

3. The main theorem.

THEOREM 1. *If $p \geq (\log_e n + w_n)/n$, $\lim_{n \rightarrow \infty} w_n = \infty$, then a minimum $(k+1)$ -cut of almost every graph drawn from $\mathcal{G}_{n,p}$ is canonical: its components are k isolated vertices and one “big” component on $n-k$ vertices.*

Proof. A noncanonical $(k+1)$ -cut necessarily contains at least one component whose number of vertices is between 2 and $(n-k-1)$. Suppose that we randomly select a subset of V , called the $(k+m)$ -subset which is to be the union of $k-1$ of the components of the $(k+1)$ -cut. The size of that subset is $k+m$ with m between 1 and $(n-k-2)$. (By random selection we mean that any of the $\sum_{m=1}^{n-k-2} \binom{n}{k+m}$ such subsets is equiprobably picked up.) The remaining vertices of V form what we refer to hereafter as the main component, whose size assumes a value between 2 and $(n-k-1)$ vertices. We then arrange the vertices of the $(k+m)$ -subset into k components, called subcomponents, such that the number of edges between subcomponents is minimized (see Fig.

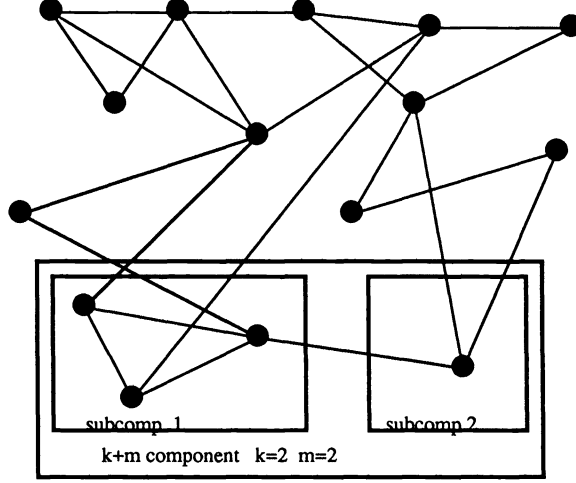


FIG. 1

1). If there is more than one such minimum cut partition, one is selected arbitrarily. Together with the main component, these subcomponents form a noncanonical $(k + 1)$ -cut which is minimal with respect to the randomly selected main component. By this, we mean that if one of the components is fixed to be that main component, the $(k + 1)$ -cut obtained is minimum.

The number of these minimal noncanonical $(k + 1)$ -cuts is equal to the number of possible $(k + m)$ -subsets, i.e., $\sum_{m=1}^{n-k-1} \binom{n}{k+m}$. We label these minimal noncanonical $(k + 1)$ -cuts by the parameters (m, j) where $m = 1, \dots, n - k - 1$, signifying the number of extra vertices in the components and $j = 1, \dots, \binom{n}{k+m}$ indicates which subset was selected of the $(m + k)$ -subsets.

Given a minimal noncanonical $(k + 1)$ -cut, say (m, j) , we construct a canonical $(k + 1)$ -cut by a prescribed method.

Let $A_{m,j}$ be the event that the canonical $(k + 1)$ -cut constructed from the cut (m, j) is larger than or equal to $(k + 1)$ -cut (m, j) . The probability that, in a graph drawn from $\mathcal{G}_{n,p}$, at least one noncanonical $(k + 1)$ -cut forms a minimum $(k + 1)$ -cut is smaller than or equal to $P[\cup A_{m,j}]$. Theorem 1 is proved by showing that $P[\cup A_{m,j}]$ tends to zero as n goes to infinity. Of course the realization of event $A_{m,j}$ depends on the method used to construct a canonical $(k + 1)$ -cut from cut (m, j) . We shall use different methods depending on the size of the $(k + m)$ -component.

$$(8) \quad P[\cup A_{m,j}] \leq \sum_{m=1}^{n-k-1} \sum_{j=1}^{\binom{n}{k+m}} P[A_{m,j}],$$

The summation in (8) is split into three parts, S_1 , S_2 , and S_3 :

$$(9) \quad S_1 = \sum_{j=1}^{\log_e n} \sum_{j=1}^{\binom{n}{k+m}} P[A_{m,j}]$$

$$(10) \quad S_2 = \sum_{m=\log_e n+1}^{(k/k+1)n} \sum_{j=1}^{\binom{n}{k+m}} P[A_{m,j}]$$

$$(11) \quad S_3 = \sum_{m=(k/k+1)n+1}^{n-k-2} \sum_{j=1}^{\binom{n}{k+m}} P[A_{m,j}].$$

In the following three lemmas, it is proven that each of these parts tends to zero as n goes to infinity. Hence, the sum in inequality (8) is asymptotically equal to zero, which concludes the proof of the theorem. $\square$

We use two different methods for the construction of a canonical $(k+1)$ -cut from a noncanonical $(k+1)$ -cut, depending on the value of m .

(1) $m \leq \log_e n$. Here m vertices are transferred from the subcomponents to the main component. These m vertices are those which have the largest number of neighbours in the main component (see Fig. 2). The subcomponents together contain $k+m$ vertices. After transferring m vertices to the main component, the subcomponents will contain k vertices, one vertex in each.

(2) $m > \log_e n$. In this case, a canonical $(k+1)$ -cut is formed by stripping each subcomponent of all of its vertices but one and transferring these vertices to the main component (see Fig. 3).

LEMMA 1. S_1 tends to zero as n goes to infinity.

Proof.

$$(12) \quad S_1 = \sum_{m=1}^{\log_e n} \sum_{j=1}^{\binom{n}{k+m}} P[A_{m,j}]$$

where $A_{m,j}$ is the event that the canonical $(k+1)$ -cut is not smaller than the cut (m, j) it is constructed from.

By transferring m vertices from the subcomponents to the main component, edges are removed from the $(k+1)$ -cut. Later, splitting the larger than one vertex subcomponents, which remain, into single vertex components, edges are added to the $(k+1)$ -cut. Let E_O be the set of edges that connect the subcomponents to the main component and let E_I be the set of edges whose ends both lie in the subcomponents (see Fig. 2).

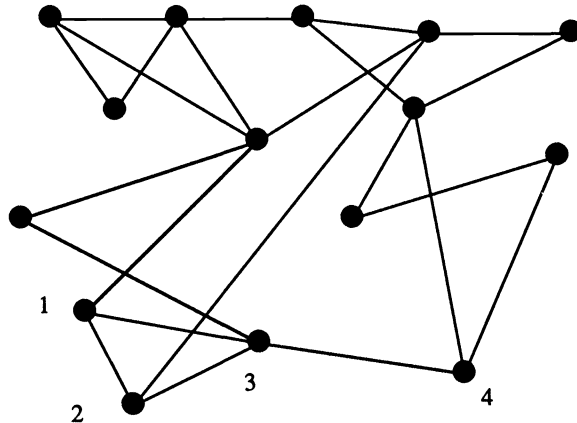


FIG. 2. The $k+m$ component is $\{1, 2, 3, 4\}$; the subcomponents 1 and 2 are $\{1, 2, 3\}$ and $\{4\}$, respectively. Vertices 3 and 4 are transferred to the main component. The 3-cut has size 7 before the transfer and size 5 after the transfer.

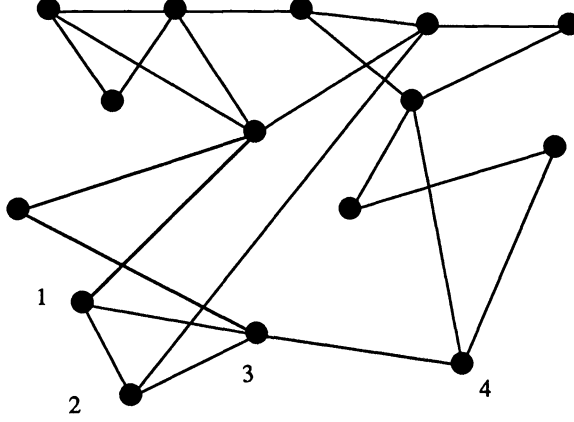


FIG. 3. The $k + m$ component is $\{1, 2, 3, 4\}$; the subcomponents 1 and 2 are $\{1, 2, 3\}$ and $\{4\}$, respectively. Vertices 1 and 2 are transferred to the main component. The 3-cut has size 7 before and after the transfer.

How many edges are removed from the $(k + 1)$ -cut (m, j) ?

The edges removed are a subset of E_O that connects the transferred vertices to the main component. Because these vertices are the most connected ones, the number of removed edges, e_s , from cut (m, j) satisfies

$$(13) \quad e_s \geq \frac{m}{k+m} |E_O|.$$

How many edges are added to the $(k + 1)$ -cut (m, j) ?

We add a subset of E_I : the ones whose ends both lie in the same subcomponent. Therefore the number of added edges, e_a , is

$$(14) \quad e_a \leq |E_I|.$$

For every possible choice of the $(k + m)$ -subset, $1 \leq m \leq \log_e n$, $A_{m,j}$ occurs if and only if $e_a \geq e_s$. Therefore,

$$(15) \quad S_1 = \sum_{m=1}^{\log_e n} \binom{n}{k+m} P[e_a \geq e_s]$$

and, by inequalities (13) and (14)

$$(16) \quad S_1 \leq \sum_{m=1}^{\log_e n} \binom{n}{k+m} P\left[|E_I| \geq \frac{m}{k+m} |E_O|\right].$$

The rest of the proof of Lemma 1 is split into two cases.

Case 1. $p \leq (\log_e n)^2/n$.

The maximum number of edges in E_I is $\binom{k+m}{2}$, and the maximum number of edges in E_O is $(n - (k + m))(k + m)$. Because the $(k + m)$ -subset is picked randomly, $|E_I|$ and $|E_O|$ are independent and have binomial distributions $\text{Bin}(\binom{k+m}{2}; p)$ and $\text{Bin}((n - (k + m))(k + m); p)$, respectively.

The desired result for S_1 will follow from bounding each of the expressions:

$$\begin{aligned}
 (17) \quad P\left[|E_I| \geq \frac{m}{k+m} |E_O|\right] &= \sum_{j=1}^{\binom{k+m}{2}} P[|E_I| = j] P\left[\frac{m}{k+m} |E_O| \leq j \mid |E_I| = j\right] \\
 &= \sum_{j=1}^{\binom{k+m}{2}} P[|E_I| = j] P\left[\frac{m}{k+m} |E_O| \leq j\right].
 \end{aligned}$$

Now,

$$(18) \quad P\left[\frac{m}{k+m} |E_O| \leq j\right] = P\left[|E_O| \leq \frac{k+m}{m} j\right].$$

To simplify the formulae, we use the notation $k' = (k+m)/m$ and $s = (n - (k+m))(k+m)$.

$$(19) \quad P[|E_O| < k'j] \leq \sum_{i=0}^{k'j} \binom{s}{i} p^i (1-p)^{s-i}.$$

CLAIM 1. *The terms in (19) are increasing.*

Proof. We know that if a random variable X has a binomial distribution $\text{Bin}(n, p)$, the probabilities increase from $P[X = 0]$ to $P[X = np]$ where np is the mean. In the case we are concerned with, the mean is

$$\begin{aligned}
 (20) \quad sp = sc_n/n &= (k+m)c_n - (k+m)^2 c_n/n \\
 &= (k+m)c_n - \frac{(k+m)^2 c_n}{n}.
 \end{aligned}$$

It now follows that $sp > k'j$ since

$$\begin{aligned}
 (k+m)c_n \left(1 - \frac{(k+m)}{n}\right) &> (k+m)c_n(1 - \varepsilon) \text{ for any } \varepsilon > 0 \\
 &\geq (k+m)(\log_e n + w_n)(1 - \varepsilon) \\
 &> \frac{(k+m)^3}{2m} \\
 &\geq ((k+m)/m) \binom{k+m}{2} \geq k'j.
 \end{aligned}$$

(Recall that for S_1 $m \leq \log_e n$). $\square$

Due to Claim 1, we can use (3) to derive the bound:

$$(21) \quad P[|E_O| \leq k'j] \leq k'j \binom{s}{k'j} p^{k'j} (1-p)^{s-k'j}.$$

We substitute (21) in (17) and obtain:

$$(22) \quad \begin{aligned} P\left[|E_I| \geq \frac{m}{k+m} | E_O\right] &\leq \sum_{j=0}^{\binom{k+m}{2}} P[|E_I| = j] k'j \binom{s}{k'j} p^{k'j} (1-p)^{s-k'j} \\ &= \sum_{j=0}^{\binom{k+m}{2}} \binom{\binom{k+m}{2}}{j} p^j (1-p)^{\binom{k+m}{2}-j} k'j \binom{s}{k'j} p^{k'j} (1-p)^{s-k'j}. \end{aligned}$$

CLAIM 2. If $0 \leq j \leq \binom{k+m}{2}$, $1 \leq m \leq \log_e n$, and $c_n \leq (\log_e n)^2$, then

$$f = \binom{\binom{k+m}{2}}{j} p^j (1-p)^{\binom{k+m}{2}-j} k'j \binom{s}{k'j} p^{k'j} \leq 1.$$

Proof. For $j = 0$, $f = 0$.

For $j > 0$, we have

$$(23) \quad (1-p)^{\binom{k+m}{2}-j} \leq 1.$$

Using (5), we write,

$$(24) \quad \binom{s}{k'j} p^{k'j} \leq \left(\frac{sep}{k'j}\right)^{k'j}.$$

Now $s/k' \leq mn$, hence,

$$(25) \quad \left(\frac{sep}{k'j}\right)^{k'j} \leq (me)^{k'j} (np)^{k'j}.$$

Using (23) and (25), and, since $\binom{\binom{k+m}{2}}{j} \leq (k+m)^{2j}$, we have

$$(26) \quad \begin{aligned} f &\leq (k+m)^{2j} k'j (me)^{k'j} c_n^{(k'+1)j} n^{-j} \\ &= k'j ((k+m)^2 (me)^{k'} c_n^{(k'+1)})^j n^{-j}. \end{aligned}$$

Since, $m \leq \log_e n$ and $c_n \leq (\log_e n)^2$, we conclude that $f = o(1)$ when $j > 0$. $\square$

Moreover,

$$(27) \quad p(s - k'j) = pn(k+m) - o(1).$$

From Claim 2 and (27), inequality (22) becomes:

$$(28) \quad P\left[|E_I| \geq \frac{m}{k+m} | E_O\right] \leq \binom{k+m}{2} e^{-pn(k+m) + o(1)}.$$

Therefore,

$$(29) \quad S_1 \leq \sum_{m=1}^{\log_e n} \binom{n}{k+m} \binom{k+m}{2} e^{-pn(k+m) + o(1)}.$$

Since

$$\begin{aligned} \binom{n}{k+m} \binom{k+m}{2} &= \frac{(n)_{k+m}}{(k+m)!} \frac{(k+m)!}{(k+m-2)!} \\ &< n^{k+m}, \end{aligned}$$

it follows that

$$(30) \quad \begin{aligned} S_1 &\leq \sum_{m=1}^{\log_e n} n^{k+m} e^{-(k+m)c_n + o(1)} \\ &\leq \sum_{m=1}^{\log_e n} e^{-(k+m)(c_n - \log_e n) + o(1)}. \end{aligned}$$

By the assumption on p , $c_n - \log_e n \rightarrow \infty$ as $n \rightarrow \infty$, so $S_1 \rightarrow 0$ as $n \rightarrow \infty$ when $c_n \leq (\log_e n)^2$. This concludes the first part of the proof of Lemma 1.

Case 2. $p > (\log_e n)^2/n$.

Because the graph is much more dense in this case, we can prove that the probability that E_O contains more than $(k+m)/m(\frac{k}{2}^m)$, edges tends to 1 sufficiently fast as n goes to infinity. Again, as in the first part of the proof, define $(k+m)/m(\frac{k}{2}^m) = k'$ and $(n - (k+m))(k+m) = s$.

$$(31) \quad P[|E_O| \leq k'] \leq \sum_{i=0}^{k'} \binom{s}{i} p^i (1-p)^{s-i}.$$

By Claim 1, the terms of the sum in (31) are increasing. Therefore, inequality (31) is bounded by:

$$(32) \quad P[|E_O| \leq k'] \leq k' \binom{s}{k'} p^{k'} (1-p)^{s-k'}.$$

Using (5) and (6), we bound the right-hand side of inequality (32):

$$(33) \quad P[|E_O| \leq k'] \leq k' \left(\frac{n(k+m)ep}{k'} \right)^{k'} e^{-p(s-k')}.$$

Now $(k+m)e/k' < 1$, so we have

$$(34) \quad \begin{aligned} P[|E_O| \leq k'] &\leq k' (np)^{k'} e^{-p(s-k')} \\ &\leq k' c_n^{k'} e^{-(k+m)c_n} e^{p((k+m)^2 + k')}. \end{aligned}$$

Therefore,

$$(35) \quad \begin{aligned} S_1 &\leq \sum_{m=1}^{\log_e n} \binom{n}{k+m} k' c_n^{k'} e^{-(k+m)c_n} e^{p((k+m)^2 + k')} \\ &\leq \sum_{m=1}^{\log_e n} n^{k+m} k' c_n^{k'} e^{-(k+m)c_n} e^{p((k+m)^2 + k')} \\ &\leq \sum_{m=1}^{\log_e n} k' e^{-(k+m)(c_n - \log_e n)} e^{k' \log_e c_n + p((k+m)^2 + k')}. \end{aligned}$$

Because $c_n - \log_e n = O((\log_e n)^2)$ and $m \leq \log_e n$, $S_1 \rightarrow 0$ as $n \rightarrow \infty$, which concludes the proof of Lemma 1. $\square$

LEMMA 2. S_2 tends to zero as n goes to infinity.

Proof.

$$(36) \quad S_2 = \sum_{m=\log_e n+1}^{(k/(k+1))n} \sum_{j=1}^{\binom{n}{k+m}} P[A_{m,j}]$$

where $A_{m,j}$ is the event that the canonical $(k+1)$ -cut constructed from cut (m, j) is not smaller than cut (m, j) .

Recall that the method we use, in this case, to construct a canonical $(k+1)$ -cut from the noncanonical $(k+1)$ -cut (m, j) , consists of taking all but one vertices from each subcomponent and reallocating these vertices to the main component. The number of vertices reallocated is m and the result is a canonical $(k+1)$ -cut. The maximum number of edges added, e_a , to the cut (m, j) is m since the added edges are only between the remaining single vertex in a subcomponent and the other vertices in that subcomponent. The set of edges removed from the cut (m, j) consists of e_s edges between the main component and the m reallocated vertices. Because the main component is randomly chosen, e_s has a binomial distribution $\text{Bin}((n - (k + m))m, p)$.

For a given m ,

$$\begin{aligned}
 (37) \quad P[A_{m,j}] &= P[e_s < e_a] \\
 &\leq P[e_s < m] \\
 &\leq \sum_{i=0}^m \binom{s}{i} p^i (1-p)^{s-i}
 \end{aligned}$$

where $s = (n - (k + m))m$.

In order to apply inequality (3) to (37), we need to show that $m < ps$, the mean of the binomial distribution of e_s .

$$\begin{aligned}
 (38) \quad ps &= \frac{c_n s}{n} \\
 &= c_n m - \frac{c_n k m}{n} - \frac{c_n m^2}{n}.
 \end{aligned}$$

Therefore $m < ps$ if and only if

$$(39) \quad c_n(1 - k/n - m/n) > 1$$

which is true since $m \leq (k/(k+1))n$, and $c_n > \log_e n$.

Therefore, by (3), (37) can be bounded by

$$(40) \quad P[A_{m,j}] \leq \binom{s}{m} p^m (1-p)^{s-m} \frac{p(s-m+1)}{p(s+1)-m}.$$

We show that $p(s-m+1)/(p(s+1)-m) = O(1)$:

$$(41) \quad \frac{p(s-m+1)}{p(s+1)-m} = 1 + \frac{(1-p)m}{p(s+1)-m}$$

$$\begin{aligned}
 (42) \quad &\frac{(1-p)m}{p(s+1)-m} < \frac{m}{ps-m} \\
 &= \frac{m}{p(nm - km - m^2) - m} \\
 &= \frac{1}{c_n(1 - k/n - m/n)} = o(1).
 \end{aligned}$$

Therefore, inequality (40) becomes

$$(43) \quad P[A_{m,j}] \leq \binom{s}{m} p^m (1-p)^{s-m} O(1).$$

By inequalities (5) and (6), inequality (43) becomes

$$(44) \quad P[A_{m,j}] \leq \left(\frac{sep}{m}\right)^m e^{-p(s-m)}.$$

Substituting the bound on $p[A_{m,j}]$ in (44) into (36) gives:

$$(45) \quad S_2 \leq \sum_{m=\log_e n}^{(k/(k+1))n} \binom{n}{k+m} \left(\frac{sep}{m}\right)^m e^{-p(s-m)}$$

and, by inequality (5),

$$(46) \quad S_2 \leq \sum_{m=\log_e n}^{(k/(k+1))n} \left(\frac{ne}{k+m}\right)^{k+m} \left(\frac{sep}{m}\right)^m e^{-p(s-m)}.$$

Since $s = (n - (k + m))m$, inequality (46) becomes:

$$(47) \quad S_2 \leq \sum_{m=\log_e n}^{(k/(k+1))n} \left(\frac{ne}{k+m}\right)^{k+m} ((n - (k + m))ep)^m e^{-p(nm - mk - m^2 - m)}.$$

Now

$$(48) \quad \frac{ne}{k+m} < \frac{ne}{m},$$

and

$$(49) \quad n - (k + m) < n;$$

therefore,

$$(50) \quad S_2 \leq \sum_{m=\log_e n}^{(k/(k+1))n} \left(\frac{ne}{m}\right)^{k+m} (nep)^m e^{-pm(n-k-m-1)}.$$

The logarithm of an element in the sum is:

$$(51) \quad (k+m) \log_e \left(\frac{ne}{m}\right) + m(\log_e c_n + 1) - m(c_n - p(k+m+1)).$$

For $m \leq (k/(k+1))n$,

$$(52) \quad c_n - p(k+m+1) = O(c_n).$$

Therefore,

$$(53) \quad m[(\log_e c_n + 1) - (c_n - p(k+m+1))] = O(-mc_n).$$

Now, $c_n \geq \log_e n + w_n$, with $\lim_{n \rightarrow \infty} w_n = \infty$. Hence,

$$(k+m) \log_e \left(\frac{ne}{m}\right) - O(mc_n) \rightarrow -\infty$$

and the logarithm of S_2 is $O(\log_e n + m(\log_e(n/m) - c_n))$ which goes to $-\infty$ asymptotically with n . Hence S_2 diminishes to zero asymptotically as required. $\square$

LEMMA 3. S_3 tends to zero as n goes to infinity.

Proof.

$$(54) \quad S_3 = \sum_{m=(k/(k+1))n+1}^{n-k-2} \sum_{j=1}^{\binom{n}{k+m}} P[A_{m,j}]$$

where $A_{m,j}$ is the event that the canonical $(k+1)$ -cut constructed from cut (m, j) is not smaller than it.

We first observe that the main component is smaller than $n - (k/(k+1))n - k < n/(k+1)$ and that the size of the largest subcomponent, r , is larger or equal than the average number of vertices per subcomponent $(k+m)/k \geq n/(k+1) + 1$.

Therefore, in the cut (m, j) the main component is smaller than the largest subcomponent.

The largest subcomponent in cut (m, j) was a main component of some minimal noncanonical $(k+1)$ -cut, say (m^o, j^o) , for $m^o \leq (k/(k+1))n$.

We show that the size of cut (m, j) is larger or equal to the size of cut (m^o, j^o) .

To see this, first observe that cut (m, j) contains the set of edges between the main component and the subcomponents of cut (m^o, j^o) . (See Fig. 4). However, the number of edges between subcomponents of cut (m^o, j^o) , being a minimum such cut, is smaller or equal to the number of edges between the subcomponents and the main component of cut (m, j) . (See Fig. 4.)

The value m^o depends on the size of the largest subcomponent in cut (m, j) , r :

$$(55) \quad \frac{k+m}{k} = \frac{m}{k} + 1 \leq r \leq k+m - (k-1) = m+1.$$

Since $n - r = m^o + k$, we have

$$(56) \quad n - m - k - 1 \leq m^o \leq n - \frac{m}{k} - k - 1.$$

Here we need the following claim.

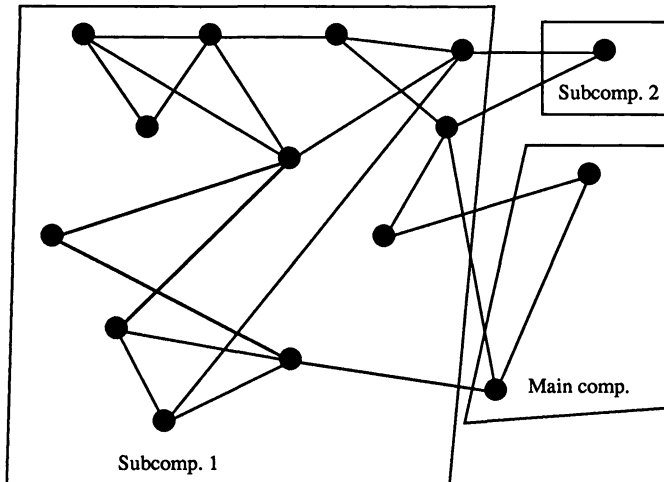


FIG. 4

CLAIM 3. If $1 \leq m \leq (k/(k+1))n$, then $P[A_{m,j}]$ is bounded by a function that decreases as m increases.

Proof. For $1 \leq m \leq \log_e n$ and $p \leq (\log_e n)^2/n$, (28) gives

$$\begin{aligned} P[A_{m,j}] &\leq \binom{k+m}{2} e^{-pn(k+m)+o(1)} \\ &< (k+m)^2 e^{-pn(k+m)+o(1)} \\ &= \exp \{ -mc_n + 2 \log_e (k+m) - kc_n + o(1) \} \end{aligned}$$

which is decreasing with m .

For the remaining range of p , i.e., $1 \leq m \leq \log_e n$ and $p > (\log_e n)^2/n$, (34) gives

$$(57) \quad P[A_{m,j}] \leq k' c_n^{k'} e^{-(k+m)c_n p((k+m)^2 + k')},$$

where

$$\begin{aligned} k' &= \frac{k+m}{m} \binom{k+m}{2} \\ &\leq \frac{(k+m)^3}{2m} \\ &< am^2 \text{ for some positive constant } a \\ &\leq am \log_e n \quad \text{since } m \leq \log_e n. \end{aligned}$$

Substituting k' by this bound in (57) and rearranging the terms of the right-hand side, we obtain

$$\begin{aligned} P[A_{m,j}] &\leq \exp \left\{ -m \left[c_n - a \log_e n \log_e c_n - 2a \frac{c_n}{n} \log_e n \right] + \log_e (am \log_e n) \right\} \\ &\leq \exp \{ -m[c_n - o(c_n)] + \log_e (am \log_e n) \} \end{aligned}$$

since $c_n > (\log_e n)^2$, which is decreasing as m increases.

For $\log_e n < m \leq (k/(k+1))n$, (44) gives

$$\begin{aligned} P[A_{m,j}] &\leq \left(\frac{sep}{m} \right)^m e^{-p(s-m)} \\ &= \exp \{ O(-mc_n) \} \text{ by (45)–(53)} \end{aligned}$$

which is decreasing as m increases. $\square$

By Claim 3, $P[A_{m^o,j^o}]$ is bounded by a function that is decreasing as m^a increases. Therefore, each of these functions is maximized when m^o assumes its smallest value, namely $n - m - k - 1$. As m increases from $(k/(k+1))n + 1$ to $n - k - 2$, the smallest values of m^o decrease from $n/(k+1) - k$ to 1. Lemma 1 and Lemma 2 proved that the sum of these functions for $m^o = 1, \dots, n/(k+1) - k$, and $j^o = 1, \dots, \binom{n}{k+m^o}$, is asymptotic to zero. This concludes the proof of Lemma 3 and Theorem 1. $\square$

4. A greedy heuristic. In this section, the results of the previous sections are converted to a linear algorithm for the k -cut on random graphs that gives the optimal solution with probability going to 1. Formally stating this, if a graph is drawn from $\mathcal{G}_{n,p}$ with

$$(\log_e n + w_n)/n \leq p \leq 1 - (w_n \log_e n)/n, \quad \lim_{n \rightarrow \infty} w_n = \infty,$$

then any k vertices of minimum degree are k components of a minimum $(k + 1)$ -cut with asymptotic probability 1.

Let $V = \{1, 2, \dots, n\}$ and d_i be the degree of vertex i . Without loss of generality, assume $d_1 \leq d_2 \leq \dots \leq d_n$.

The property that validates the algorithm derives from a result on a separation of the degree sequence in random graphs in this range of p [Bol85, Lemma 5, Chap. VII].

THEOREM 2. *If $(\log_e n + w_n)/n \leq p \leq 1 - (w_n \log_e n)/n$, $\lim_{n \rightarrow \infty} w_n = \infty$, then $K = \{1, 2, \dots, k\}$ are k components of a minimum $(k + 1)$ -cut with asymptotic probability 1.*

Proof. From Theorem 1, a minimum $(k + 1)$ -cut is canonical with asymptotic probability 1. Therefore, it remains to show that, for $(\log_e n + w_n)/n \leq p \leq 1 - (w_n \log_e n)/n$, $\lim_{n \rightarrow \infty} w_n = \infty$, the canonical $(k + 1)$ -cut, whose k isolated vertices are precisely the vertices of the set K , is minimum among all canonical $(k + 1)$ -cuts.

All possible $\binom{n}{k}$ canonical $(k + 1)$ -cuts are sorted such that $|C_1|, \leq \dots \leq |C_{\binom{n}{k}}|$. Let C^* be the canonical $(k + 1)$ -cut whose isolated vertices are K . The size of C_i , $i = 1, \dots, \binom{n}{k}$, is equal to the number of edges between single-vertex components, $\text{int}(C_i)$, plus the number of edges between the single-vertex components and the large component, $\text{ext}(C_i)$. The sum of the degrees of the isolated vertices in C_i is $2 \text{int}(C_i) + \text{ext}(C_i)$. The size of C^* is strictly larger than the size of C_1 if and only if

$$\text{int}(C^*) + \text{ext}(C^*) > \text{int}(C_1) + \text{ext}(C_1).$$

Since C^* consists of the minimum degree vertices,

$$2 \text{int}(C^*) + \text{ext}(C^*) = 2 \text{int}(C_1) + \text{ext}(C_1) - s, \quad \text{for some } s \geq 0.$$

Therefore, C^* is indeed larger than C_1 if and only if

$$(58) \quad \text{int}(C^*) < \text{int}(C_1) - s, \quad \text{for some } s \geq 0.$$

Since

$$0 \leq \text{int}(C_i) \leq \binom{k}{2}, \quad \forall i,$$

s can take only the following values:

$$0 \leq s \leq \binom{k}{2}.$$

Hence, if the sum of degrees of the isolated vertices in any canonical $(k + 1)$ -cut, other than C^* , exceeds the sum of degrees of vertices in K by at least $\binom{k}{2}$, then the theorem holds.

In Lemma 4, it is proved that, if $p = o(1)$, then $\text{int}(C_1) = 0$ with probability going to 1. Therefore, $\text{int}(C^*)$ cannot be strictly smaller than $\text{int}(C_1)$ and, by (58), the theorem holds for $p = o(1)$.

Lemma 5 shows that, if $(w_n \log_e n)/n \leq p \leq 1 - (w_n \log_e n)/n$, $\lim_{n \rightarrow \infty} w_n = \infty$, then

$$(59) \quad d_{k+1} - d_k \geq \binom{k}{2}$$

with probability going to 1. Hence the sum of vertex degrees of any set other than K exceeds the sum of vertex degrees of K by at least $\binom{k}{2}$ almost certainly, which concludes the proof of Theorem 2. $\square$

LEMMA 4. *If $p = o(1)$, then the number of edges between isolated vertices in a minimum canonical $(k + 1)$ -cut is asymptotically equal to zero.*

Proof. Let C_i be a canonical $(k + 1)$ -cut randomly selected. The probability that there are no edges between the k isolated vertices is the probability that the k isolated vertices form an independent set:

$$\begin{aligned} P[k \text{ random vertices are independent}] &= (1 - p)^{\binom{k}{2}} \\ &\geq 1 - \binom{k}{2}p \text{ by (7)} \\ &\rightarrow 1 \text{ if } p = o(1). \end{aligned}$$

Now, C_i contains stochastically more edges than C_1 , hence $\text{int}(C_i)$ is stochastically larger than $\text{int}(C_1)$.

LEMMA 5. *If $(w_n \log_e n)/n \leq p \leq 1 - (w_n \log_e n)/n$, $\lim_{n \rightarrow \infty} w_n = \infty$, then $\lim_{n \rightarrow \infty} P[d_{k+1} - d_k < \binom{k}{2}] = 0$.*

Proof. Lemma 5 follows directly from the following theorem [Bol85, p. 68]: If $m = o((pqn/\log_e n)^{1/4})$, $m \rightarrow \infty$ and $\alpha_n \rightarrow 0$ arbitrarily slowly, then almost every graph drawn from $\mathcal{G}_{n,p}$ is such that

$$(60) \quad d_{k+1} - d_k \geq \frac{\alpha_n}{m^2} \left(\frac{pqn}{\log_e n} \right)^{1/2} \quad \text{for every } k < m.$$

If $(w_n \log_e n)/n \leq p \leq 1 - (w_n \log_e n)/n$, $\lim_{n \rightarrow \infty} w_n = \infty$, then

$$\lim_{n \rightarrow \infty} \left(\frac{pqn}{\log_e n} \right) = \infty$$

and it suffices to choose m and α_n such that

$$(61) \quad \lim_{n \rightarrow \infty} \frac{\alpha_n}{m^2} \left(\frac{pqn}{\log_e n} \right)^{1/2} > \binom{k}{2}. \quad \square$$

Given a graph whose vertices are sorted by increasing degree, the running time of the algorithm is $O(k)$. Otherwise, we can find the k minimum degree vertices in linear time kn . Alternatively, we can improve the constant coefficient of n to $\log_2 k$ by maintaining a k -long array. In either case, the running time on a given graph is linear.

5. Summary. In this paper, it is proved that, when the graph is connected, the minimum k -cut has almost certainly $(k - 1)$ one-vertex components and a large component on the remaining vertices.

It is also proved that the $(k - 1)$ isolated vertices are precisely the ones with minimum degree in the graph. This last result holds for all connected graphs except when p , the probability of an edge, is very close to 1. An intuitive reason for this is that, when the graph is extremely dense, the vertex degrees are very intricate. As an example, if all vertices have degree $n - 1$, except for some which have degree $n - 2$, in order to find a minimum 3-cut, we need to pick, not any two vertices of degree $n - 2$, but two vertices of degree $n - 2$ that are connected.

The results of this paper yield a linear time algorithm to find the minimum k -cut for random graphs. This is extremely fast in the light of the best known worst-case bound which is $O(n^{k^2})$.

Remaining open questions include the structure of a k -cut when k increases with n or when each edge has a nonnegative weight. Since when k increases with n , the maximum

clique problem is reducible to the k -cut problem [GH88], any information about the structure of the k -cut for this case will imply related information about the structure of the maximum clique in random graphs.

REFERENCES

- [Bol85] B. BOLLOBÁS, *Random Graphs*, Academic Press, New York, 1985.
- [ER60] P. ERDÖS AND A. RÉNYI, *On the evolution of random graphs*, Publ. Math. Inst. Hung. Acad. Sci., 5 (1960), pp. 17–61.
- [GH88] O. GOLDSCHMIDT AND D. S. HOCHBAUM, *A polynomial algorithm for the k -cut problem*, Proceedings of the 29th Annual Symposium on Foundations of Computer Science, IEEE, October, 1988, pp. 444–451.
- [Lin65] D. V. LINDLEY, *Introduction to Probability and Statistics From a Bayesian Point of View*, Cambridge University Press, Cambridge, 1965.
- [Mit70] D. S. MITRINOVIC, *Analytic Inequalities*, Springer-Verlag, Berlin, New York, 1970.
- [Pal86] E. M. PALMER, *Graphical Evolution*, Academic Press, New York, 1986.
- [Pul82] W. PULLEYBLANK, Presentation at SIAM Meeting on Optimization, Boston, 1982.

BROOKS COLORING IN PARALLEL*

PÉTER HAJNAL† AND ENDRE SZEMERÉDI‡

Abstract. A theorem of Brooks guarantees that a maximum-degree- D graph can be properly colored with D colors if the graph does not contain a large complete subgraph. It is proved that finding such a coloring is in NC.

Key words. parallel computation, NC, PRAM model, Brooks theorem

AMS(MOS) subject classifications. primary 68Q20; secondary OSC15, 05-04, 68R10

1. Introduction. It is very easy to show that any graph can be vertex colored with a number of colors at most one greater than its maximum degree. The proof gives a very simple sequential algorithm, also. Usually though, colorings with fewer colors exist.

THEOREM 1.1. (Brooks, see [Lo]). *A simple graph G of maximum degree D can be colored with D colors if and only if either $D \geq 3$ and G contains no clique on $D + 1$ nodes, or $D \leq 2$ and G has no odd cycle.*

The proofs in [Lo] and [Be], although not trivial, are algorithmic. Converting the standard algorithms to parallel algorithms seems difficult. (We use the CRCW PRAM model for parallel computation. A PRAM is a collection of synchronous parallel processors sharing a global memory. The CRCW model is one in which two different processors can write in the same cell if they want to write the same value. For precise definitions, see [AHU].) Luby remarked that from the maximal independent set algorithm ([KW], [Lu]), it is easy to get an NC algorithm that colors a graph with $D + 1$ colors where D is the maximum degree of the graph. He defined the following general problem and stated the theorem above in a more general form. The general question is the maximal coloring problem: The input consists of a graph, and associated with each vertex is a set of allowable colors. The answer is a partial coloring in which every point is colored with one color from its given color set in such a way that the output coloring cannot be extended to additional nodes (while still insisting that each node's color is chosen from its color set). By reducing the maximal coloring problem to the maximal independent set problem, Luby (in [Lu]) exhibited an NC algorithm to find a maximal coloring. From this algorithm we can easily conclude that we can find a maximal extension of a given coloring in NC, where a *maximal extension* of a coloring c is a coloring c' that uses the same number of colors and for which each uncolored node is adjacent to a point of each color.

(To do so we apply Luby's maximal coloring algorithm on the graph induced by points not colored by c and the allowable color set at point v is the set of colors not represented on v 's neighbors.) We will apply this form of Luby's algorithm so we state this in a separate lemma.

THEOREM 1.2. [Lu]. *Given a graph with a partial coloring, we can find a maximal extension of it in NC.*

Another expression of this result is that the following NC procedure exists.

* Received by the editors June 15, 1987; accepted for publication (in revised form) March 9, 1989.

† Department of Computer Science, University of Chicago, Chicago, Illinois 60637 and Department of Computer Science, Eötvös Loránd University, Budapest, Hungary. The work of this author was supported by the National Science Foundation under grant NSF 5-27561 and by the Air Force Office of Scientific Research under contract OSR-86-0076. Present address, Department of Computer Science, Princeton University, Princeton, New Jersey 08544.

‡ Department of Computer Science, University of Chicago, Chicago, Illinois 60637 and Hungarian Academy of Sciences, Budapest, Hungary. Present address, Department of Computer Science, Rutgers University, New Brunswick, New Jersey 08903.

PROCEDURE EXTEND (S)

Given: A graph G , partial coloring c with exactly t colors, and a subset S of the points. Let C be the set of colored vertices.

Compute: A partial coloring c' , using exactly t colors, that is an extension of c . Nodes not in $C \cup S$ are still uncolored, and every uncolored point in S is adjacent to a point of each of the t colors.

Luby's maximal coloring algorithm did not settle the question of finding a Brooks coloring. In his thesis [Ka1], Karloff gave an NC algorithm for the case when the maximum degree is three. Using this result and the fact that the case $D \leq 2$ is easy, we can assume that $D \geq 4$. Our result is summarized in the following theorem.

THEOREM 1.3. *For a graph G of maximum degree $D \geq 3$ and G containing no copy of K_{D+1} , we can construct a coloring with D colors in $O(\log^4 n)$ time on a CRCW PRAM using n^4 processors (where n is the number of nodes in G).*

We use the usual graph-theoretical notations that we can find in [Lo]. We mention that G will mean a simple graph, $G(A)$ will denote the subgraph of G induced by a vertex set A , and K_i will denote the complete graph on i vertices. A k -coloring is a function $c: V(G) \rightarrow \{1, 2, \dots, k\} \cup \{*\}$ for which no two adjacent nodes have the same integer label. The set of nodes with image $*$ is the set of uncolored points.

2. Modification of the uncolored part.

2.0. Outline of the algorithm. Using the procedure *Extend*, we can find a $D - 3$ -coloring such that every uncolored vertex is adjacent to a node of each color. Since we seek a D coloring, we can assume that our coloring uses exactly $D - 3$ colors; from this point until § 2.3 we assume that our coloring uses exactly $D - 3$ colors. From the maximality it follows that the uncolored graph's maximum degree is at most 3. This is not enough for us—if this part contains a K_4 as a subgraph, then we cannot extend our coloring to a total D -coloring of the whole graph. In this section we resolve this problem by modifying the original coloring.

Denote the color classes by C_i ($i = 1, \dots, D - 3$) and the set of uncolored points by U . As stated above, it is easy to see that in $G(U)$ every vertex has degree at most 3. The definition of maximal coloring gives a stronger result, namely that every point u of degree 3 in $G(U)$ has exactly one neighbor in every C_i . For every point of degree two in $G(U)$, there are two possibilities: it has exactly one neighbor in every C_i or there is exactly one exception where it has two. These remarks are true not only for the initial coloring but for every maximal coloring. CN_u will denote the colored neighborhood of the uncolored point u : $CN_u = \{\text{colored nodes adjacent to } u\}$. (The definitions of this paragraph depend on the current coloring. During the algorithm we will be changing it many times.)

To color U with three colors we need to get rid of K_4 's in $G(U)$. We can get rid of an uncolored K_4 by coloring one of its vertices. Using this method, we will get an edge inside a color class, so we have to remove the color from its other endpoint. If we are unlucky then we can create a new K_4 in $G(U)$. In this case we must continue this procedure until we reach another configuration where we have the hope of using the color extension theorem. So the modification goes along a path that alternates between colored points and uncolored triangles. The procedure that finds this path, and the method that cleans up after every exchange phase are described in § 2.2. Finishing the coloring is in § 2.3. Before we start with the whole method we must overcome some problems. If we have a K_4 in $G(U)$ all of whose vertices have the same colored neighborhood then our method will get stuck in an infinite loop. To avoid this situation we need the procedure that is in § 2.1.

2.1. Elimination of bad K_4 's. The last sentences of the previous paragraph suggest the following definition of "bad" K_4 .

DEFINITION. A K_4 A in $G(U)$ is a *bad* K_4 if and only if for all $v, u \in A$, $CN_u = CN_v$. If a K_4 is not bad then we say that it is *good*. If A is a bad K_4 then its common colored neighborhood contains two nonadjacent vertices, as otherwise G would have K_{D+1} as a subgraph. Using these two points and two points from the K_4 we can perform an exchange procedure that will improve our situation. The parallel implementation is the following procedure.

PROCEDURE ELIMINATE

Given: A graph G with maximum degree $D \geq 4$, not having K_{D+1} as a subgraph, and a maximal $D - 3$ -coloring of G .

Compute: A new coloring of G , not having any bad K_4 's, for which every uncolored K_4 was also uncolored when the procedure was called.

Algorithm:

1) Let U be the set of uncolored nodes.

Let $A_1, \dots, A_r$ be the bad K_4 's in $G(U)$.

Let $A_{r+1}, \dots, A_t$ be the good K_4 's in $G(U)$.

2) Let U' be the empty set.

For all i , $1 \leq i \leq r$, do in parallel:

Let CN_i = the common colored neighborhood of nodes in A_i .

CN_i does not induce a clique, so there exist $u_i \neq v_i \in CN_i$ that are nonadjacent. Add u_i, v_i to U' .

Let $x_i \neq y_i$ be in A_i (arbitrarily). Color x_i with $c(u_i)$ and y_i with $c(v_i)$. Uncolor u_i and v_i .

(It is worth noting that several processors may simultaneously be uncoloring a node.) U' consists of the newly uncolored nodes. The new coloring is proper because every new node with the same color came from a different component of $G(U)$ and its neighbor in this color class lost its color.

3) Call procedure *Extend* ($A_{r+1} \cup \dots \cup A_t$).

4) Call procedure *Extend* (U').

5) Call procedure *Extend* ($V(G)$).

End *Eliminate*.

The following claim shows that we achieved our goal.

Claim 2.1. When *Eliminate* terminates, the $D - 3$ -coloring of G is maximal, every uncolored K_4 was also uncolored when procedure *Eliminate* was called, and there are no bad K_4 's.

Proof. That the coloring of G is maximal is obvious. Assume K is a currently uncolored K_4 that was not completely uncolored when procedure *Eliminate* was called. Hence, $z \in K$ lost its color during execution of step 2 of procedure *Eliminate*.

Case (a). $z \in CN_i \cap CN_j$ for $1 \leq i < j \leq r$. So z is adjacent to every node in A_i and every node in A_j . So after step 2, z has at least 4 uncolored neighbors, none of which is colored in step 3. Hence step 4 colors z , a contradiction.

Case (b). $z \in CN_i$ for exactly one i , $1 \leq i \leq r$. The same argument shows that after step 4, z has exactly 2 uncolored neighbors in A_i . Because no node in a bad K_4 had an uncolored neighbor outside of that K_4 , the 4th node w in K must have been colored before procedure *Eliminate* was called. So $w \in CN_j$ for some $1 \leq j \leq r$. If $j \neq i$, w is adjacent to two uncolored nodes in A_j just before step 4 is run. So step 4 will color w . If $i = j$, $\{w, z\} = \{u_i, v_i\}$, yet u_i and v_i are nonadjacent, a contradiction.

The only thing remaining to show is that there are no bad K_4 's. Because all the currently uncolored K_4 's were good when the procedure was called, it is enough to show that the corresponding colored neighborhoods didn't change. This is clear from the fact that otherwise step 3 would have colored the node in the K_4 .

2.2. The alternating paths. After executing *Eliminate* we can execute our procedure that decreases the number of uncolored K_4 's by a constant fraction.

PROCEDURE MODIFY_COLORING

Given: A graph G of maximum degree D , and a maximal $D - 3$ -coloring without bad K_4 's. Let U be the set of uncolored nodes.

Compute: A maximal coloring of G for which, if U' is the set of uncolored nodes,

$$[\# \text{ of } K_4 \text{'s in } G(U')] \leq \frac{3}{4} [\# \text{ of } K_4 \text{'s in } G(U)].$$

Algorithm

1) Let $\bar{Q} = \{K_4 \text{'s in } G(U)\}$.

Let Q be the set of nodes in $\bar{Q}$. (Q stands for “quadrilaterals.”)

Let $\bar{T} = \{K_3 \text{'s that are components in } G(U)\}$. (T stands for “triangles.”)

Let T be the set of nodes in $\bar{T}$.

For $1 \leq i \leq D - 3$, let $\bar{T}_i = \{K_3 \text{'s in } \bar{T} \text{ such that each node in the } K_3 \text{ has two neighbors colored } i\}$. (Note that each node in such a K_3 has two neighbors colored i and exactly one colored j , for all $j \neq i$.)

Let $\bar{T}_0 = \bar{T} - (\bar{T}_1 \cup \dots \cup \bar{T}_{D-3})$.

For $0 \leq i \leq D - 3$, let T_i be the set of nodes in $\bar{T}_i$.

Let $R = U - (Q \cup T)$. (R —which stands for “remainder”—is the part of the set of uncolored nodes lying in components that are neither K_4 's nor K_3 's.)

Let C_i be the set of nodes colored i .

2) For $1 \leq i \leq D - 3$ build an auxiliary digraph H_i on $C_i \cup \bar{T}_i$ as follows:

Start with no arcs.

For each triangle $L \in \bar{T}_i$, choose a representative node.

For all $u \in C_i$ and $t \in \bar{T}_i$, add arc (u, t) if and only if u is adjacent to every node in t , and there is no $t' \in \bar{T}$, $t' \neq t$, every node of which is adjacent to u .

Then, for each arc $(u, t) \in E(H_i)$, add to $E(H_i)$ an arc from t to the node $v \in C_i$, that is different from u and adjacent to t 's representative. The outdegrees of nodes in C_i are at most one. A node t in $\bar{T}_i$ either has indegree = outdegree ≤ 1 , or there are two nodes $a \neq b \in C_i$, and the only arcs in H_i incident with t are (a, t) , (t, b) , (b, t) , (t, a) .

Motivation: If we remove the color of a node colored by i then this node may create a new uncolored K_4 with a previously uncolored triangle. If this triangle doesn't belong to $\bar{T}_i$ then we can handle the problem by using the coloring extension procedure. Otherwise we must continue the exchange method. We color the representative of this triangle with color i and uncolor its neighbor in C_i . The edges of H_i tell the exchange procedure what to do next after we remove a color or color a representative.

2a) We need the following notation. Say $v \in C_i$ is an *endpoint* if and only if v 's outdegree in H_i is 0, and an *inner point* otherwise.

3) For each $K \in \bar{Q}$ (K is a K_4), choose a colored vertex v such that v has 1, 2 or 3 neighbors in K —such a vertex always exists, because K is a good K_4 —and if there are several such points choose one among them that has the most neighbors in K .

(Why we choose the vertex with the most neighbors in K will become clear only at the end of the proof of Lemma 2.2.1.) Node v is an *initial point with respect to* K . It is the *initial point* if and only if it is an initial point with respect to some K . Where v is the initial point with respect to K , choose one of v 's neighbors in K . We will refer to this chosen point as K 's *representative*.

Let $\bar{Q}_i = \{K \in \bar{Q} \mid \text{the initial point with respect to } K \text{ lies in } C_i\}$.

Label K with “color the representative with color i ” where $K \in \bar{Q}_i$.

4) For each i , label $x \in C_i$ with “remove color” if and only if there exists a path in H_i from an initial point in C_i to x , and label $y \in \bar{T}_i$ with “color the representative with color i ” if and only if there exists a path in H_i from an initial point in C_i to y .

5) For each i in parallel do:

Uncolor each node labeled “remove color.”

Next, color with color i the representative of any K_3 or K_4 labeled with “color the representative with color i .”

Note that the coloring remains proper.

6) For each i , $1 \leq i \leq D - 3$, do:

Let $\bar{E}_i = \{T - (\text{representative of } T) \mid T \text{ was a labeled } K_3 \text{ in } \bar{T}_i\}$. ($\bar{E}_i$ consists of the uncolored K_2 's that remain after each labeled K_3 's representative is colored.)

Let $\bar{T}'_i = \{\text{unlabeled } T \in \bar{T}_i\}$.

Let $\bar{Q}'_i = \{(K - \text{representative of } K) \mid K \in \bar{Q}_i\}$.

Let E_i, T'_i, Q'_i be the respective sets of points.

Let $N_i = \{\text{labeled vertices in } C_i\}$. (The N stands for “newly uncolored.”) The nodes in N_i can be either inner points or endpoints. It is easy to see that $[\# \text{ of endpoints in } N_i] \leq |\bar{Q}_i|$.

Note that the uncolored vertices of G are exactly $R \cup T_0 \cup \{\cup_i [E_i \cup T'_i \cup Q'_i \cup N_i]\}$.

7) For each $L \in \cup_i (T'_i \cup \bar{Q}'_i \cup \bar{E}_i)$, find a vertex in L , if it exists, not having a neighbor of every color (find only one, even if 2 or 3 exist). Color that vertex with some color not occurring on its neighbors. Note that the coloring remains proper.

8) Apply procedure *Extend* ($\cup_i N_i$).

Now every completely uncolored K_4 K either contains 2 endpoints, or contains one endpoint v and $K - \{v\} \subset R$. We prove this fact below. (If the first case held for every completely uncolored K_4 , we would have succeeded in halving the number of uncolored K_4 's.)

9) For each completely uncolored K_4 K containing exactly one endpoint v (so that $K - \{v\} \subset R$), choose one node $y \in K - \{v\}$ with 4 or more uncolored neighbors. (This y must exist.) Let $Y \subset R$ be the set of chosen y 's. $G(R)$ has maximum degree at most 3. Find a maximal independent set of $G(Y)$. (Because of the degree bound on $G(R)$ the cardinality of this independent set must be at least $|Y|/(4)$.) Color each node in the independent set with some color not occurring on its neighborhood.

10) Apply procedure *Extend* (V).

End *Modify_Coloring*.

In order to prove the correctness of the algorithm we must first prove the lemma mentioned under Step 8.

LEMMA 2.2.1. *Just after Step 8 of Modify_Coloring, every completely uncolored K_4 K either contains two endpoints, or contains one endpoint v and $K - \{v\} \subset R$.*

Proof. Each uncolored K_4 K , just after Step 8, must fall into (at least) one of the following classes.

- (a) K contains exactly one newly uncolored node z , and z is an inner point.
- (b) K contains at least two inner points.
- (c) K contains exactly one endpoint and exactly one inner point.
- (d) K contains exactly one endpoint, and the remaining nodes are in T .
- (e) K contains exactly one endpoint, and the remaining nodes are in Q .
- (f) K contains exactly one endpoint, and the remaining nodes are in R .
- (g) K contains at least 2 endpoints.

We now prove that cases (a)–(e) cannot occur. We will be making heavy use of the following two simple facts.

Remark 1. Just after Step 5, every inner point originally colored with color i has, in its uncolored neighborhood, an edge in E_i . At least one of these points remains uncolored after Steps 7 and 8 are executed.

Remark 2. Immediately after Step 8 every newly uncolored node in an uncolored K_4 has exactly $D - 3$ colored neighbors, and its three uncolored neighbors are exactly the remaining nodes of the K_4 .

Case (a) cannot occur, because if z is the only newly uncolored node in an uncolored K_4K , then z 's uncolored neighbors span a K_3 , yet by Remarks 1 and 2, this cannot occur.

If, instead, any uncolored K_4K contained distinct inner points u, v just after Step 8, u and v must have originally had colors i and j , respectively, $i \neq j$. As for case (a), by Remarks 1 and 2, after Step 8 u has one uncolored neighbor from E_i in K . The same goes for v , with " E_i " replaced by " E_j ." This is a contradiction, because no node in E_i is adjacent to a node in E_j . Therefore, case (b) is impossible.

Now we discuss case (c). Let x be K 's lone endpoint, and y its lone inner point. Node x originally has color i , and node y has color j . Again, $i \neq j$. Let z and t be the remaining two vertices of K . By Remarks 1 and 2, z or t came from E_j . In fact, both must be in E_j . By the definition of T_j , z and t have exactly one neighbor colored i , namely x . Step 5 uncolored x so that after Step 5, z and t had no neighbor colored i . This is a contradiction, since Step 7 will color either z or t .

Let K be a case (d) K_4 ; z , its lone endpoint; and i , its original color. If $K - \{z\} \subset T_j, j \neq i$ (possibly $j = 0$), for one of those three nodes w , z was w 's only neighbor colored i (otherwise K would be in T_i). After color i is removed from z , Step 7 will color at least one of those three nodes. Hence the other three nodes are in T_i . Since z is an endpoint, z had to have three pairwise adjacent neighbors, none of which is in K , and all of which are in T . At least one of these, s , will remain uncolored after Step 7. Hence, after Step 7, s and all of K are still uncolored, and Step 8 will color z .

Let K be a case (e) K_4 ; z , its lone endpoint; and i , its original color. If the other three nodes are in $Q_j, j \neq i$, then the argument in case (d) produces a contradiction. So $K - \{z\} \subset A$, where A is a K_4 in $\bar{Q}_i$. By the method used to choose an initial point in Step 3, z must be the initial point with respect to A , and therefore some node in $K - \{z\}$ is A 's representative and is colored in Step 5.

Using this lemma the proof of correctness is easy.

LEMMA 2.2.2. *The procedure Modify_Coloring decreases the number of uncolored K_4 's to at most $\frac{3}{4}$ of its previous value.*

Proof. All the uncolored K_4 's existing when the algorithm terminated were uncolored after Step 8. Using Lemma 2.2.1 we divide the uncolored K_4 's into two classes. This implies a classification of endpoints: in an uncolored K_4 an endpoint is *ugly* if immediately after Step 8 it is the lone endpoint of its uncolored K_4 , and *nice* otherwise. Using this notation it is easy to see the following inequalities.

$$\begin{aligned} (\# \text{ of old uncolored } K_4\text{'s}) &\geq (\# \text{ of endpoints}) \\ &= (\# \text{ of ugly endpoints}) + (\# \text{ of nice endpoints}). \end{aligned}$$

But we know that

$$\frac{3}{4}(\# \text{ of ugly endpoints}) \geq (\# \text{ of uncolored } K_4 \text{ having exactly one endpoint}),$$

and

$$\frac{1}{2}(\# \text{ of nice endpoints}) \geq (\# \text{ of uncolored } K_4 \text{ having at least two endpoints}).$$

So

$$\frac{3}{4}(\# \text{ of old uncolored } K_4\text{'s}) \geq (\# \text{ of uncolored } K_4\text{'s}).$$

2.3. Finishing the coloring. Using the previous result we can find in NC a partial coloring with $D - 3$ colors such that the uncolored part has maximum degree 3 and doesn't contain any K_4 's. So completing the coloring involves solving the original question on maximum-degree-3 graphs, which is an easier problem. We use Karloff's NC algorithm [Ka1] for 3-coloring a maximum-degree-3 graph without K_4 's. So the final algorithm is the following:

PROGRAM BROOKS_COLORING

Given: A graph G with maximum degree $D \geq 4$, not having K_{D+1} as a subgraph.

Compute: A D -coloring of G .

Algorithm:

- 1) Using procedure *Extend* get an initial maximal $D - 3$ -coloring.
- 2) Call procedure *Eliminate*.
- 3) Call procedure *Modify_Coloring*.
- 4) If there are four pairwise-adjacent uncolored nodes then go to 2.
- 5) Color the set of uncolored nodes with 3 new colors.

End Brooks_Coloring

The correctness of the algorithm easily follows from the lemma.

3. Conclusion. The main result of this paper is an application of the maximal coloring algorithm. The essence of our Brooks coloring algorithm is a procedure designed to get rid of the troublesome K_4 's, and we perform a natural exchange procedure along alternating paths to do so. The extension lemma takes care of the messy uncolored part after the exchange. The natural question is whether there are further applications of the maximal coloring algorithm.

Recently Karloff [Ka2] and Karchmer and Naor [KN] have announced alternate NC algorithms for Brooks' theorem.

REFERENCES

- [AHU] A. AHO, J. HOPCROFT, AND J. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.
- [Be] C. BERGE, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973.
- [Ka1] H. KARLOFF, *Fast parallel algorithms for graph-theoretic problems: matching, coloring and partitioning*, Ph.D. Thesis, Berkeley, 1985.
- [Ka2] ———, *An NC algorithm for Brooks' theorem*, Theoret. Comput. Sci., to appear.
- [KN] J. NAOR AND M. KARCHMER, *A fast parallel algorithm to color a graph with Δ colors*, J. Algorithms, to appear.
- [KW] R. M. KARP AND A. WIGDERSON, *A fast parallel algorithm for the maximal independent set problem*, Proc. 16th ACM Symposium on Theory of Computing, 1984, pp. 266–272.
- [Lo] L. LOVÁSZ, *Combinatorial Problems and Exercises*, North Holland, Amsterdam, 1979.
- [Lu] M. LUBY, *A simple parallel algorithm for the maximal independent set problem*, Proc. 17th ACM Symposium on Theory of Computing, 1985, pp. 1–10.

AN ALMOST LINEAR TIME ALGORITHM FOR GENERALIZED MATRIX SEARCHING*

MARIA M. KLAWE† AND DANIEL J. KLEITMAN‡

Abstract. An $O(m\alpha(n) + n)$ time algorithm is given for finding row-maxima and minima in totally monotone partial $n \times n$ matrices. As a result, faster algorithms are obtained for some optimization problems concerning distance and visibility between vertices of two convex polygons. Also shown is how the algorithm can be modified to give an $O(n\alpha(n))$ algorithm for a class of dynamic programming problems satisfying convex quadrangle inequalities. This results in faster algorithms for a number of problems arising in molecular biology, speech recognition, and geology.

Key words. monotone matrix, algorithm, dynamic programming, Ackermann's function

AMS(MOS) subject classifications. 68C05, 68E05, 90C39, 52A40

1. Introduction. A matrix $M = (M_{ij})$ is *totally monotone* if for every i, j, k, l such that $i < k, j < l$, and $M_{ij} \leq M_{il}$, we have $M_{kj} \leq M_{kl}$. A falling staircase matrix is a lower triangular fragment of a totally monotone matrix. More precisely,

$$(M, \{f(i) : 0 \leq i \leq n+1\})$$

is an $n \times m$ *falling staircase matrix* if

(a) for $i = 0, \dots, n+1$, $f(i)$ is an integer with $0 = f(0) < f(1) \leq f(2) \leq \dots \leq f(n) < f(n+1) = m+1$.

(b) M_{ij} is a real number if and only if $1 \leq i \leq n$ and $1 \leq j \leq f(i)$. Otherwise, M_{ij} is blank.

(c) $i < k, j < l \leq f(i)$, and $M_{ij} \leq M_{il}$, we have $M_{kj} \leq M_{kl}$.

Similarly, $(M, \{f(i) : 0 \leq i \leq n+1\})$ is an $n \times m$ *rising staircase matrix* if

(d) for $i = 0, \dots, n+1$, $f(i)$ is an integer with

$$0 = f(n+1) < f(n) \leq f(n-1) \leq \dots \leq f(1) < f(0) = m+1.$$

(e) M_{ij} is a real number if and only if $1 \leq i \leq n$ and $1 \leq j \leq f(i)$. Otherwise, M_{ij} is blank.

(f) $i < k, j < l \leq f(k)$, and $M_{ij} \leq M_{il}$, we have $M_{kj} \leq M_{kl}$.

The sequence $\{f(i)\}$ is called the *boundary sequence* of the staircase matrix. When the boundary sequence is clear we will simply use M to denote the staircase matrix. Rising and falling staircase matrices are illustrated in Fig. 1.

Totally monotone matrices were introduced by Aggarwal et al. in [AKMSW87], who showed that a wide variety of problems in computational geometry could be reduced to the problem of finding row-maxima or row-minima in totally monotone matrices. Aggarwal et al. [AKMSW87] give a simple $O(m+n)$ algorithm for finding row-maxima in totally monotone $n \times m$ matrices. It is easy to see that if we negate the entries of a totally monotone matrix and reverse the order of the rows, we obtain another totally monotone matrix. Thus any row-maxima finding algorithm can be trivially converted

* Received by the editors June 15, 1988; accepted for publication (in revised form) May 1, 1989.

† Department of Computer Science, University of British Columbia, Vancouver, British Columbia, Canada V6T 1W5.

‡ Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139. This research was partially supported by National Science Foundation grant DMS 86-06225 and Air Force Office of Scientific Research grant OSR-86-0078.

3	1	2								
1	2	3								
1	2	3	2	0						
1	2	3	3	1	0					
1	2	3	4	5	1					
1	2	3	4	5	6					
1	2	3	4	5	6	0	1	2		

FIG. 1(a). *Falling staircase matrix.*

4	3	2	1	0	1	2	3			
3	2	1	0	1	2	3				
3	2	2	1	1	1					
1	2	3	1							
1	2	3	4							
1	2	3								

FIG. 1(b). *Rising staircase matrix.*

to a row-minima finding algorithm and vice versa. Similarly, the problem of row-minima finding in falling staircase matrices is equivalent to row-maxima finding in rising staircase matrices. Following Wilber [W88] we will refer to the [AKMSW87] linear time algorithm for finding row-minima in totally monotone matrices as the SMAWK algorithm.

In [AK87] Aggarwal and Klawe introduce staircase matrices and show that if P and Q are nonintersecting n and m vertex convex polygons, respectively, then the problem of finding, for each vertex of P , the farthest (nearest) vertex of Q which is invisible to it, can be reduced to the problem of finding the row-maxima of rising staircase matrices. It is not hard to see that any staircase matrix can be “completed” to form a totally monotone matrix by replacing the blank entries by appropriate real numbers. An example is shown in Fig. 2. However, it is generally not possible to “complete” a rising (falling) staircase matrix without introducing new row-maxima (row-minima), and hence we cannot use this technique for finding row-maxima (row-minima) in rising (falling) staircase matrices. Aggarwal and Klawe also give an $O(m \log \log n)$ time algorithm for finding row-maxima in rising staircase matrices of size $n \times m$. This paper gives a somewhat more complicated algorithm with $O(m\alpha(n) + n)$ running time, where $\alpha(n)$ is a very slowly growing function often called the inverse of Ackermann’s function. We will give a precise definition of $\alpha(n)$ in the next section, immediately preceding Theorem 2.6.

We will actually give an equivalent result, namely an $O(m\alpha(n) + n)$ time row-minima finding algorithm for falling staircase matrices, since the notation is slightly simpler for this version. From now on all staircase matrices will be falling staircase matrices so we will refer to them simply as staircase matrices.

Our row-minima finding algorithm can also be modified to give a fast solution for some dynamic programming problems whose weight functions satisfy an inequality known

3	1	2								
1	2	3								
1	2	3	2	0						
1	2	3	3	1	0					
1	2	3	4	5	1					
1	2	3	4	5	6					
1	2	3	4	5	6	0	1	2		

3	1	2	-1	-2	-3	-4	-5	-6		
1	2	3	-1	-2	-3	-4	-5	-6		
1	2	3	2	0	-1	-2	-3	-4		
1	2	3	3	1	0	-1	-2	-3		
1	2	3	4	5	1	-1	-2	-3		
1	2	3	4	5	6	-1	-2	-3		
1	2	3	4	5	6	0	1	2		

FIG. 2. *Completing a staircase matrix.*

as the convex quadrangle inequality. The convex quadrangle inequality, and its opposite, the concave quadrangle inequality, were introduced by Yao in [Y82], who showed that certain dynamic programming problems, which have cubic time solutions using the usual methods, could be solved in quadratic time if their weight functions satisfied a quadrangle inequality and some fairly natural other conditions. In [HL87], Hirschberg and Larmore obtained a similar result, showing that a slightly different class of dynamic programming problems, which have quadratic time solutions via standard techniques, could be solved in $O(n \log n)$ time if the weight functions satisfied the concave quadrangle inequality. Hirschberg and Larmore also described a number of applications where their class of problems arise including text formatting and data structures. Similarly, Eppstein, Galil, and Giancarlo [EGG88] give an $O(n \log n)$ algorithm for a class of dynamic programming problems whose weight functions satisfy the convex quadrangle inequality, and describe a number of applications in biology, speech recognition, and geology where such problems arise. Since the quadrangle inequalities imply total monotonicity, it is not too surprising that fast algorithms for searching totally monotone matrices can be helpful in solving these types of dynamic programming problems. In particular, Wilber [W88] used the SMAWK algorithm in a very clever way to obtain a linear time algorithm for the problems presented by Hirschberg and Larmore, and by modifying our row-minima finding algorithm for staircase matrices, we obtain an $O(n\alpha(n))$ algorithm for the Eppstein–Galil–Giancarlo dynamic programming problems. This is done in § 4, where we introduce a generalization of the Eppstein–Galil–Giancarlo dynamic programming problems called the dynamic matrix searching problem and show how our algorithm can be modified to solve it.

2. A bound on the comparisons needed to find row-minima. We will use the term *row-minimum* of a row R in a staircase matrix to mean the minimum nonblank value in R . If there is more than one minimum value, we will take it to be the leftmost minimum value. We will use the term *processing a staircase matrix* to mean finding the positions and values of its row-minima.

Let $(M, \{f(i)\})$ be a staircase matrix. We will call row i a *step-row* of M if $i > 0$ and $f(i) < f(i + 1)$, and will say that M has t *steps* if it has exactly t step-rows. Note that by the definition in § 1 (a) of a boundary sequence $\{f(i)\}$, the bottom row is always a step-row. Thus a totally monotone matrix is a staircase matrix with one step, and Fig. 1 (a) shows a staircase matrix with four steps. We will call the staircase matrix consisting of the step-rows of M the *step-row matrix* of M , and denote it by $S(M)$. The step-row matrix of the staircase matrix in Fig. 1 (a) is shown in Fig. 3.

We say that a staircase matrix M has *shape* (t, n, m) if M has at most t steps, at most n rows, and at most m columns. Let $q(t, n, m)$ be the worst-case amount of time needed to process a staircase matrix of shape (t, n, m) . We will prove a number of results about the function $q(t, n, m)$ which will eventually yield the desired row-minima finding algorithm. To make the underlying structure of the algorithm clearer, in this section we

1	2	3							
1	2	3	2	0					
1	2	3	4	5	6				
1	2	3	4	5	6	0	1	2	

FIG. 3. The step-row matrix for 1(a).

will ignore the time needed to maintain data-structures, and instead concentrate on bounding the comparisons required. We use $q_c(t, n, m)$ to denote the worst-case number of comparisons which must be made to determine the row-minima of staircase matrices of shape (t, n, m) . In § 3 we will describe the data-structures used by the algorithm and prove that the time spent by our algorithm in maintaining these data-structures is of the same order as the number of comparisons made.

For $i \leq k$ and $j \leq l$, we will use the notation $M[i, k; j, l]$ to denote the staircase matrix obtained by taking the intersection of rows $i, \dots, k$ of M with columns $j, \dots, l$ of M . For any positive integer a , we define the *stepsize a approximation* of M to be the submatrix of M with boundary sequence $f_a(i)$, where $f_a(i)$ is defined as $f_a(i) = f(\lfloor i/a \rfloor a)$. We denote the stepsize a approximation of M by M_a . An example is illustrated in Fig. 4. Although the first $a - 1$ rows of M_a are blank, we will always treat these as rows in M_a so that M_a will have the same set of rows as M though the lengths (of the nonblank parts) of the rows will be different. Thus M_a is a staircase matrix of shape $(\lfloor n/a \rfloor, n, m)$. We define the *a -border matrices of M* , which we denote by $M(a, i)$, for $i = 1, \dots, \lceil n/a \rceil$, by $M(a, i) = M[(i - 1)a + 1, \min((i - 1)a + 1, n)]$. An example of a -border matrices is also illustrated in Fig. 4. We will denote the set of a -border matrices of M by $\Gamma(M, a)$. Clearly each a -border matrix has at most $a - 1$ rows. It is easy to see that the staircase matrices $M_a, M(a, 1), \dots, M(a, \lceil n/a \rceil)$ disjointly cover the nonblank entries of M .

The stepsize approximation and border matrices are essential for our $O(m\alpha(n) + n)$ algorithm which recurses on a function involving the ratio of number of steps to number of rows. This is because the stepsize a approximation M_a has the same number of rows as M but fewer steps. Moreover, if we find the row-minima in M_a and the $M(a, i)$ we can easily determine the row-minima in M . The actual recurrence, given in Theorem 2.6 and Corollary 2.7, is somewhat complicated, so we start with some very simple lemmas and immediate corollaries. Most of these preliminary observations are in [AKMSW87] or [AK87], but we include them here for the sake of completeness and clarity of presentation.

LEMMA 2.1. *For any positive integer a , we have*

$$q_c(n, n, m) \leq q_c(n/a, n, m) + O(am + n).$$

Proof. Let M be a staircase matrix of shape (n, n, m) . Since the stepsize a approximation M_a is of shape $(n/a, n, m)$, it can be processed in $q_c(n/a, n, m)$ time. Since

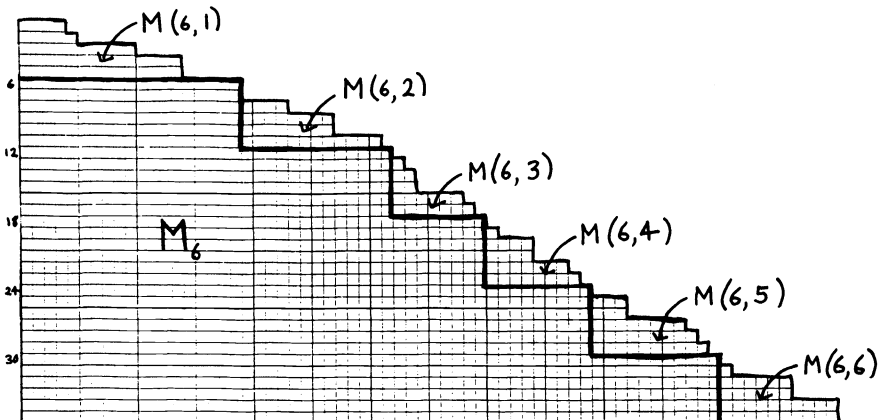


FIG. 4. Stepwise approximation and border matrices.

the total number of nonblank entries in the a -border matrices $M(a, i)$ for $i = 1, \dots, \lfloor n/a \rfloor$, is less than am , we can process these border matrices in $O(am)$ time. Finally in $O(n)$ comparisons we can compare the row-minima found in M_a with the row-minima found in the a -border matrices, and hence determine the row-minima of M . $\square$

For each row i we will use $j(i)$ to denote the column of M which contains the minimum in row i of M .

LEMMA 2.2. *If $i \leq k$ with $f(i) = f(k)$, then there are no row-minima of M in the regions $M[i, k; 1, j(k) - 1]$ and $M[k, n; j(k) + 1, f(k)]$.*

Proof. This follows immediately from properties (a) and (c) in the definition of a (falling) staircase matrix. The two regions which cannot contain row-minima are shown in Fig. 5.

COROLLARY 2.3. *Let $0 = k_0 < k_1 < k_2 < \dots < k_r = n$ be such that the set of rows $\{k_i : 1 \leq i \leq r\}$ contains all the step-rows of M . For each $i = 1, \dots, r$ let $J(k_i)$ be the set of columns $\{j : j(k_i) \leq j \leq f(k_i) \text{ and } j \notin \cup_{1 \leq h \leq i-1} \{j(k_h) + 1, \dots, f(k_h)\}\}$, and let $A(k_i)$ be the matrix consisting of the intersection of rows $k_{i-1} + 1, \dots, k_i - 1$ of M with the columns in $J(k_i)$. Then each $A(k_i)$ is a totally monotone matrix, and $\sum_{i=1}^r |J(k_i)| \leq m + r$.*

Proof. It is easy to see that each $A(k_i)$ is totally monotone since it is a rectangular submatrix of $M[k_{i-1} + 1, k_i; j(k_i), f(k_i)]$, and $M[k_{i-1} + 1, k_i; j(k_i), f(k_i)]$ contains no blank entries since every step-row is contained in $\{k_1, \dots, k_r\}$. For each i let $J'(k_i) = J(k_i) \setminus \{j(k_i)\}$. It is easy to check that the $J'(k_i)$ are disjoint, and hence $\sum_{i=1}^r |J'(k_i)| \leq m$, which implies $\sum_{i=1}^r |J(k_i)| \leq m + r$ as desired.

An example illustrating the matrices $A(k_i)$ is shown in Fig. 6. $\square$

COROLLARY 2.4. *For any positive integer a , we have*

$$q_c(n, n, m) \leq q_c(n/a, n/a, m) + O(am + n).$$

Proof. Let N be a staircase matrix of shape (n, n, m) , and let $M = N_a$. By Lemma 2.1 and its proof, it suffices to show that the number of comparisons needed to process M is at most $q_c(n/a, n/a, m) + O(m + n)$. Let $r = \lfloor n/a \rfloor$ and let $k_0 = 0, k_r = n$ and $k_i = (i + 1)a - 1$ for $1 \leq i \leq r - 1$. Let S be the staircase matrix consisting of rows $k_1, \dots, k_r$ of M . Note that S contains all the step-rows of M , and is of shape $(n/a, n/a, m)$. Let $A(k_i)$ be defined as in Corollary 2.3. Since S is of shape $(n/a, n/a, m)$, the num-

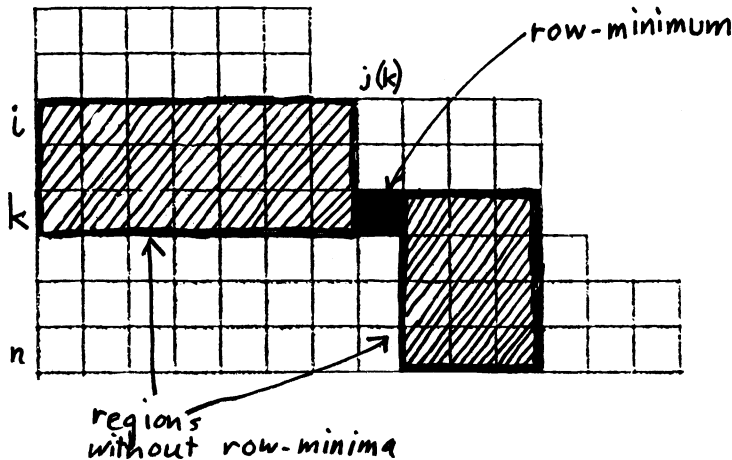


FIG. 5. Regions not containing row-minima.

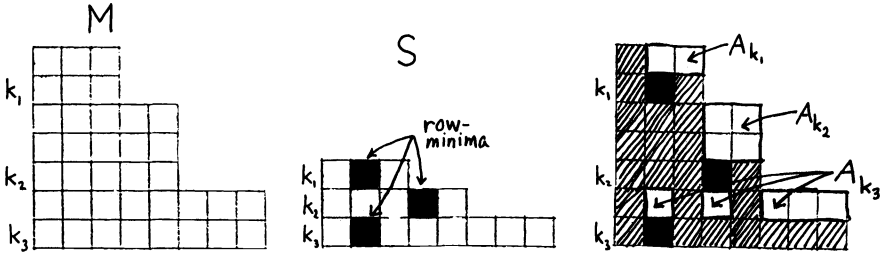


FIG. 6. The matrices $A(k_i)$.

ber of comparisons needed to process it is at most $q_c(n/a, n/a, M)$. After processing S we know the $A(k_i)$ and hence can process them using SMAWK in $O(n + m + r) = O(n + m)$ time, since the total number of rows in the $A(k_i)$ is n and the total number of columns is at most $m + r$. $\square$

COROLLARY 2.5. $q_c(n, n, m) \leq q_c(n/a^2, n/a^2, m) + O(am + n)$ for any $a > 0$.

Proof. Apply Corollary 2.4 twice. $\square$

Before giving the main theorem of this section, which yields the $O(m\alpha(n) + n)$ algorithm, we present some further notation that we will need in the theorem and to define the function $\alpha(n)$. We define the functions $L_i(n)$ for $i = -1, 0, 1, 2, \dots$ recursively as follows. $L_{-1}(n) = n/2$, and for $i \geq 0$, $L_i(n) = \min_s \{L_{i-1}^s(n) \leq 1\}$. Thus $L_0(n) = \lceil \log n \rceil$, $L_1(n)$ is essentially $\log^*(n)$, $L_2(n)$ is essentially $\log^{**}(n)$, etc. We now define $\alpha(n) = \min \{s : L_s(n) \leq s\}$. Next, if $(M, f(i))$ is a staircase matrix, we will call the columns $f(i-1) + 1, \dots, f(i)$ the i th slice of M . This is illustrated in Fig. 7.

We are now ready to prove the key result, which will lead to the $O(m + \alpha(n))$ time algorithm, and since both the theorem and its proof are quite technical, we first provide a sketch of the underlying ideas. Let us regard the number of steps t of a staircase matrix M as a function $t(n)$ of the number of rows n of M . If $t(n)$ is the constant function 2, then it is trivial to partition M into two totally monotone matrices and use SMAWK to find the row-minima in linear time. On the other hand, the general case we want our algorithm to handle is that $t(n) = n$, although by Lemma 2.1, to achieve an $O(m\alpha(n) + n)$ upper bound on $q_c(n, n, m)$ it would suffice for our algorithm to han-

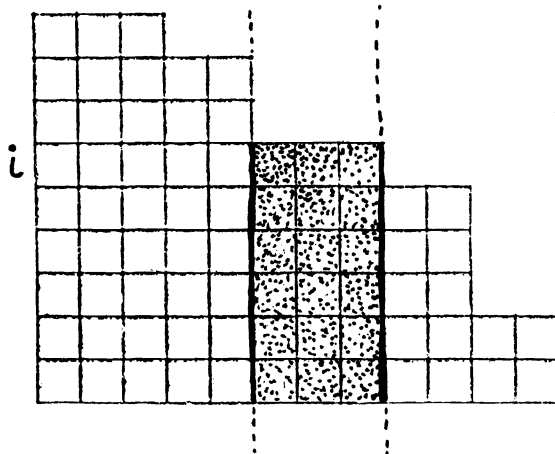


FIG. 7. The i th slice of a staircase matrix.

dle the case that $t(n) = n/\alpha(n)$. What we would like to be able to do is, given a staircase matrix with $t(n) = n/L_s(n)$, in a linear number of comparisons reduce the problem to processing a staircase matrix with $t(n) = n/L_{s-1}(n)$. Since $L_{\alpha(n)}(n) = \alpha(n)$ and $L_{-1}(n) = n/2$, by repeating this $\alpha(n) + 1$ times (and hence doing $O(\alpha(n)(n + m))$ work), we could reduce processing a staircase matrix with $t(n) = n/\alpha(n)$ to processing a staircase matrix with $t(n) = 2$, which could be done in linear time. This would thus yield an $O(\alpha(n)(n + m))$ upper bound on $q_c(n, n, m)$. In reality things are slightly more complicated. In particular, we actually reduce the processing of the staircase matrix M with $t(n) = n/L_s(n)$ to the processing of a set, $\{G_i\}$, of staircase matrices, with each $t(n_i) = n_i/L_{s-1}(n_i)$. In doing this we must take care that the total number of rows and columns in the G_i is not too much larger than the number of rows and columns in M , and this is where some of the technicalities arise. To clarify the presentation, we separate the reduction into two pieces. In Theorem 2.6, we show how to reduce the processing of a staircase matrix of shape (n, n, m) to the processing of staircase matrices of shape $(n_i/L_{s-1}(n_i), n_i, m_i)$ in such a way that the bounds on the comparisons needed, the total number of rows, and the total number of columns are not too bad. Then in Corollary 2.7 we combine this with Corollary 2.4 to achieve the desired reduction.

THEOREM 2.6. *There is a constant c_1 such that for each $s \geq 0$ we have*

$$\begin{aligned} q_c(n, n, m) \leq & c_1(m + nL_s(n)) \\ & + \max \left\{ \sum_{i=1}^k q_c(n_i/L_{s-1}(n_i), n_i, m_i) : \sum_{i=1}^k n_i \leq nL_s(n) \right. \\ & \left. \text{and } \sum_{i=1}^k m_i \leq m + nL_s(n) \right\}. \end{aligned}$$

Proof. Let M be a staircase matrix of shape (n, n, m) . We will show that after some preprocessing, which requires at most $O(m + nL_s(n))$ comparisons, we can reduce the problem of determining the row-minima of M to determining the row-minima of a set of submatrices $G_1, \dots, G_k$, where G_i is of shape $(n_i/L_{s-1}(n_i), n_i, m_i)$, $\sum_{i=1}^k n_i \leq nL_s(n)$, and $\sum_{i=1}^k m_i \leq m + nL_s(n)$.

For $i = 0, \dots, L_s(n) - 1$ we define families $\mathcal{B}_i$ of staircase matrices recursively as follows. For any staircase matrix B , let n_B be the number of rows in B . Recall that, as defined in the fourth paragraph of this section, $\Gamma(B, a)$ denotes the set of a -border matrices of B , and B_a denotes the stepsize a approximation of B . We define $\mathcal{B}_0 = \{M\}$, and for $i > 0$, $\mathcal{B}_i = \bigcup \{ \Gamma(B, L_{s-1}(n_B)) : B \in \mathcal{B}_{i-1} \}$. In other words, $\mathcal{B}_i$ is formed by taking border matrices of the appropriate size of the matrices in $\mathcal{B}_{i-1}$. Now for $i = 1, \dots, L_s(n) - 1$ we define families $\mathcal{A}_i$ of staircase matrices by

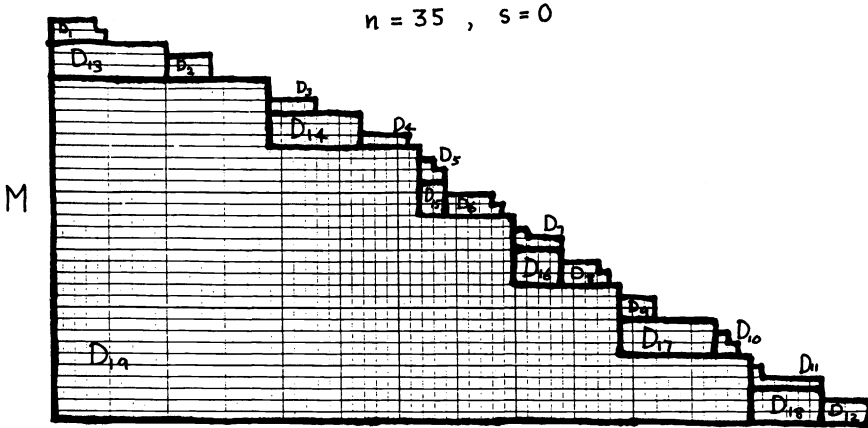
$$\mathcal{A}_i = \bigcup \{ B_{L_{s-1}(n_B)} : B \in \mathcal{B}_{i-1} \}.$$

In other words $\mathcal{A}_i$ is formed by taking approximations of the appropriate stepsize of the matrices in $\mathcal{B}_{i-1}$. Finally, let $\mathcal{D} = \mathcal{B}_{L_s(n)-1} \cup \bigcup \{ \mathcal{A}_i : 1 \leq i \leq L_s(n) - 1 \}$. Write $\mathcal{D} = \{D_1, \dots, D_k\}$ and for $i = 1, \dots, k$, let n_i and p_i be the number of rows and columns in D_i , respectively. An example is shown in Fig. 8.

It is not hard to verify that $\mathcal{D}$ has the following properties.

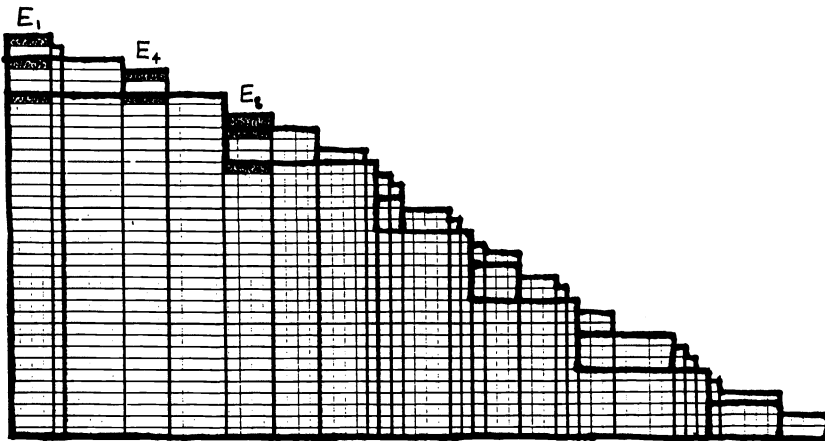
- (i) $D_i \cap D_j = \emptyset$ if $i \neq j$.
- (ii) D_i has shape $(n_i/L_{s-1}(n_i), n_i, p_i)$.
- (iii) Every nonblank entry of M is in some D_i .
- (iv) Every row of M intersects at most $L_s(n)$ of the D_i .
- (v) Every column of M intersects at most $L_s(n)$ of the D_i .

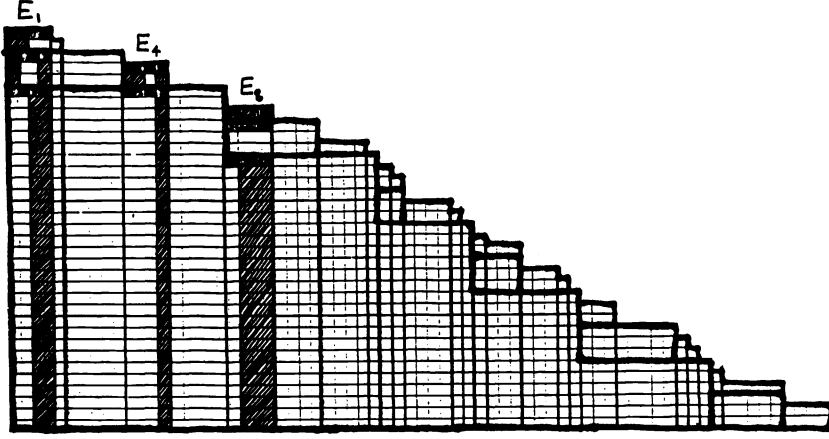
The matrices D_i are almost the desired matrices G_i since they have the correct ratio of steps to rows, and by (iv) we have that $\sum_{i=1}^k n_i \leq nL_s(n)$ as desired. However, we have $\sum_{i=1}^k p_i \leq mL_s(n)$, and $mL_s(n)$ may be larger than the desired upper bound of

FIG. 8. $\{D_1, \dots, D_k\}$.

$m + nL_s(n)$. Thus, before processing the D_i , we must do some preprocessing to delete columns from the D_i to obtain the matrices G_i with the desired bound on the total number of columns. We now describe this preprocessing.

For $i = 1, \dots, n$, let E_i be the matrix formed by taking the intersection of the i th slice, i.e., columns $f(i-1) + 1, \dots, f(i)$, with each row r of M such that for some j we have $M_{r,f(i)} \in D_j$ and $M_{r-1,f(i)} \notin D_j$. In other words, E_i is formed by taking the intersection of the i th slice with one row from each D_j which has a nonblank entry in the i th slice. Moreover, the row taken is the uppermost row with a nonblank entry in the i th slice. An example is shown in Fig. 9. It is easy to check that by the definition of border matrices and approximation matrices, if $M_{r,f(i)}$ is in some D_j , then $M_{r,l}$ is also in that D_j for $f(i-1) + 1 \leq l \leq f(i) - 1$. Thus, for each i , E_i is a totally monotone matrix, and by (v), has at most $L_s(n)$ rows. Since the columns of the E_i are disjoint, we can process (i.e., find the row-minima of) all the E_i with $O(m + nL_s(n))$ comparisons. After doing this, we use (as described in Lemma 2.2) the information about the positions of row-minima in E_i to delete as many columns as possible from the D_j which intersect E_i . We will call this cleaning the D_j . An example is shown in Fig. 10. Moreover, by the arguments

FIG. 9. E_i .

FIG. 10. Cleaning the D_j .

used in the proof of Corollary 2.3, after deleting these columns, the total number of columns remaining in the cleaned D_j between $f(i-1)+1$ and $f(i)$ inclusive is at most $f(i) - f(i-1) + L_s(n)$. If for each j , we let G_j be what remains of D_j after all cleaning possible from the processing of the E_i has been done, and let m_j be the number of columns in G_j , we have $\sum_{j=1}^k m_j \leq \sum_{i=1}^n (f(i) - f(i-1) + L_s(n)) \leq m + nL_s(n)$. Moreover, as deleting columns cannot increase the number of steps in a staircase matrix, G_j is of shape $(n_j/L_{s-1}(n_j), n_j, m_j)$.

Finally, we process the G_j , and then determine the minimum value in each row r of M by comparing the minima found in the appropriate row of each G_j which intersects r . By (iv), this final step takes $O(nL_s(n))$ comparisons. Combining all this, it is easy to see that there is a constant c_1 such that the total number of comparisons is at most $c_1(m + nL_s(n)) + \sum_{i=1}^k q_c(n_i/L_{s-1}(n_i), n_i, m_i)$ as desired. $\square$

COROLLARY 2.7. *There is a constant c_2 such that for each $s \geq 0$ we have*

$$q_c(n/L_s(n), n, m) \leq c_2(m + n) + \max \left\{ \sum_{i=1}^k q_c(n_i/L_{s-1}(n_i), n_i, m_i) : \sum_{i=1}^k n_i \leq n \text{ and } \sum_{i=1}^k m_i \leq m + n \right\}.$$

Proof. By Corollary 2.4 we have $q_c(n/L_s(n), n, m) \leq O(m + n) + q_c(n/L_s(n), n/L_s(n), m)$, and since $(n/L_s(n))L_s(n/L_s(n)) \leq n$, by Theorem 2.6 we have

$$q_c(n/L_s(n), n/L_s(n), m) \leq c_1(m + n) + \max \left\{ \sum_{i=1}^k q_c(n_i/L_{s-1}(n_i), n_i, m_i) : \sum_{i=1}^k n_i \leq n \text{ and } \sum_{i=1}^k m_i \leq m + n \right\}.$$

Combining these yields the desired result. $\square$

COROLLARY 2.8. *There is a constant c_3 such that for each $s \geq 0$ we have $q_c(n/L_s(n), n, m) \leq c_3((s+1)m + (s+1)^2n)$.*

Proof. The proof is by induction on s . First consider the case when $s = 0$. Since $L_{-1}(n_i) = n_i/2$, by Corollary 2.7 we have $q_c(n/L_0(n), n, m) \leq c_2(m + n) + \max \left\{ \sum_{i=1}^k q_c(2, n_i, m_i) : \sum_{i=1}^k n_i \leq n \text{ and } \sum_{i=1}^k m_i \leq m + n \right\}$. As remarked earlier, there is a positive constant c_0 such that $q_c(2, n_i, m_i) \leq c_0(m_i + n_i)$ for each i , since any

staircase matrix with at most two steps can be partitioned into two totally monotone matrices. Thus if we set $c_3 = c_2 + c_0$ we have $q_c(n/L_0(n), n, m) \leq c_3(m + n)$ as desired.

Now suppose $s \geq 1$ and that the statement holds for $s - 1$ with $c_3 = c_0 + c_2$ as above. By Corollary 2.7 and the inductive hypothesis we have

$$\begin{aligned}
 q_c(n/L_s(n), n, m) &\leq c_2(m + n) \\
 &\quad + \max \left\{ \sum_{i=1}^k q_c(n_i/L_{s-1}(n_i), n_i, m_i) : \right. \\
 &\quad \left. \sum_{i=1}^k n_i \leq n \text{ and } \sum_{i=1}^k m_i \leq m + n \right\} \\
 &\leq c_2(m + n) + \max \left\{ \sum_{i=1}^k c_3(sm_i + s^2n_i) : \right. \\
 &\quad \left. \sum_{i=1}^k n_i \leq n \text{ and } \sum_{i=1}^k m_i \leq m + n \right\} \\
 &\leq c_2(m + n) + c_3(s(m + n) + s^2n).
 \end{aligned}$$

Finally

$$c_2(m + n) + c_3(s(m + n) + s^2n) \leq c_3((s + 1)m + (s + 1)^2n)$$

since $c_3 \geq c_2$. $\square$

THEOREM 2.9. $q_c(n, n, m) = O(m\alpha(n) + n)$.

Proof. Let $p = n/(\alpha(n))^2$. By Corollary 2.5 we have $q_c(n, n, m) \leq q_c(p, p, m) + O(m\alpha(n) + n)$. By Lemma 2.1 we have

$$q_c(p, p, m) \leq q_c(p/\alpha(p), p, m) + O(\alpha(p)m + n).$$

By Corollary 2.8 we have $q_c(p/\alpha(p), p, m) \leq c_3((\alpha(p) + 1)m + (\alpha(p) + 1)^2p)$. Combining these yields $q_c(n, n, m) = O(m\alpha(n) + n)$ since $\alpha(p) \leq \alpha(n)$ and $(\alpha(p) + 1)^2p = O(n)$. $\square$

3. Data structures. Because of the recursive structure of our algorithm and the fact that it operates on substaircase matrices of the original staircase matrix, and since any staircase matrix can be extended to a totally monotone rectangular matrix, it makes sense to represent all our staircase matrices as substaircase matrices of some large totally monotone matrix U .

We will represent a staircase matrix M by the following set of data-structures, which we will refer to as the initial data-structures for M .

An integer variable $\#ROW_M$ containing the number of rows.

An integer array, R_M , of length $\#ROW_M$ where $R_M[i]$ contains the number of the row of U which contains the i th row of M .

A doubly-linked list C_M which contains the column information for M . More precisely, the i th item in C_M contains the number of the column of U which is the i th column of M plus pointers to the $(i - 1)$ st and $(i + 1)$ st items in C_M . For simplicity we will abuse our notation and use $C_M[i]$ to denote the number of the column of U which is the i th column of M . We will always have $C_M[1] < \dots < C_M[m]$, where m is the number of columns of M .

A pointer array F_M of length $\#ROW_M$ with $F_M[i]$ containing a pointer to the $f(i)$ th item of C_M where $\{f(i)\}$ is the boundary sequence of M , i.e., column $f(i)$ is the rightmost column in M with a nonblank entry in the i th row of M .

The output information will be stored in a pointer array of length $\#ROW_M$ named MIN_M . For each i , the entry $MIN_M[i]$ will contain a pointer to the $j(i)$ th item of C_M , where the $j(i)$ th column of M contains the minimum value of the i th row of M .

Some of these data-structures are illustrated in Fig. 11. As well as verifying that our algorithm can initialize and maintain these data-structures, it is necessary to make sure

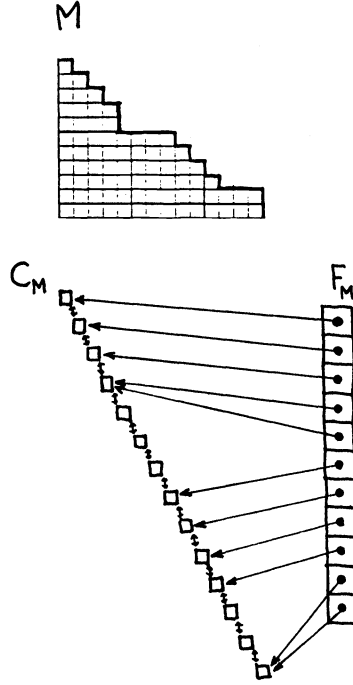


FIG. 11. Data-structures.

that the SMAWK algorithm will still run in linear time if the totally monotone matrix is represented with these data-structures. However, this is straightforward to check, and in fact is done in [AK87].

We now show that the upper bounds on the function q_c in the previous section also apply to the function q , up to constant multiplicative factors. In order to do this it suffices to re-prove Lemma 2.1, Corollary 2.4, and Theorem 2.6 since these are the only bounds in which an algorithm is actually used (rather than a simple manipulation of previous bounds on q_c). In each case it suffices to show that the time needed to create the initial data-structures of each staircase matrix involved can be bounded (up to constant factors) by the corresponding bound on the number of comparisons. Moreover, since the number of rows of M can obviously be determined during the initialization of R_M , we need only consider the time needed to initialize the data-structures R_M , C_M , and F_M . It is trivial to see that creating and maintaining the data-structures needed for Lemma 2.1 can easily be done in $O(am + n)$ time so we can concentrate on Corollary 2.4 and Theorem 2.6.

PROPOSITION 3.1. *For any positive integer a , we have*

$$q(n, n, m) \leq q(n/a, n/a, m) + O(am + n).$$

Proof. Let N , M , S , and $\{A_{k_i} : 1 \leq i \leq r\}$ be as in the proof of Corollary 2.4. Given the initial data-structures for N it is straightforward to create the initial data-structures for M and S in $O(m + n)$ time, so it suffices to show that given these data-structures and the positions of the row-minima for S , we can create the initial data-structures for the A_{k_i} in $O(n)$ time.

It is clear that in $O(n)$ time we can initialize all of the $R_{A_{k_i}}$, so we will restrict our attention to the initialization of the $C_{A_{k_i}}$ and $F_{A_{k_i}}$. Since each A_{k_i} is a rectangular matrix, all the entries of $F_{A_{k_i}}$ will point to the last entry in $C_{A_{k_i}}$. Hence it suffices to show how to

initialize the $C_{A_{k_i}}$. We will use MIN_S to repeatedly modify the linked list C_M , and take $C_{A_{k_i}}$ to be a piece of the $(i-1)$ st modification of C_M , which we will denote by $C_M(i-1)$. Let $C_M(0)$ be C_M . Given $C_M(i-1)$, we split $C_M(i-1)$ into two lists, $C_{A_{k_i}}$ and $C_M(i)$ as follows. $C_{A_{k_i}}$ is the sublist of $C_M(i-1)$ consisting of all items between (and including) the items pointed to by $MIN_S[i]$ and $F_S[i]$. $C_M(i)$ is the list obtained by deleting all but the first item in $C_{A_{k_i}}$ from $C_M(i-1)$. Since the lists are doubly linked it is clear that $C_{A_{k_i}}$ and $C_M(i)$ can be created from $C_M(i-1)$ in constant time. Thus the total time needed is $O(n)$.

For the purposes of the dynamic algorithm in §4 it is important to note that in order to create the initial data structures for A_{k_i} it was only necessary to have the initial portion of C_M up to $C_M[F_S[i]]$, and the pointers $MIN_S[1], \dots, MIN_S[i]$ and $F_S[1], \dots, F_S[i]$. $\square$

PROPOSITION 3.2. *There is a constant c_4 such that for each $s \geq 0$ we have*

$$\begin{aligned} q(n, n, m) \leq & c_4(m + nL_s(n)) \\ & + \max \left\{ \sum_{i=1}^k q(n_i/L_{s-1}(n_i), n_i, m_i) : \right. \\ & \left. \sum_{i=1}^k n_i \leq nL_s(n) \text{ and } \sum_{i=1}^k m_i \leq m + nL_s(n) \right\}. \end{aligned}$$

Proof. Let $M, \{E_i : 1 \leq i \leq n\}, \{\mathcal{B}_i : 0 \leq i \leq L_s(n) - 1\},$

$$\{\mathcal{A}_i : 1 \leq i \leq L_s(n) - 1\},$$

$\mathcal{D}, \{D_i, n_i, G_i, m_i : 1 \leq i \leq k\}$ be as in the proof of Theorem 2.6. We must show that in $O(m + nL_s(n))$ time we can create the initial data-structures for the E_i , and that given $\{MIN_{E_i} : 1 \leq i \leq n\}$ we can create the initial data-structures for all $\{G_i : 1 \leq i \leq k\}$ in $O(m + nL_s(n))$ time also. In creating the initial data structures we will need to evaluate $L_{s-1}(x)$ for many integral values of x between 1 and n . For $s = 0$ and $s = 1$ this is not a problem since $L_{-1}(x) = x/2$ and $L_0(x) = \lceil \log x \rceil$. For $s \geq 2$ it is not hard to see that a table of the values $L_{s-1}(1), \dots, L_{s-1}(n)$ can be built in $O(n)$ time. Thus we will assume that we can evaluate $L_{s-1}(x)$ in constant time for any integer x between 1 and n .

We begin by creating data-structures for $\{D_i : 1 \leq i \leq k\}$ and $\{E_i : 1 \leq i \leq n\}$. It is easy to see that we can create the column lists $\{C_{E_i} : 1 \leq i \leq n\}$ in $O(m + n)$ time using C_M and F_M , and since the E_i are rectangular it will be trivial to initialize the F_{E_i} once the R_{E_i} have been created. We will create the R_{E_i} while building the data-structures for the D_i . Before doing this, in $O(m + n)$ time we enhance the linked list C_M by adding, to each item which is pointed to by some $F_M[i]$, a back pointer to the smallest i with $F_M[i]$ pointing to it. Since the total number of columns in the D_i is too large, we will not build separate column lists for the D_i . Instead we let F'_{D_i} be an array of pointers such that $F'_{D_i}[0]$ points to the item in C_M which is the leftmost column in D_i , and for $j \geq 1$, the entry $F'_{D_i}[j]$ is a pointer to the item in C_M which represents the rightmost column of the j th row of D_i . It is easy to see that $\{R_{D_i} : 1 \leq i \leq k\}$ and $\{F'_{D_i} : 1 \leq i \leq k\}$ provide all the information we need but only require $O(nL_s(n))$ storage.

As in Theorem 2.6 we will use n_B to denote the number of rows in any staircase matrix B . Recall that $\{D_1, \dots, D_k\} = \mathcal{A}_1 \cup \dots \cup \mathcal{A}_{L_s(n)-1} \cup \mathcal{B}_{L_s(n)-1}$, where $\mathcal{B}_0 = \{M\}$, $\mathcal{B}_i = \cup \{\Gamma(B, L_{s-1}(n_B)) : B \in \mathcal{B}_{i-1}\}$, and $\mathcal{A}_i = \cup \{B_{L_{s-1}(n_B)} : B \in \mathcal{B}_{i-1}\}$. Thus it suffices to show that given R_B and F'_B for any B in some $\mathcal{B}_j$, we can build $R_{B_{L_{s-1}(n_B)}}$, $F'_{B_{L_{s-1}(n_B)}}$, and $\{R_H, F'_H : H \in \Gamma(B, L_{s-1}(n_B))\}$ in $O(n_B)$ time. However, this follows immediately from the definitions of $B_{L_{s-1}(n_B)}$ and $\Gamma(B, L_{s-1}(n_B))$, and from being able to evaluate $L_{s-1}(n_B)$ in constant time.

Finally, it is easy to see that for each D_i , using R_{D_i} , F'_{D_i} , F_M , and the back pointers in the enhanced C_M , starting from the leftmost E_j intersecting D_i , for each E_j which intersects D_i we can initialize one entry in R_{E_j} , namely the number of the top row of D_i which intersects E_j . In order to construct the data structures for the G_i we will also need to enhance the R_{E_j} with back pointers to the D_i . Specifically, for each entry in R_{E_j} we include a back pointer to the D_i to which it belongs. We will denote these enhanced data structures by R'_{E_j} . This is illustrated in Fig. 12. It is not hard to see that all the initialization of R'_{E_j} arising from one particular D_i can be done in $O(n_{D_i})$ time. Furthermore, if these initializations are done starting with the D_i in $\mathcal{B}_{L_s(n)-1}$, followed by the D_i in $\mathcal{A}_{L_s(n)-1}$, then $\mathcal{A}_{L_s(n)-2}$, etc., the entries in each R'_{E_j} will be in the correct order. The total time needed to initialize all the R'_{E_j} is thus $O\{\sum_{i=1}^k n_{D_i}\} = O(nL_s(n))$.

Since $R_{G_i} = R_{D_i}$ for each i , all that remains to be done is to build the C_{G_i} and the F_{G_i} . Now we show how, given the $MIN_{E_1}, \dots, MIN_{E_n}$ in order, we can do this in $O(m + nL_s(n))$ time. Let x_j and y_j be the number of rows and columns, respectively, in E_j . Using the information in MIN_{E_1} and R'_{E_1} , it is easy to see that in $O(x_1 + y_1)$ time we can split C_{E_1} into x_1 sublists which are the beginnings of the column linked lists for the G_j intersecting E_1 . By checking whether the item in C_M which corresponds to the last item in C_{E_1} has a back pointer to a D_j , we can decide whether or not it is time to initialize some pointers in the F_{G_j} . By doing all this for each of the E_i in order, we initialize the data-structures for the G_j in $O(\sum_{i=1}^n x_i + y_i) = O(nL_s(n) + m)$ time. $\square$

Remark 3.3. Although it was convenient to construct the R_{E_j} from the data-structures for the D_i , it will be important in modifying our algorithm for the dynamic case in the next section to note that we could do it with less information in the following sense. If we know r , j , and n but not C_M or F_M , we can determine whether row r would be in E_j if the j th slice is nonempty, even though we cannot actually determine whether the j th slice is nonempty. Moreover, determining this for all j and r can be done in $O(nL_s(n))$ time.

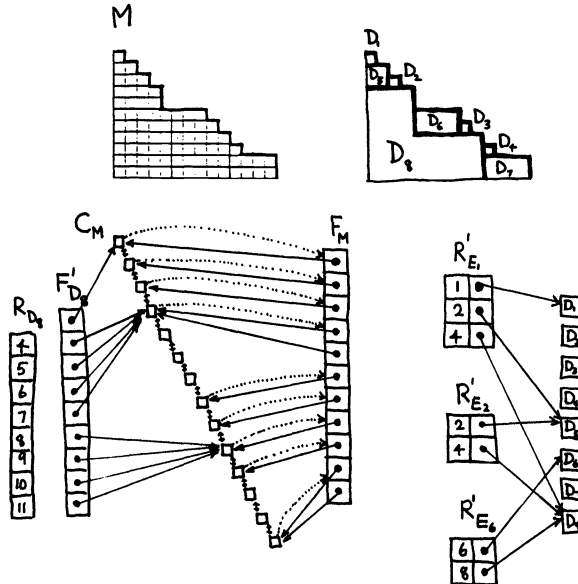


FIG. 12. Enhanced data-structures.

4. Applications to dynamic programming. Eppstein, Galil, and Giancarlo ([EGG88]) give an $O(n \log n)$ time algorithm for the following problem. Suppose w_{ij} for $1 \leq j \leq i \leq n$ is a weight function which satisfies

$$w_{ij} + w_{hk} \leq w_{ik} + w_{hj} \quad \text{for all } j < k < h < i.$$

Determine

$$E[i] = \min_{1 \leq j \leq i} \{ D[j-1] + w_{ij} \}, \quad i = 1, 2, \dots, n,$$

where $D[0]$ is given, and for $j \geq 1$, $D[j]$ can be calculated from $E[j]$ in constant time. We will call this problem the convex dynamic programming problem. (Our definition differs slightly from that of Eppstein–Galil–Giancarlo in that we have reversed the role of rows and columns and shifted the indexing by 1 to be more consistent with our row-minima finding algorithms and definition of staircase matrix.)

Eppstein, Galil, and Giancarlo showed that several dynamic programming problems arising in molecular biology, geology, and speech recognition can be reduced to solving $O(n)$ convex dynamic programming problems of size n , where n is the size of the original dynamic programming problem. Eppstein, Galil, and Giancarlo thus obtained $O(n^2 \log n)$ algorithms for the original dynamic programming problems—a substantial improvement over the standard $O(n^3)$ time algorithms. In this section we give an $O(n\alpha(n))$ time algorithm for solving convex dynamic programming problems of size n , and hence improve the algorithms for the dynamic programming problems described by Eppstein, Galil, and Giancarlo to $O(n^2\alpha(n))$.

Following Eppstein, Galil, and Giancarlo, we define $M_{ij} = D[j] + w_{ij}$, and let M be the lower triangular matrix (M_{ij}) . Obviously, calculating the $E[j]$ is simply the problem of finding the row-minima in M . As noted by Eppstein, Galil, and Giancarlo, from the inequality satisfied by the w_{ij} , it is trivial to prove that whenever $j < k < h < i$ and $M_{hj} \leq M_{hk}$, then $M_{ij} \leq M_{ik}$. Thus M is a staircase matrix with boundary sequence $f(i) = i$. However, we cannot apply our row-minima finding algorithm to M directly, since by the definition of M_{ij} , we cannot evaluate the entry M_{ij} until after we have determined the minimum value in row $j - 1$. Thus, in order to apply our matrix searching techniques to dynamic programming we need to introduce a new version of matrix searching in which the timing of when an entry may be examined is restricted. As before, we will restrict our attention to falling staircase matrices.

Let $(M, \{f(i)\})$ be an $n \times m$ staircase matrix. For each j with $1 \leq j \leq m$ let $b(j) = \max \{i : f(i) < j\}$. We will call row $b(j)$ the *foundation row* for column j . This is illustrated in Fig. 13. Note that if we adopt the convention that the 0th row is a step-row which contains only blanks, then the foundation row for a column is the highest numbered step-row with which the column's intersection is blank. Recall that the i th slice is the set of columns $f(i-1) + 1, \dots, f(i)$. Note that the i th slice is nonempty if and only if row $i-1$ is a step-row. We will call a row-minima finding algorithm for staircase matrices *ordered* if it satisfies the following two constraints.

(1) For each j with $b(j) > 0$, the algorithm always determines the minima of rows $1, \dots, b(j)$ before querying values of the nonblank entries in column j . We will refer to this constraint as the *evaluation constraint*.

(2) The algorithm uses no information about the columns in the i th slice until after the minimum in row $i-1$ has been found. Thus when the algorithm begins processing the staircase matrix M , the only initial information it has is the set of rows of M (i.e., the information in R_M), and the column information for the first slice. We will call this the *online constraint*.

in M_a , thus determining the minimum value of that row for M . When $Z(i)$ finishes, we resume running Z_a on M_a until Z_a has determined the minimum in row $\min((i+1)a-1, n)$, and so on. Since M_a has no nonblank entries in rows $1, \dots, a-1$, we use the convention that Z_a has already determined the minima for these rows of M_a before it actually starts running. Thus the ordered algorithm for M actually begins with $Z(1)$ on $M(a, 1)$. It is not hard to check that this combination of Z_a and $Z(i)$ obeys the evaluation and online constraints, and clearly $O(n)$ additional work is done in comparing the two possible minima for each row. $\square$

LEMMA 4.2. *For any positive integer a , we have*

$$q_o(n, n, m) \leq q_o(n/a, n/a, m) + O(am + n).$$

Proof. Let N, M, S , and $\{A_{k_i} : 1 \leq i \leq t\}$, be as in the proof of Corollary 2.4, and let Z_S be an ordered row-minima finding algorithm for S . As in the proof of Corollary 2.4, it will suffice to show that there is an ordered algorithm for M which requires at most $q_o(n/a, n/a, m) + O(m + n)$ time. We will create an ordered algorithm for M by interleaving Z_S with running SMAWK on the A_{k_i} . Note that given the row information for N and a we can compute the row information for S before any processing begins, and this computation can be done in $O(n/a)$ time. In addition, note that the definition of $A(k_i)$ depends only on the positions of the row-minima of rows

$$\{k_j : 1 \leq j \leq i-1 \text{ and } k_j \text{ is a step-row}\}.$$

Since Z_S cannot determine the minimum of row k_i without already having determined the row-minima of all the rows $\{k_j : 1 \leq j \leq i-1 \text{ and } k_j \text{ is a step-row}\}$, $A(k_{i+1})$ is known as soon as Z_S has found the minimum in row k_i . In addition, the columns in S have the same foundation rows as they do in M . Thus we can create an ordered row-minima finding algorithm for M by running Z_S , but for $i = 0, 1, \dots, t-1$, as soon as the row-minima of row k_i is determined, we interrupt Z_S and run SMAWK on $A(k_{i+1})$, resuming Z_S as soon as the row-minima of $A(k_{i+1})$ are determined. Since the foundation row of every column in $A(k_{i+1})$ is among $\{k_0, k_1, \dots, k_i\}$, it is easy to check that this algorithm satisfies the evaluation and online constraints. $\square$

THEOREM 4.3. *There is a constant c_5 such that for each $s \geq 0$ we have*

$$\begin{aligned} q_o(n, n, m) &\leq c_5(m + nL_s(n)) \\ &+ \max \left\{ \sum_{i=1}^k q_o(n_i/L_{s-1}(n_i), n_i, m_i) : \right. \\ &\quad \left. \sum_{i=1}^k n_i \leq nL_s(n) \text{ and } \sum_{i=1}^k m_i \leq m + nL_s(n) \right\}. \end{aligned}$$

Proof. We will use the notation $M, \{E_i : 1 \leq i \leq n\}, \{\mathcal{B}_i : 0 \leq i \leq L_s(n) - 1\}, \{\mathcal{A}_i : 1 \leq i \leq L_s(n) - 1\}, \mathcal{D}, \{D_i, n_i, G_i, m_i : 1 \leq i \leq k\}$ as in the proof of Theorem 2.6. As before, we will interleave ordered algorithms for the G_i to obtain an ordered algorithm Z for M . At the beginning of processing M , the algorithm Z only knows the set of rows in M , but it is not hard to see (as noted in Remark 3.3) that from this information the data structures, $R_{D_i} = R_{G_i}$ and $R_{E_i}^t$, can be created in $O(nL_s(n))$ time. For each i , let $\{g(i, 1), \dots, g(i, h_i)\} = \{j : G_j \text{ intersects row } i \text{ of } M\}$, ordered so that for $1 \leq h \leq h_i - 1$ the intersection of $G_{g(i,h)}$ with row i is to the left of the intersection of $G_{g(i,h+1)}$ with row i . It is not hard to see that a data structure containing this information can also be initialized in $O(nL_s(n))$ time.

For $i = 1, \dots, k$ let Z_i be an ordered row-minima finding algorithm for G_i . It suffices to show that with an additional $O(m + nL_s(n))$ amount of work, we can interleave the Z_i to create Z . The general idea is as follows. We will view Z as having n phases, where the i th phase will conclude with the determination of the minimum of row i of M . We now describe the i th phase of Z . Immediately after determining the minimum

in row $i - 1$, we have the column information for the i th slice, and since we already know the rows in E_i , we can find the row minima of E_i using SMAWK. After processing E_i we know the intersection of each G_j with the i th slice, and can initialize that part of the column list for each G_j with nonempty intersection with the i th slice. After doing this, continue running $Z_{g(i,1)}$ until the minimum of the intersection of row i with $G_{g(i,1)}$ is found, then run $Z_{g(i,2)}$ on $G_{g(i,2)}$ until the intersection of row i with $G_{g(i,2)}$ has been found, and continue in this manner to find the minimum of the intersection of row i with each $G_{g(i,h)}$ for $h = 1, \dots, h_i$ in increasing order of h . Finally, by comparing these values, we find the minimum for all of row i .

In order for this algorithm to work (let alone be an ordered algorithm) it must be true that while we are running $Z_{g(i,h)}$ on $G_{g(i,h)}$ during the i th phase, $Z_{g(i,h)}$ does not evaluate any entries of $G_{g(i,h)}$ which lie in columns to the right of the i th slice of M . This is necessary because during the i th phase we do not have any information about columns of M lying to the right of the i th slice. However, this follows from $Z_{g(i,h)}$ being an ordered algorithm, since if j is a column of $G_{g(i,h)}$ lying to the right of the i th slice of M , then the foundation row of j in $G_{g(i,h)}$ cannot lie above row i since row i terminates to the left of column j . Thus $Z_{g(i,h)}$ must determine the minimum of the intersection of row i with $G_{g(i,h)}$ before evaluating any entries in column j .

It is easy to see that this algorithm obeys the online constraint, so it suffices to show that it obeys the evaluation constraint. We must show that Z does not evaluate any entries in column j of M before the beginning of phase $b(j) + 1$. However, from the definition of Z and the argument in the preceding paragraph, it is clear that in phase i , Z does not evaluate any entries in columns to the right of the i 'th slice, where $i' = \max \{k : \text{the } k\text{th slice is nonempty and } k \leq i\}$. Since column j is in the $(b(j) + 1)$ -st slice of M , this shows that Z does not evaluate entries in column j before phase $b(j) + 1$. $\square$

5. Open problems. The most obvious open problem is to resolve the gap between the trivial linear lower bound and this paper's slightly superlinear upper bound for finding row-minima in staircase matrices. Other natural questions include characterizing which other shapes of totally monotone partial matrices can be handled by the techniques in this paper and finding parallel algorithms with optimal speedup for row-minima finding in staircase matrices. The first author has recently made some progress in extending this paper's almost linear time algorithm to more general shapes of totally monotone partial matrices. In particular, the algorithm can be extended to matrices in which the nonblank entries in each column form a continuous segment ending at the bottom row. Matrices of this shape arise in optimization problems in computational geometry.

REFERENCES

- [AKMSW87] A. AGGARWAL, M. KLAWE, S. MORAN, P. SHOR, AND R. WILBER, *Geometric applications of a matrix searching algorithm*, *Algorithmica*, 2 (1987), pp. 195–208.
- [AK87] A. AGGARWAL AND M. KLAWE, *Applications of generalized matrix searching to geometric algorithms*, *Discrete Appl. Math.*, to appear.
- [AP88] A. AGGARWAL AND J. PARK, *Notes on searching in multidimensional arrays*, in *Proc. 29th Ann. IEEE Symposium on Found. Comp. Sci.*, (1988), pp. 497–512.
- [EGG88] D. EPPSTEIN, Z. GALIL, AND R. GIANCARLO, *Speeding up dynamic programming*, in *Proc. 29th Ann. IEEE Symposium on Found. Comp. Sci.*, (1988), pp. 488–496.
- [HL87] D. S. HIRSCHBERG AND L. L. LARMORE, *The least weight subsequence problem*, *SIAM J. Comput.*, 16 (1987), pp. 628–638.
- [W88] R. WILBER, *The concave least weight subsequence revisited*, *J. Algorithms*, 9 (1988), pp. 418–425.
- [Y82] F. YAO, *Speed-up in dynamic programming*, *SIAM J. Algebraic Discrete Methods*, 3 (1982), pp. 532–540.

THE BOOLEAN BASIS PROBLEM AND HOW TO COVER SOME POLYGONS BY RECTANGLES*

ANNA LUBIW†

Abstract. For $S \subseteq \{0, 1\}^n$ the *Boolean basis* (or *set basis*) *problem* is to find a minimum size set $B \subseteq \{0, 1\}^n$ such that each $s \in S$ is a Boolean sum of vectors from B . This paper examines tractable special cases of this NP-complete problem. In particular a construction preserving tractability is given. The *rectangle cover problem*—expressing a rectilinear polygon as the union of a minimum number of rectangles—is a main application.

Key words. Boolean basis problem, set basis problem, rectangle cover problem, min-max relations, perfect graphs

AMS(MOS) subject classifications. 68R99, 05C50, 05C70

1. Introduction. The *Boolean basis problem* is as follows: Given a set of vectors $S \subseteq \{0, 1\}^n$ find a minimum size *Boolean basis* for S —i.e., a minimum size set $B \subseteq \{0, 1\}^n$ such that each $s \in S$ is a Boolean sum of vectors from B . (Boolean sum means $1 + 1 = 1$.) Equivalently, given a set S of subsets of a finite set E find a minimum size set B of subsets of E such that each $s \in S$ is a union of sets in B . In general the Boolean basis problem is NP-hard ([St], or see [GJ, p. 222] where the problem is called the “set basis problem”).

In this paper tractable special cases of the Boolean basis problem will be examined. The main result is a composition of instances which preserve this tractability.

Another purpose of this paper is to show that the problem of expressing a rectilinear polygon as the union of a minimum number of rectangles—the *rectangle cover problem*—is a special case of the Boolean basis problem. Other applications of the Boolean basis problem include problems in compiler design and problems of data compression and feature extraction arising in the transmission and storage of digitized sounds and images [BND], [Gi].

The Boolean basis problem can be viewed as a problem on matrices as formulated by Edmonds and Weideman: Let M be a 0-1 matrix with rows R and columns C . Matrices will be considered to be unordered unless otherwise specified. Let

$$I(M) = \{(r, c) \in R \times C : M(r, c) = 1\}.$$

A *block* of M is a submatrix which is constantly 1—more formally, a set $R' \times C'$, where $R' \subseteq R$, $C' \subseteq C$, and such that $R' \times C' \subseteq I(M)$. A *Boolean basis* of M is a set of blocks whose union is $I(M)$. The (*matrix*) *Boolean basis problem* is to find a minimum size Boolean basis of M . Let $\beta(M)$ denote the minimum size of a Boolean basis of M .

This matrix Boolean basis problem is the same as the above vector Boolean basis problem: Define a *Boolean row* (*column*) *basis* of M to be a Boolean basis of the row (column, respectively), vectors of the matrix according to the first definition above. It can easily be shown that any Boolean row (or column) basis of M yields an equicardinal Boolean basis of M , and conversely. Thus the minimum size of a Boolean row (or column) basis of a 0-1 matrix is equal to the minimum size of a Boolean basis of the matrix, and the two problems are equivalent. Another implication here is that the min-

* Received by the editors July 20, 1988; accepted for publication (in revised form) May 19, 1989.

† Department of Computer Science, University of Waterloo, Waterloo, Canada.

$$\begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

FIG. 1.1. Matrices violating the Boolean basis min max equality. In the left-hand matrix the maximum size of an antiblock is 5 but the minimum size of a Boolean basis is 6. In the right-hand matrix the maximum size of an antiblock is 6 but the minimum size of a Boolean basis is 7.

imum size of a Boolean row basis of a 0-1 matrix is equal to the minimum size of a Boolean column basis of the matrix. This is the Boolean analogue of the linear algebra theorem that the row rank of a matrix is equal to the column rank. It can be found in [Ki].

An *antiblock* in the 0-1 matrix M is a set $A \subseteq I(M)$ such that $|B \cap A| \leq 1$ for every block B of M . An equivalent characterization providing an easy recognition algorithm is that $A \subseteq I(M)$ is an antiblock if and only if $(r, c), (r', c') \in A$ implies $(r, c') \notin I(M)$ or $(r', c) \notin I(M)$. Clearly any Boolean basis of M is at least as big as any antiblock of M . In particular the minimum size of a Boolean basis of M is greater than or equal to the maximum size of an antiblock. Let $\alpha(M)$ denote the maximum size of an antiblock of M . Then $\beta(M) \geq \alpha(M)$. This is called the *Boolean basis min max relation*. In general the inequality may be strict. Two examples are shown in Fig. 1.1.

A 0-1 matrix is *firm* if the Boolean basis min max equality holds for M and all its submatrices—i.e., $\beta(M') = \alpha(M')$ for all submatrices M' of M .

We propose the following problems: Characterize firm matrices. Do they form an NP class? A co-NP class? Are they recognizable in polynomial time? How hard is the Boolean basis problem on firm matrices?

Parallels between the study of firm matrices and the study of *perfect graphs* will be discussed in § 6.

The main result in this paper is a construction of 0-1 matrices (from simpler 0-1 matrices) that preserves firmness. This can be seen in § 3. In § 5 it will be shown how the construction can be used to build up bigger classes of firm matrices starting from certain initial classes. One such initial class is the class of *interval matrices*. A 0-1 matrix is an *interval matrix* if there is an ordering of its columns making the 1's in each row consecutive. In [Gy] Györi proved that interval matrices are firm. Another more trivial initial class of firm matrices consists of the 0-1 matrices with no block of size 2×2 or more.

In § 4 a polynomial time algorithm will be given that solves the Boolean basis problem on any class of 0-1 matrices built up by the construction from smaller classes for which such algorithms are available.

Throughout the paper implications of the results for the rectangle cover problem will be given. The following section first discusses the rectangle cover problem in detail.

2. The rectangle cover problem. Given a *rectilinear polygon* P —i.e., a polygon in the plane with horizontal and vertical sides—the *rectangle cover problem* is to find a minimum number of rectangles (with horizontal and vertical sides) whose union is P . The complexity of the rectangle cover problem had been open for a long time. A proof that the problem is NP-complete has recently been found [CR].

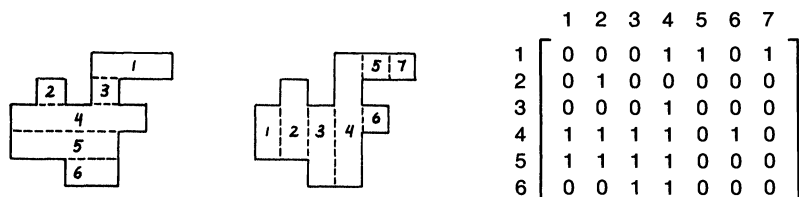


FIG. 2.1. A rectilinear polygon partitioned into horizontal and into vertical swaths, and the corresponding swath matrix.

The rectangle cover problem arises in the manufacture of integrated circuits when photolithographic masks of rectilinear polygonal shapes must be created by exposing a photo-sensitive surface to light one rectangle at a time ([MC, pp. 93–98]).

It will first be shown how the rectangle cover problem is a special case of the Boolean basis problem. There is no loss of generality in assuming that polygons have integer coordinates. Then every polygon is a union of squares of the integer lattice. These will be called *grid squares*. For the rectangle cover problem only maximal rectangles contained in the polygon need be considered and these are also unions of grid squares. A *horizontal* (*vertical*) *swath* in a rectilinear polygon is a maximal sequence of contiguous grid squares of the polygon at the same horizontal (*vertical*) position. The *swath matrix* of a rectilinear polygon P is a 0-1 matrix with a row for each horizontal swath of P and a column for each vertical swath of P and an entry of 1 in a position of the matrix if and only if the corresponding horizontal and vertical swaths contain a common grid square. Thus the 1's of the swath matrix are in direct correspondence with the grid squares of the polygon. See Fig. 2.1.

Let P be a rectilinear polygon with swath matrix M .

PROPOSITION 2.1. *Maximal rectangles in P correspond exactly to maximal blocks in M .*

Thus the rectangle cover problem is exactly the Boolean basis problem on the class of swath matrices.

Antiblocks in M correspond exactly to *antirectangles* of P . An *antirectangle* of P is a set of grid squares (or equivalently, a set of points) inside P , no two contained in a rectangle contained in P . The *rectangle cover min max relation* is that the minimum number of rectangles with union P is greater than or equal to the maximum size of an antirectangle in P . Equality holds if and only if the Boolean basis min max equality holds for M .

Chvátal once conjectured that the rectangle cover min max equality holds for any rectilinear polygon. A counter-example due to Chung is shown in Fig. 2.2, and another counter-example is shown in Fig. 2.3. It is an open problem to characterize rectilinear polygons satisfying the rectangle cover min max equality.

Masek [M] proved a generalization of the rectangle cover problem to be NP-hard: namely, when rectilinear polygons are generalized to rectilinear polygons with rectilinear holes cut out. Note that in this case Proposition 2.1 fails.

Turning to special cases of the rectangle cover problem, consider first the implications of Györi's result [Gy] that interval matrices are firm. Recall that interval matrices were defined in § 1. A polygon is *y-convex* (*x-convex*) if every vertical (horizontal) line segment joining points inside the polygon is entirely contained inside the polygon. The swath matrix of any *y-convex* polygon is an interval matrix. In fact it was this transformation of the *y-convex* rectangle cover problem to a problem on intervals (suggested to Györi by Frank) that inspired the notion of swath matrices. Since interval matrices are firm,

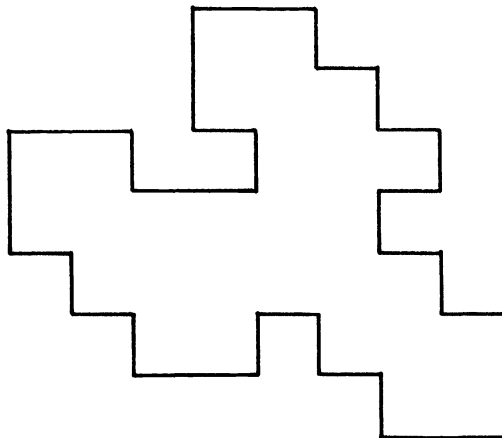


FIG. 2.2. *Chung's polygon which violates the rectangle cover min-max equality (see [CKSS] for argument).*

y -convex rectilinear polygons satisfy the rectangle cover min max equality. The algorithm given in [FK] to solve the Boolean basis problem on interval matrices in polynomial time also solves the rectangle cover problem on y -convex polygons. The case of polygons that are x - and y -convex had been handled earlier [CKSS]. It is not true that every interval matrix is the swath matrix of some y -convex polygon.

The composition of 0-1 matrices which preserves firmness can be applied to rectilinear polygons. This will be discussed in § 3.

In approaching the rectangle cover problem by way of the Boolean basis problem on swath matrices, the question of characterizing swath matrices arises. This problem is open. One uncomfortable fact about swath matrices is that a submatrix of a swath matrix need not be a swath matrix. It may be better to characterize the class of 0-1 matrices that are submatrices of swath matrices, this being a hereditary class. The remainder of this section contains some properties of (submatrices of) swath matrices.

THEOREM 2.2. *Swath matrices are totally balanced.*

THEOREM 2.3. *Swath matrices are totally unimodular.*

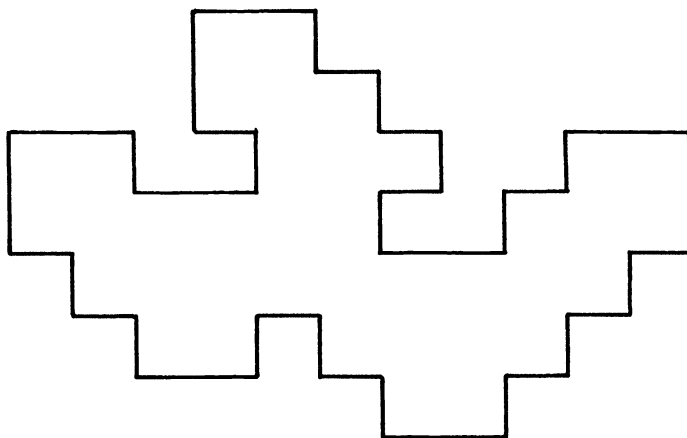


FIG. 2.3. *Another polygon violating the rectangle cover min-max equality.*

A 0-1 matrix is *totally balanced* if it has no $n \times n$ cycle submatrices for $n \geq 3$. A *cycle matrix* is a square 0-1 matrix with exactly two 1's in each row and in each column and such that no proper submatrix has this property. A *totally unimodular* matrix is a 0, +1, -1 matrix for which the determinant of any square submatrix is 0, +1, or -1. Interval matrices are totally balanced and totally unimodular.

Proof of Theorem 2.2. Let P be a rectilinear polygon with swath matrix M . Suppose that M has an $n \times n$ cycle submatrix M' on row set R' and column set C' . Rows (columns) of M and the corresponding horizontal (vertical) swaths of P will be identified by the same names. The 1's of M' can be associated with $2n$ grid squares of P . The center points of these grid squares can be ordered cyclically: $p_1, p_2, \dots, p_{2n}, p_{2n+1} = p_1$ in such a way that p_{2i}, p_{2i+1} are both in a horizontal swath of R' and p_{2i-1}, p_{2i} are both in a vertical swath of C' . Thus the line segments joining consecutive points in the cyclic order are inside the polygon.

Suppose that p_1 and p_2 are both in the vertical swath $k \in C'$. Let r_1 (r_2) be the horizontal swath from R' containing p_1 (p_2 , respectively). Then p_{2n} is in r_1 and p_3 is in r_2 . The sequence $p_3, p_4, \dots, p_{2n}$ and the line segments between them connect p_3 and p_{2n} . Since no horizontal swath of $R' - \{r_1, r_2\}$ shares a grid square with the vertical swath k , p_3 and p_{2n} must still be connected after removing from P the vertical swath k and all horizontal swaths intersecting k in some grid square. Removing a swath from a rectilinear polygon disconnects the polygon. (This property fails when holes are allowed.) Thus p_3 and p_{2n} cannot be on opposite sides of k with respect to x coordinate; otherwise, removing k disconnects them. The horizontal swaths r_1 and r_2 must lie side by side; otherwise, there is a horizontal swath between them intersecting k , and deleting this disconnects p_3 and p_{2n} .

Suppose p_3 is closer to k than p_{2n} in the x direction. Then the vertical swath k' of C' containing p_3 intersects the horizontal swath r_1 as well as the horizontal swath r_2 . But then r_1, r_2, k, k' form a 2×2 submatrix of M' with four 1's, contradiction to M' being a cycle. $\square$

Theorem 2.4 can be proved by means of the following lemmas.

LEMMA 2.4. *Swath matrices are directed path matrices.*

LEMMA 2.5. *Directed path matrices are totally unimodular.*

A *directed path matrix* is a 0-1 matrix with rows indexed by vertices of a tree with directed edges and columns indexed by some directed paths in the tree and a 1 in the matrix if and only if the corresponding vertex is in the corresponding path. Lemma 2.5 is not new.

Proof of Lemma 2.4. Let P be a rectilinear polygon. Make a tree T with a vertex for each horizontal swath in P and a directed edge from h to g if some grid square of the horizontal swath h lies directly above some grid square of the horizontal swath g . This creates a tree with directed edges. For any vertical swath v of P the set of horizontal swaths of P sharing a grid square with v forms a directed path in T . $\square$

One way to prove Lemma 2.5 is by using the following theorem of Ghouila-Houri which characterizes totally unimodular matrices.

THEOREM 2.6 [G-H]. *A matrix M with rows R and columns C is totally unimodular if and only if each subset S of R can be split into two sets S_1 and S_2 so that the sum of the rows in S_1 minus the sum of the rows in S_2 is a vector in $\{0, +1, -1\}^C$.*

Proof of Lemma 2.5. Let T be a tree with vertices V and with directed edges. Let P be a collection of directed paths in T . By Theorem 2.6 it suffices to show that any subset U of V can be coloured with colours pink and purple so that each path of P has at most one more and at most one fewer pink than purple vertices.

It will be shown by induction on the decreasing size of U that the vertices of U can be coloured pink and purple so that any directed path in T has alternately pink and purple vertices (interspersed with uncoloured vertices). For $U = V$ colour V so that each edge of T is incident with one pink and one purple vertex. Then any directed path in T has alternately pink and purple vertices.

Suppose such a colouring is given for some $U \subseteq V$. To remove vertex u from U , let $v_1, \dots, v_k \in V$ be the vertices for which there are directed edges (u, v_i) in T . Exchange the colours pink and purple in the subtrees containing $v_1, \dots, v_k$ formed by deleting vertex u from T . Uncolour u . Then every directed path in T has alternately pink and purple vertices (interspersed with uncoloured vertices). $\square$

3. A composition preserving firmness. It is more convenient to describe the composition with respect to another formulation of the Boolean basis problem: Any 0-1 matrix M with rows R and columns C can be represented as a bipartite graph $G = G(M)$ with vertices $R \cup C$ and edges $\{(r, c) \in R \times C : M(r, c) = 1\}$. Blocks of M correspond exactly to complete bipartite subgraphs of G ; Boolean bases of M correspond exactly to sets of complete bipartite subgraphs of G covering all the edges. Antiblocks of M correspond exactly to sets of edges in G , no two incident with the same vertex and no two in a 4-cycle. The terms "block," "antiblock," and "Boolean basis" will be applied to bipartite graphs, as will the notation $\beta(G)$ for the minimum size of a Boolean basis of G and $\alpha(G)$ for the maximum size of an antiblock of G . The more general problem of finding $\beta(G)$ for nonbipartite graphs will be discussed later in this section.

Given bipartite graphs G_1 and G_2 with specified vertices v_1 and v_2 (v_i in G_i), construct a bipartite graph $G = G_1(v_1) \otimes G_2(v_2)$ by joining all neighbours of v_1 to all neighbours of v_2 then deleting v_1 and v_2 . See Fig. 3.1. The reverse process is to decompose a bipartite graph at a complete bipartite subgraph whose removal separates the graph. Decomposing involves introducing two new vertices. At the end of this section is a description of what the decomposition means in terms of 0-1 matrices and in terms of rectilinear polygons.

This type of decomposition, applied more generally to directed graphs, is the subject of a paper of Cunningham [C]. He proves a uniqueness result which, specialized to undirected bipartite graphs, says that if a graph G is decomposed into pieces stopping at stars and indecomposable graphs, then these pieces are unique up to relabelling of the introduced vertices. He also gives a polynomial time algorithm to decompose a graph into indecomposable pieces.

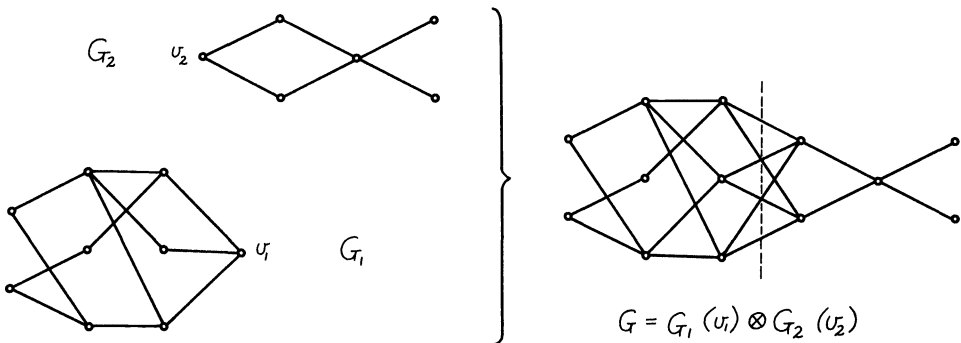


FIG. 3.1

THEOREM 3.1. *Let G_1 and G_2 be bipartite graphs with specified vertices v_1 and v_2 (v_i in G_i). If $G_1, G_1 - v_1, G_2, G_2 - v_2$ all satisfy the Boolean basis min max equality then so does $G = G_1(v_1) \otimes G_2(v_2)$. In particular, if G_1 and G_2 are firm then so is G .*

This will be proved using the following lemmas.

LEMMA 3.2. *If $G = G_1(v_1) \otimes G_2(v_2)$ then*

$$\beta(G) = \min \{ \beta(G_1) + \beta(G_2 - v_2), \beta(G_1 - v_1) + \beta(G_2) \}.$$

LEMMA 3.3. *If $G = G_1(v_1) \otimes G_2(v_2)$ then*

$$\alpha(G) = \max \{ \alpha(G_1) + \alpha(G_2) - 1, \alpha(G_1 - v_1) + \alpha(G_2 - v_2) \}.$$

Proof of Theorem 3.1. It will be proved that $\beta(G) = \alpha(G)$ by using the above lemmas together with the hypotheses that $\beta(H) = \alpha(H)$ for $H = G_1, G_2, G_1 - v_1, G_2 - v_2$.

Suppose first that $\alpha(G_1) > \alpha(G_1 - v_1)$ and $\alpha(G_2) > \alpha(G_2 - v_2)$. Then $\alpha(G) = \alpha(G_1) + \alpha(G_2) - 1$ so $\beta(G_2 - v_2) = \alpha(G_2 - v_2) = \alpha(G_2) - 1$ and $\beta(G_1) + \beta(G_2 - v_2) = \alpha(G_1) + \alpha(G_2) - 1 = \alpha(G)$. Similarly $\beta(G_1 - v_1) + \beta(G_2) = \alpha(G)$. Thus $\beta(G) = \alpha(G)$.

Next suppose that $\alpha(G_1) = \alpha(G_1 - v_1)$. In this case $\beta(G_1) = \beta(G_1 - v_1)$. Then $\beta(G) = \beta(G_1) + \beta(G_2 - v_2) = \alpha(G_1 - v_1) + \alpha(G_2 - v_2) = \alpha(G)$. The case when $\alpha(G_2) = \alpha(G_2 - v_2)$ is symmetric. $\square$

Proof of Lemma 3.2. Let $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, $N_1 = \{v \in V_1: (v, v_1) \in E_1\}$, $N_2 = \{v \in V_2: (v, v_2) \in E_2\}$, $G = (V, E)$.

(i) ($\leq$) Any block of G_1 that uses an edge (x, v_1) can be expanded to a block of G using all edges (x, y) for $y \in N_2$. Thus a Boolean basis of G is produced from the union of a Boolean basis of $G_2 - v_2$ and the result of expanding the blocks of a Boolean basis of G_1 . So $\beta(G) \leq \beta(G_1) + \beta(G_2 - v_2)$. Similarly $\beta(G) \leq \beta(G_1 - v_1) + \beta(G_2)$.

(ii) ($\geq$) Let $\mathcal{B}$ be a minimum size Boolean basis of G . No block $B \in \mathcal{B}$ contains edges of $G_1 - v_1$ and of $G_2 - v_2$. Let $\mathcal{B}_1$ be the blocks of $\mathcal{B}$ containing edges of $G_1 - v_1$ and let $\mathcal{B}_2 = \mathcal{B} - \mathcal{B}_1$. Note that there may be blocks of $\mathcal{B}$ that contain only edges of $N_1 \times N_2$ and these are put in $\mathcal{B}_2$. Any block in $\mathcal{B}_1$ can be compressed to form a block of G_1 by replacing all edges of the form (x, y) for $y \in N_2$ by the single edge (x, v_1) . Let $\mathcal{B}'_1$ be the result of compressing all the blocks in $\mathcal{B}_1$ to form blocks of G_1 . Similarly let $\mathcal{B}'_2$ be the result of compressing all the blocks in $\mathcal{B}_2$ to form blocks of G_2 . If $\mathcal{B}'_1$ is not a Boolean basis of G_1 and $\mathcal{B}'_2$ is not a Boolean basis of G_2 then there exist edges (u_1, v_1) in G_1 and (u_2, v_2) in G_2 not covered by $\mathcal{B}'_1$ and $\mathcal{B}'_2$, respectively. But then in G the edge (u_1, u_2) is not covered by $\mathcal{B}$. Thus either $\mathcal{B}'_1$ is a Boolean basis of G_1 or $\mathcal{B}'_2$ is a Boolean basis of G_2 . In the first case $\mathcal{B}$ splits into a Boolean basis of G_1 —namely $\mathcal{B}'_1$ —and a Boolean basis of $G_2 - v_2$ formed by deleting edges containing v_2 from blocks of $\mathcal{B}'_2$. In the second case $\mathcal{B}$ splits into a Boolean basis of G_2 —namely $\mathcal{B}'_2$ —and a Boolean basis of $G_1 - v_1$ formed by deleting edges containing v_1 from blocks of $\mathcal{B}'_1$. Thus $\beta(G) \geq \min \{ \beta(G_1) + \beta(G_2 - v_2), \beta(G_1 - v_1) + \beta(G_2) \}$. $\square$

Proof of Lemma 3.3. Let $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, $N_1 = \{v \in V_1: (v, v_1) \in E_1\}$, $N_2 = \{v \in V_2: (v, v_2) \in E_2\}$, $G = (V, E)$.

(i) ($\geq$) An antiblock of $G_1 - v_1$ and an antiblock of $G_2 - v_2$ can be combined to produce an antiblock of G . So $\alpha(G) \geq \alpha(G_1 - v_1) + \alpha(G_2 - v_2)$. Now consider making an antiblock of G from an antiblock $\mathcal{A}_1$ of G_1 and an antiblock $\mathcal{A}_2$ of G_2 . If $\mathcal{A}_1$ contains an edge (x, v_1) and $\mathcal{A}_2$ contains an edge (v_2, y) then combine these two edges to the single edge (x, y) in G . Otherwise if $\mathcal{A}_1$ contains an edge (x, v_1) or $\mathcal{A}_2$ contains an edge (v_2, y) , delete that one edge. Thus $\alpha(G) \geq \alpha(G_1) + \alpha(G_2) - 1$.

(ii) ($\leq$) Let $\mathcal{A}$ be a maximum size antiblock of G . If all edges of $\mathcal{A}$ are either in $G_1 - v_1$ or in $G_2 - v_2$ then $\mathcal{A}$ splits into an antiblock of $G_1 - v_1$ and an anti-

block of $G_2 - v_2$. Otherwise, $\mathcal{A}$ contains one edge $(x, y) \in N_1 \times N_2$. The remaining edges split into an antiblock $\mathcal{A}_1$ of $G_1 - v_1$ and an antiblock $\mathcal{A}_2$ of $G_2 - v_2$. Then $\mathcal{A}_1 \cup \{(x, v_1)\}$ is an antiblock of G_1 and $\mathcal{A}_2 \cup \{(v_2, y)\}$ is an antiblock of G_2 . So $\alpha(G) \leq \max \{ \alpha(G_1 - v_1) + \alpha(G_2 - v_2), \alpha(G_1) + \alpha(G_2) - 1 \}$. $\square$

A generalization of the Boolean basis problem is to cover the edges of a graph (bipartite or not) by a minimum number of complete bipartite subgraphs. This generalization may be of algorithmic significance but is vacuous as far as questions of firmness are concerned. It is easy to show that if a graph and all its induced subgraphs satisfy the Boolean basis min max equality in this more general sense, then the graph must be bipartite since the min max equality fails for any odd cycle.

Section 4 uses Cunningham's decomposition algorithm to develop a polynomial time algorithm to solve the Boolean basis problem on a matrix given such algorithms to handle the indecomposable parts of the matrix.

The remainder of the present section contains descriptions of the composition in terms of matrices, and for the special case of rectilinear polygons.

In terms of matrices: starting with matrices M_1 on rows R_1 and columns C_1 , and M_2 on rows R_2 and columns C_2 , and a specified $r^* \in R_1$, $c^* \in C_2$, create a matrix M on rows $(R_1 - \{r^*\}) \cup R_2$ and columns $C_1 \cup (C_2 - \{c^*\})$ where

$$M(r, c) = \begin{cases} M_1(r, c) & \text{if } r \in R_1 - \{r^*\}, c \in C_1 \\ M_2(r, c) & \text{if } r \in R_2, c \in C_2 - \{c^*\} \\ 1 & \text{if } r \in R_2, c \in C_1 \text{ and } M(r, c^*) = M(r^*, c) = 1 \\ 0 & \text{otherwise.} \end{cases}$$

See Fig. 3.2.

For the special case of rectilinear polygons: starting with rectilinear polygons P_1 and P_2 , with a specified row swath r in P_1 consisting of k_r grid squares and separating P_1 into P_1^0 and P_1^1 , and with specified column swath c in P_2 consisting of k_c grid squares and separating P_2 into P_2^0 and P_2^1 , build a rectilinear polygon P starting with a rectangle k_r grid squares wide and k_c grid squares high, and attaching the subpolygons P_1^0 , P_1^1 , P_2^0 , P_2^1 on the appropriate sides as shown in Fig. 3.3.

Note that this construction will not in general produce a polygon, due to the potential overlap of pieces of P_1 with pieces of P_2 . It is perhaps more appropriate to consider the

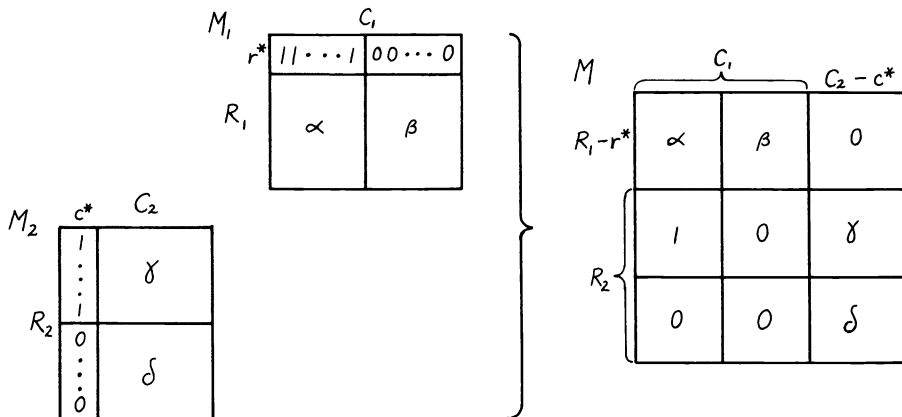


FIG. 3.2

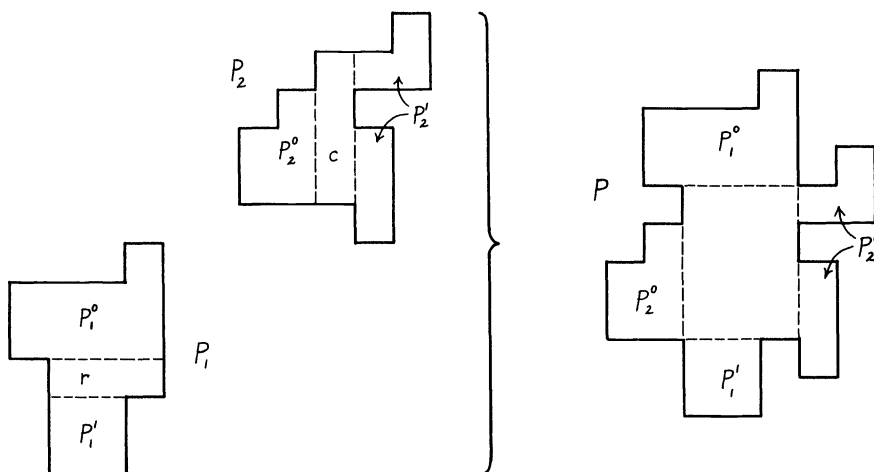


FIG. 3.3

decomposition. In order to decompose one needs a rectangle R in P that is a union of grid squares, has all four corners on the boundary of P , but not both horizontal sides and not both vertical sides completely contained in the boundary of P . A rectilinear polygon can be decomposed in polynomial time simply by trying all possibilities for R ; note that a swath matrix has only polynomially many blocks.

4. Algorithms. The Boolean basis problem will still be considered in terms of bipartite graphs as described at the start of § 3. The present section contains a polynomial time algorithm to solve the Boolean basis problem on any graph composed as described in § 3 from graphs on which the problem can already be solved. More precisely, a polynomial time algorithm will be described to solve the Boolean basis problem on any class $\mathbf{G}$ of bipartite graphs that is *hereditary*—i.e., closed under taking subgraphs—and for which there exists a polynomial time algorithm $\mathcal{A}$ to solve the Boolean basis problem on indecomposable graphs of $\mathbf{G}$. In particular, if there are some hereditary classes $\mathbf{G}_1, \dots, \mathbf{G}_k$ for which polynomial time Boolean basis algorithms are known, then there is a polynomial time algorithm to solve the Boolean basis problem on the closure under composition of the union of $\mathbf{G}_1, \dots, \mathbf{G}_k$.

One necessary step is to decompose efficiently. Cunningham in [C] gave an algorithm to decompose a graph into indecomposable pieces. For a graph on n vertices the algorithm takes time $O(n^3)$. (Aside: The resulting pieces are not necessarily unique. However, Cunningham shows that if, during the decomposition, a star is not decomposed further, then the resulting pieces are unique. Note that for the purpose of solving the Boolean basis problem it is fine to stop at stars since they are blocks. I have described Cunningham's work for the special case of undirected bipartite graphs only; in fact, he deals more generally with directed graphs.)

Recall from Lemma 3.2 that if $G = G_1(v_1) \otimes G_2(v_2)$ then

$$\beta(G) = \min \{ \beta(G_1) + \beta(G_2 - v_2), \beta(G_1 - v_1) + \beta(G_2) \}.$$

The most obvious approach to finding $\beta(G)$ —to decompose and recurse on $G_1, G_1 - v_1, G_2, G_2 - v_2$ —involves solving too many subproblems and cannot be implemented in polynomial time.

An approach that *is* implementable in polynomial time is as follows: decompose G into $G_1(v_1) \otimes G_2(v_2)$ in such a way that G_1 is indecomposable (or a star); find

$\beta(G_1)$ using the hypothesized algorithm $\mathcal{A}$, and find $\beta(G_1 - v_1)$ recursively; if $\beta(G_1) = \beta(G_1 - v_1)$, add this quantity to the cumulative total and recurse on $G_2 - v_2$, and if $\beta(G_1) > \beta(G_1 - v_1)$ add $\beta(G_1 - v_1)$ to the cumulative total and recurse on G_2 . To justify this, note that if $\beta(G_1) = \beta(G_1 - v_1)$ then $\beta(G_1) + \beta(G_2 - v_2)$ cannot exceed $\beta(G_1 - v_1) + \beta(G_2)$; and if $\beta(G_1) > \beta(G_1 - v_1)$ then $\beta(G_1 - v_1) + \beta(G_2)$ cannot exceed $\beta(G_1) + \beta(G_2 - v_2)$.

To carry out this approach, first use Cunningham's algorithm to decompose the graph G completely into pieces that are indecomposable or stars. Let $\mathcal{G}$ be the resulting collection of indecomposable and star graphs. Each graph in $\mathcal{G}$ has vertices that are either original vertices of G or new vertices introduced by the decomposition. New vertices come in pairs, no pair lying together in the same graph. Each original vertex of G appears in exactly one graph of $\mathcal{G}$.

CLAIM 4.1. *Starting with the collection $\mathcal{G}$ and successively composing any two members of the collection that contain a pair of new vertices yields G , regardless of the order in which the compositions are performed.*

Note that $\mathcal{G}$ must contain a graph G_1 with only one new vertex v_1 . By composing in an order that keeps G_1 until last, it can be seen that $G = G_1(v_1) \otimes G_2(v_2)$ where v_2 is v_1 's mate and G_2 is the result of composing together the graphs in $\mathcal{G}' \equiv \mathcal{G} - \{G_1\}$. Now apply the approach outlined above. Find $\beta(G_1)$ using the hypothesized algorithm $\mathcal{A}$. Apply the algorithm recursively to find $\beta(G_1 - v_1)$. (Note that $G_1 - v_1$ need not be indecomposable, but still lies in the hereditary class $\mathbf{G}$). If $\beta(G_1) > \beta(G_1 - v_1)$ recurse on G_2 by continuing the algorithm on $\mathcal{G}'$, regarding v_2 as an original vertex. If $\beta(G_1) = \beta(G_1 - v_1)$ recurse on $G_2 - v_2$; a decomposition of $G_2 - v_2$ can be obtained by finding the graph F of $\mathcal{G}'$ which contains v_2 , decomposing $F - v_2$ into indecomposable and star pieces, and replacing F in $\mathcal{G}'$ by these pieces.

Timing: Let G have n vertices. Consider a tree with root labelled by G , and children of the root labelled by the graphs of $\mathcal{G}$, and where any graph of $\mathcal{G}$ that is stripped of a vertex and decomposed further during the course of the algorithm is given children labelled by the pieces of that decomposition, etc. The number of calls to Cunningham's decomposition algorithm is bounded by the number of nodes in the tree. The number of calls to the hypothesized Boolean basis algorithm $\mathcal{A}$ is bounded by twice the number of nodes in the tree. The tree has no vertex with a single child, so the number of nodes is of the same order as the number of leaves. If each new vertex introduced while decomposing is regarded as a pseudo-node formed by grouping together some of the original vertices, then in the leaves of the tree each original vertex appears at most once as itself and once in a pseudo-node. Thus, since each leaf of the tree is labelled by a graph of at least two vertices, the number of leaves is at most n . Thus the tree has $O(n)$ nodes, and the algorithm uses $O(n)$ calls to $\mathcal{A}$, and $O(n)$ calls to Cunningham's algorithm. In total this makes $O(n)$ calls to $\mathcal{A}$ (on graphs of no more than n vertices) and $O(n^4)$ additional work.

Proof of Claim 4.1. It suffices to show that the order of two successive compositions can be reversed—i.e., that $(A(v_1) \otimes B(v_2))(w_1) \otimes C(w_2) = A(v_1) \otimes (B(w_1) \otimes C(w_2))(v_2)$, where A is a graph containing vertex v_1 , C is a graph containing vertex w_2 , and B is a graph containing vertices v_2 and w_1 . It is clear that the left-hand graph G_L and the right-hand graph G_R have the same vertex set. If (v_2, w_1) is not an edge of B , then the edges of G_L not appearing in A , B , or C are the edges joining neighbours of v_1 with neighbours of v_2 and the edges joining neighbours of w_1 with neighbours of w_2 . G_R is the same. If (v_2, w_1) is an edge of B , then the above edges are not the only new ones; the first composition on the left-hand side joins w_1 with neighbours of v_1 and so the second composition joins neighbours of w_2 with neighbours of v_1 . On the right-hand side the first composition

joins v_2 with neighbours of w_2 and the second composition joins neighbours of v_1 with neighbours of w_2 . There are no other new edges. Thus the resulting graphs G_L and G_R are the same. $\square$

5. Some classes of firm matrices.

THEOREM 5.1 (Györi [Gy]). *Interval matrices are firm.*

Recall that a 0-1 matrix is an *interval matrix* if there is an ordering of its columns making the 1's in each row consecutive. Alternate proofs of Györi's result can be found in [FK] and [Lu]. A polynomial time algorithm to recognize interval matrices and to find an appropriate ordering of the columns can be found in [FG] and [Tu]. A polynomial time algorithm to solve the Boolean basis problem on interval matrices can be found in [FK]. As mentioned in § 2, any rectilinear polygon that is x -convex or y -convex has an interval swath matrix.

The second class to be considered is the class of *blockless* matrices. A 0-1 matrix is *blockless* if it has no blocks of size 2×2 or more—i.e., the only blocks are contained in some row or column of the matrix. Note that any cycle matrix of size 3×3 or more is blockless but not an interval matrix.

LEMMA 5.2. *Any blockless matrix is firm.*

Proof. Let M be a 0-1 matrix with no block of size 2×2 (or more). Then the only blocks of M consist of 1's from the same row or from the same column, and antiblocks of M are simply sets of 1's of M , no two in the same row or column. Translating to the bipartite graph $G(M)$, a Boolean basis is a vertex cover and an antiblock is a matching. König's theorem says that in any bipartite graph the minimum size of a vertex cover is equal to the maximum size of a matching. Thus M is firm. $\square$

There is a polynomial time algorithm to solve the Boolean basis problem for blockless matrices since this is exactly the problem of finding a minimum vertex cover in a bipartite graph (see [L]). It is trivial to recognize blockless matrices in polynomial time. A rectilinear polygon has a blockless swath matrix if and only if the polygon contains no rectangle of size 2×2 or more.

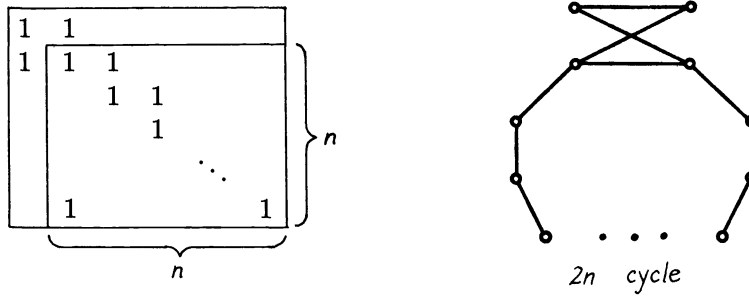
A larger class of firm matrices is formed by taking the closure under composition of the union of the class of interval matrices and the class of blockless matrices. Membership in this class can be tested in polynomial time by combining a decomposition algorithm with membership tests for the two initial classes. The Boolean basis problem can be solved in polynomial time for the class using the method described in § 4. It would be interesting to have other characterizations—for example in terms of forbidden submatrices—of this class, or of the closure of just the interval matrices.

The remainder of this section is about the closure of the class of blockless matrices. These will be called *completely decomposable matrices*. Any completely decomposable matrix is either blockless or decomposable.

LEMMA 5.3. *The 0-1 matrix M is completely decomposable if and only if it does not have a submatrix M^n for any $n \geq 2$, where M^n is as shown in Fig. 5.1.*

Proof of Lemma 5.3. Any submatrix of a completely decomposable matrix is completely decomposable. Thus to show that a completely decomposable matrix has no submatrix M^n , it suffices to note that M^n is not blockless and not decomposable.

Conversely, suppose that the matrix M has no submatrix M^n . If M is blockless, fine. So suppose that M has a maximal block with rows R' and columns C' , each with at least two members. It will be shown that M can be decomposed at this block. It suffices to show in terms of the corresponding bipartite graph that there is no path from an $r \in R'$ to a $c \in C'$ using no edge of $R' \times C'$. Consider a shortest such path $r, v_1, v_2, \dots, v_{2n}, c$ with $n \geq 1$. Since the path is shortest, no v_i will be in R' or C' , and there will be no

FIG. 5.1. The matrix M^n and the corresponding bipartite graph.

edge from r to a v_i or from a v_i to c or from a v_i to a v_j . Since the block was maximal there must be a vertex $r' \in R'$ not adjacent to v_1 (otherwise v_1 could be added to C'). Since the path is shortest there will be no edge from r' to a v_i . Similarly, there will be a vertex $c' \in C'$ not adjacent to any v_i . But then $r', r, v_1, \dots, v_{2n}, c, c'$ form a submatrix M^n . $\square$

Before turning to the implications of this lemma for rectilinear polygons, some relationships to other matrix classes will be described. A 0-1 matrix is *balanced* if it has no $n \times n$ cycle submatrices for n odd, $n \geq 3$. A 0-1 matrix is *strongly balanced* [CHI], [CRa] if it and any matrix resulting from changing one 1 to a 0 are balanced. It is easy to see using Lemma 5.3 that any strongly balanced matrix is completely decomposable.

If a matrix is totally balanced and completely decomposable then, since it has no $n \times n$ cycle submatrices for $n \geq 3$, the submatrices M^n are already excluded except for M^2 . Totally balanced matrices with no M^2 submatrix are called *plaid matrices* in [Lu2], and *γ -acyclic matrices* in [F] (where they are used for acyclic data bases). A fast recognition algorithm was given in [Lu2].

Consider now the rectilinear polygons whose swath matrices are completely decomposable. One such polygon, neither x - nor y -convex, is shown in Fig. 5.2. Since swath matrices are totally balanced (Theorem 2.2) these rectilinear polygons are exactly those whose swath matrices have no submatrix M^2 . The following lemma expresses this condition directly in terms of the polygon.

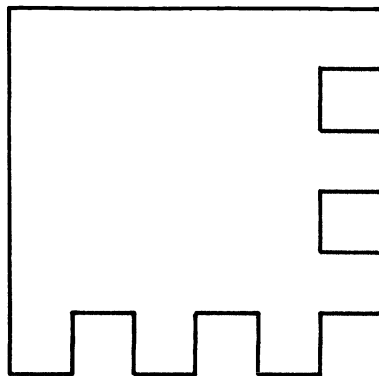


FIG. 5.2

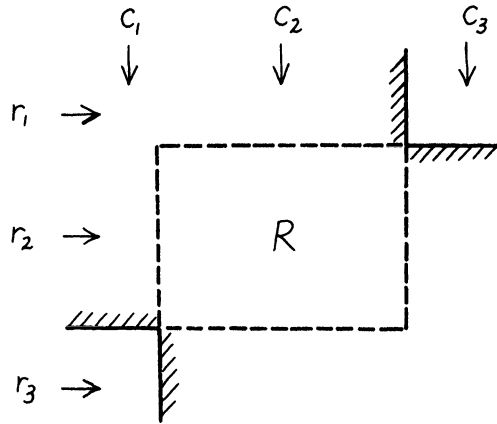


FIG. 5.3

LEMMA 5.4. For a rectilinear polygon P the following three conditions are equivalent:

- (i) P has a completely decomposable swath matrix.
- (ii) P 's swath matrix has no M^2 submatrix.
- (iii) There is no rectangle inside P having two opposite corners and nothing else in contact with the boundary of P .

Proof. By the comments above it remains to prove the equivalence of (ii) and (iii).

(ii) $\Rightarrow$ (iii). Suppose P is a rectilinear polygon that contains a rectangle R touching the boundary of P exactly at two opposite corners. Things must be as shown in Fig. 5.3 or a rotation of this. Let c_2 be a vertical swath containing points interior to R . Let c_1 be the vertical swath containing points just to the left of R , and let c_3 be the vertical swath containing points just to the right of R . Similarly, let r_2 be a horizontal swath containing points interior to R , let r_1 be the horizontal swath containing points just above R , and let r_3 be the horizontal swath containing points just below R . In the swath matrix of P these swaths form the submatrix M^2 .

(iii) $\Rightarrow$ (ii). Suppose the swath matrix M of the rectilinear polygon P has a submatrix M^2 . Name the rows and columns of this M^2 submatrix as shown in Fig. 5.4. The corresponding horizontal and vertical swaths of P will be called by the same names.

Since the horizontal swath b intersects the vertical swaths A , B , and C , it must intersect them in some order. Suppose B is *not* in the middle. Then one of the horizontal swaths a or c intersects the outer two vertical swaths but not the middle one. This is impossible since P does not have holes. Thus B is in the middle. Symmetrically, the horizontal swath b is between the other two horizontal swaths. Suppose without loss of generality that A is to the left of B , and a above b .

Let v_1, v_2, v_3, v_4 be the vertical grid lines through the left side of A , the left side of B , the right side of B , and the right side of C , respectively. Let h_1, h_2, h_3, h_4 be the horizontal grid lines through the top side of a , the top side of b , the bottom side of b , and the bottom side of c , respectively. See Fig. 5.5.

	A	B	C
a	1	1	0
b	1	1	1
c	0	1	1

FIG. 5.4

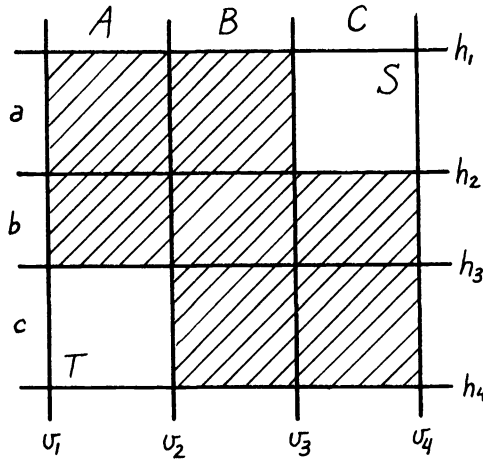


FIG. 5.5

These lines create nine rectangles. The three rectangles between v_2 and v_3 are contained in B ; the three rectangles between h_2 and h_3 are contained in b . The top left rectangle is contained in the polygon P , because all four of its sides are contained in P . Similarly, the bottom right rectangle is contained in P . This leaves the top right rectangle—call it S —and the bottom left rectangle—call it T . It cannot happen that C goes as high as h_1 and a goes as far right as v_4 ; otherwise, these two swaths would intersect. Therefore, there must be a point below h_1 and to the left of v_4 outside P . Let s be a bottom-most left-most boundary point of P inside S . (s may be on the bottom or left side of S , but cannot be on the top or right sides.) s must be a vertex of P . Similarly, let t be a top-most right-most boundary point of P inside the bottom left rectangle T . t will be a vertex of P . Let R be the rectangle determined by s and t . R is contained in P , and R contains the intersection of B and b , so it is not degenerate. R touches the boundary of P precisely at s and t . $\square$

6. Perfect graphs. This section relates questions on firm matrices to questions on perfect graphs.

A graph G is *perfect* if for every induced subgraph H of G the maximum size of an independent set in H is equal to the minimum size of a covering of the vertices of H by cliques. (An *independent set* in a graph is a set of vertices, no two joined by an edge. A *clique* is a set of vertices every two joined by an edge.)

The problems of finding a minimum Boolean basis and a maximum antiblock in a 0-1 matrix M can be expressed as problems of finding a minimum cover by cliques and a maximum independent set, respectively, in a graph; in particular, define a graph $G_1(M)$, called the *1's graph* of M , on vertex set $I(M)$ with an edge between two 1's of M if and only if they are together in some block of M . See Fig. 6.1 for an example. Obviously, independent sets in $G_1(M)$ correspond to antiblocks in M .

CLAIM 6.1. *Maximal blocks in M correspond to maximal cliques of $G_1(M)$.*

Proof. Clearly a block in M yields a clique in $G_1(M)$. In the other direction let K_1 be a maximal clique in $G_1(M)$, corresponding to a set K of 1's in M . Let $R'(C')$ be the rows (columns) of M containing 1's of K . It will be shown that $K = R' \times C'$; thus K is a block. In fact it suffices to show that $R' \times C' \subseteq I(M)$ because then $R' \times C'$ corresponds to a clique of $G_1(M)$ containing the maximal clique K_1 , so $K = R' \times C'$. Let $r \in R'$, $c \in C'$; prove that $M(r, c) = 1$. Row r contains a 1 of K , say (r, c') for some $c' \in C'$. If c' is

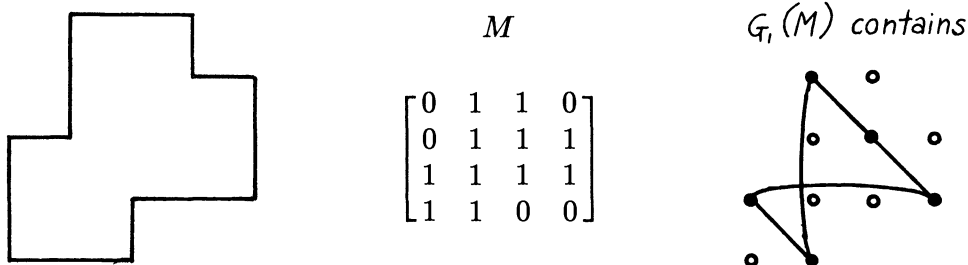


FIG. 6.1

c , fine. Similarly, column c contains a 1 of K , say (r', c) , and if r' is r , fine. The 1's (r, c') and (r', c) are together in some block of M . Any such block contains (r, c) . Thus $M(r, c) = 1$. $\square$

Claim 6.1 implies that Boolean bases of M correspond to coverings of the vertices of $G_1(M)$ by cliques.

Now the Boolean basis min max equality holds for M if and only if the maximum size of an independent set in $G_1(M)$ is equal to the minimum size of a cover of $G_1(M)$ by cliques. Firmness and perfection both ask for these equalities to hold for induced substructures, but the substructures are different. Perfection of $G_1(M)$ clearly implies firmness of M . The converse is not true: for example the matrix in Fig. 6.1 is firm (it is an interval matrix, and the swath matrix of the rectilinear polygon is shown) but its 1's graph contains an induced 5-cycle and so is not perfect.

Perfection of $G_1(M)$ corresponds to the property that for all matrices N arising from M by replacing some 1's of M by the new entry $?$, the minimum number of *generalized blocks* needed to cover the 1's of N is equal to the maximum number of 1's of N , no two contained in a generalized block. A *generalized block* in a matrix of 0's, 1's, and ?'s is defined to be a submatrix containing no 0's. One may think of the ?'s as "don't care elements," which may be used to form generalized blocks but need not be covered. Call a 0-1 matrix M *superfirm* if $G_1(M)$ is perfect.

As will be proved below, the composition of § 3 preserves superfirmness. Consider the classes described in § 5: Interval matrices are not in general superfirm as shown by the example above; blockless matrices are superfirm as can easily be seen, thus completely decomposable matrices are superfirm.

THEOREM 6.2. *Composition preserves superfirmness.*

Proof. A direct proof would not be difficult, but the approach taken here will be to examine what the composition does to the corresponding 1's graph, and show that the same thing can be accomplished using some of the many well-studied graph composition operations that preserve perfection. (See, e.g., [CC], [B, p. 361], [Lo]. In fact the composition of § 3, applied to not necessarily bipartite graphs, is itself a perfection preserving operation [Bi], but this does not help in the present context.) Let M be a 0-1 matrix composed from M^1 and M^2 . Let G, G^1, G^2 be the corresponding bipartite graphs. Using the notation of § 3, $G = G^1(v_1) \otimes G^2(v_2)$ for some vertices v_i of G^i . Let G_1, G_1^1, G_1^2 be the corresponding 1's graphs. Note that vertices of G_1 correspond to edges of G , with two vertices of G_1 adjacent if and only if the corresponding edges of G are incident or in a 4-cycle. Let $T_i, i = 1, 2$ be the vertices of G_1^i corresponding to the edges of G^i adjacent to v_i .

Suppose that M^1 and M^2 are superfirm, i.e., that G_1^1 and G_1^2 are perfect. It will be proved that G_1 is perfect, i.e., that M is superfirm. Observe that G_1 is obtained from

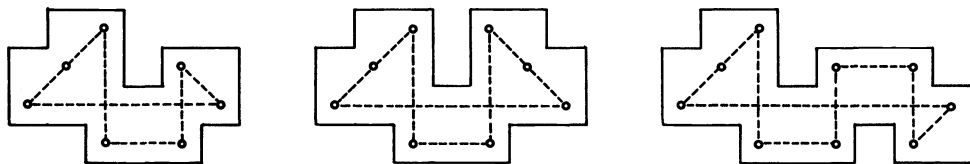


FIG. 6.2

G_1^1 and G_2^2 in the following way: duplicate each vertex x of T_1 to get $|T_2|$ copies of it, each labelled by a pair consisting of x and a unique label from T_2 ; duplicate each vertex y of T_2 to get $|T_1|$ copies of it, each labelled by a pair consisting of y and a unique label from T_2 ; then every pair of labels in $T_1 \times T_2$ labels a pair of vertices, one in each of the resulting two graphs; identify these pairs of vertices to create a single graph. (*Duplicating* a vertex v means adding a new vertex adjacent to v and its neighbours.)

G_1 can thus be obtained from G by the operations of vertex duplication and *clique identification*. This latter operation combines two graphs into one by identifying a clique in one graph with a same-size clique in the other using an arbitrary pairing of their vertices. Both of these operations preserve perfection ([B, p. 361] for clique identification; [Lo] for vertex duplication). Therefore, G_1 is perfect and M is superfirm. $\square$

Although superfirmness is a strictly stronger property than firmness, it would still be interesting to characterize those M 's that are superfirm. Of course a characterization of perfect graphs would suffice. There is a long standing open conjecture of Berge's that a graph is perfect if and only if it has no odd *holes* and no odd *antiholes* as induced subgraphs. A *hole* is a cycle of at least four vertices; an *antihole* is the complement of a hole. Characterizing superfirm matrices M may be easier than characterizing perfect graphs in general, because only graphs that are 1's graphs of some matrix need be considered. An even more special case would be to characterize swath matrices that are superfirm. A beginning might be to characterize (swath) matrices whose 1's graphs have no odd holes or odd antiholes. One result is the following theorem.

THEOREM 6.3. *If the 0-1 matrix M has no 3×3 cycle submatrices then $G_1(M)$ has no odd antihole of size 7 or more.*

Recall the definition of cycle matrices given in § 2. In particular (since swath matrices are totally balanced) the theorem implies that the 1's graph of a swath matrix has no odd antihole of size 7 or more. They may contain any size odd hole, as shown in Fig. 6.2.

Proof of Theorem 6.3. Suppose that M has no 3×3 cycle submatrix but that $G_1(M)$ has an odd antihole A_1 of n vertices where n is odd and at least 7. Consider the set A of 1's of M corresponding to vertices of A_1 . By duplicating rows/columns of M it can be assumed that no two of these 1's lie in the same row or column. Note that duplicating rows/columns will not introduce 3×3 cycles. Let M' be the $n \times n$ submatrix of M

$$\begin{bmatrix} 1 & ? & 1 & 1 & \cdots & 1 & ? \\ ? & 1 & ? & 1 & & 1 & 1 \\ 1 & ? & 1 & ? & \cdots & 1 & 1 \\ 1 & 1 & ? & 1 & & 1 & 1 \\ \vdots & \vdots & & & \ddots & & \vdots \\ 1 & 1 & 1 & 1 & & 1 & ? \\ ? & 1 & 1 & 1 & \cdots & ? & 1 \end{bmatrix}$$

FIG. 6.3. The matrix M' . "?" indicates an entry which may be 0 or 1.

$$\begin{bmatrix} 1 & 0 & 1 & 1 & \cdots & 1 & ? \\ ? & 1 & 1 & 1 & & 1 & 1 \\ 1 & 0 & 1 & 0 & \cdots & 1 & 1 \\ 1 & 1 & 1 & 1 & & 1 & 1 \\ \vdots & & \vdots & & \ddots & & \vdots \\ 1 & 1 & 1 & 1 & & 1 & ? \\ ? & 1 & 1 & 1 & \cdots & ? & 1 \end{bmatrix}$$

FIG. 6.4

consisting of the rows and columns of M which contain 1's of A . Order M' so that the 1's of A appear down the main diagonal. Number the rows and columns of M' . See Fig. 6.3. For all i at least one of $M(i, i+1)$, $M(i+1, i)$ must be 0, where addition of indices is modulo n .

Suppose, without loss of generality, that $M(1, 2) = 0$. Suppose that $M(2, 3) = 0$. Then if $M(6, 5) = 0$ the submatrix formed by rows 1, 2, 6 and columns 2, 3, 5 is a 3×3 cycle; and if $M(5, 6) = 0$ the submatrix formed by rows 1, 2, 5 and columns 2, 3, 6 is a 3×3 cycle. Thus $M(2, 3)$ must be 1. So $M(3, 2)$ must be 0. More generally, exactly one of $M(i, i+1)$, $M(i+1, i+2)$ will be 0, and the 0's must zig-zag as shown in Fig. 6.4. But this is impossible for odd n . $\square$

The final part of this section is about a duality property possessed by perfect graphs but not by firm matrices. Lovász [Lo] proved that a graph is perfect if and only if its complement is perfect. Thus if a graph G is perfect, then for every induced subgraph H of G the maximum size of a clique in H is equal to the minimum size of a covering of the vertices of H by independent sets. (A covering of vertices by independent sets is called a "colouring.") In analogy one might wonder if for a firm matrix M the maximum size of a block of M is equal to the minimum size of a covering of the 1's of M by antiblocks. This problem was considered for the special case of rectilinear polygons by Boucher [Bo] who provided an x - and y -convex polygon as a counterexample.

REFERENCES

- [B] C. BERGE, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973.
- [BCCS] C. BERGE, C. C. CHEN, V. CHVÁTAL, AND C. S. SEOW, *Combinatorial properties of polyominoes*, *Combinatorica*, 1 (1981), pp. 217–224.
- [Bi] R. BIXBY, *A composition for perfect graphs*, *Ann. Discrete Math.*, 21 (1984), pp. 221–224.
- [BND] H. D. BLOCK, N. J. NILSSON, AND R. O. DUDA, *Determination and detection of features in patterns*, *Computer and Information Sciences*, J. T. Tou and R. H. Wilcox, eds., Spartan Books, Washington, D.C., 1964, pp. 75–110.
- [Bo] A. BOUCHER, *It's hard to color antirectangles*, *SIAM J. Algebraic and Discrete Methods*, 5 (1984), pp. 162–163.
- [CKSS] S. CHAIKEN, D. J. KLEITMAN, M. SAKS, AND J. SHEARER, *Covering regions by rectangles*, *SIAM J. Algebraic and Discrete Methods*, 2 (1981), pp. 394–410.
- [CRa] M. CONFORTI AND M. R. RAO, *Structural properties and recognition of restricted and strongly unimodular matrices*, *Math. Programming*, 38 (1987), pp. 17–27.
- [CHI] Y. CRAMA, P. L. HAMMER, AND T. IBARAKI, *Strong unimodularity for matrices and hypergraphs*, *Discrete Appl. Math.*, 15 (1986), pp. 221–239.
- [CR] J. C. CULBERSON AND R. A. RECKHOW, *Covering polygons is hard*, *Proceedings of the 29th Annual IEEE Symposium on Foundations of Computer Science*, 1988, pp. 601–611.
- [C] W. H. CUNNINGHAM, *Decomposition of directed graphs*, *SIAM J. Algebraic and Discrete Methods*, 3 (1982), pp. 214–228.
- [CC] W. H. CUNNINGHAM AND G. CORNUÉJOLS, *Compositions for perfect graphs*, *Discrete Math.*, 55 (1985), pp. 245–254.

- [F] R. FAGIN, *Degrees of acyclicity for hypergraphs and relational database schemes*, J. Assoc. Comput. Mach., 30 (1983), pp. 514–550.
- [FK] D. S. FRANZBLAU AND D. J. KLEITMAN, *An algorithm for covering polygons with rectangles*, Inform. and Control, 63 (1984), pp. 164–189.
- [FG] D. R. FULKERSON AND O. A. GROSS, *Incidence matrices and interval graphs*, Pacific J. of Math., 15 (1965), pp. 835–855.
- [GJ] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, 1979.
- [Gi] J. F. GIMPEL, *The minimization of spatially-multiplexed character sets*, Comm. ACM, 17 (1974), pp. 315–318.
- [G-H] A. GHOUILA-HOURI, *Caractérisation des matrices totalement unimodulaires*, C.R. Acad. Sci. Paris, 254 (1962), pp. 1192–1194.
- [Gy] E. GYÖRI, *A minimax theorem on intervals*, J. Combin. Theory Ser. B, (1984), pp. 1–9.
- [Ki] K. H. KIM, *Boolean Matrix Theory and Applications*, Monographs and Textbooks in Pure and Applied Mathematics 70, Marcel Dekker, New York, 1982.
- [L] E. LAWLER, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart, and Winston, New York, 1976.
- [Lo] L. LOVÁSZ, *Normal hypergraphs and the perfect graph conjecture*, Discrete Math., 2 (1972), pp. 253–267.
- [Lu] A. LUBIW, *A weighted min-max relation for intervals*, J. Combin. Theory Ser. B, submitted.
- [Lu2] ———, *Doubly lexical orderings*, SIAM J. Comput., 16 (1987), pp. 854–879.
- [M] W. J. MASEK, *Some NP-complete set covering problems*, unpublished manuscript, referenced in [GJ], p. 232].
- [MC] C. MEAD AND L. CONWAY, *Introduction to VLSI Systems*, Addison-Wesley, Reading, MA, 1980.
- [St] L. J. STOCKMEYER, *The minimal set basis problem is NP-complete*, IBM Research Report RC 5431, May 1975.
- [Tu] A. TUCKER, *A structure theorem for the consecutive 1's property*, J. Combin. Theory Ser. B, 12 (1972), pp. 153–162.

A THEORY OF PROPORTIONAL REPRESENTATION*

JEAN-LUC PETIT† AND ERIC TÉROUANNE†

Abstract. Integer proportional sharing problems mainly arise in electoral studies: apportionment of seats among states, or repartition of seats among the candidate lists in a ballot. This problem of fair sharing has given rise to various mathematical studies. These studies, culminating in Balinski and Young's *Fair Representation, Meeting the Ideal of One Man, One Vote* [Yale University Press, New Haven, CT, 1982], fail to yield a global and satisfactory theory of proportional representation because they almost exclusively focus on a particular family of "multiplicative processes." In order to build such a theory, several new concepts are introduced and two different approaches are considered to the problem, an algorithmic one and a geometric one. Both approaches prove the existence of a family of additive sharing processes in parallel to multiplicative ones, and the geometric approach allows both families to merge into a new wide family of processes containing all classical methods as particular cases. A series of properties is also studied that look desirable for such sharing processes. Introduction of an order structure over measure spaces illuminates classical results about some of these properties and allows the definition and study of a powerful tool for the comparison of sharing processes.

Key words. proportional representation, fair representation, apportionment, integer sharing

AMS(MOS) subject classification. 90

Introduction. Proportional representation is a very natural and widespread arithmetical problem, characterized by both a great variety of solutions and a certain lack of theoretical study. It typically arises in electoral situations. For instance, the 1787 Constitution of the United States of America stipulates that seats at the House of Representatives should be apportioned among the states in proportion to their respective populations. But it does not make precise what *in proportion to* exactly means. During the last two centuries, the distribution of U.S. population among states changed tremendously, and every ten years, with every new census, the problem of choosing a *fair* repartition of seats has been posed again. It gave rise to a number of stormy debates and to the application of several different solutions. A survey of this history is provided in a recent book by Balinski and Young [2]. A list ballot is another example of a proportional integer sharing problem. It is formally equivalent to the above problem of apportionment when replacing states by lists and populations by numbers of votes. An example of an integer sharing problem in a completely different field is the problem of selecting pixels in order to draw a line in computer graphics.

Many mathematicians, including such famous ones as Condorcet, Morse or Von Neumann, have been involved in the study, invention or discussion of various proportional methods [2]–[8], [11], [14]–[16]. In [2], Balinski and Young study a particular class of methods, the *divisor methods*, which generalize a great variety of classical methods. But all these authors failed to give an overall description of the proportional representation problem. In the present paper, we introduce large classes of *integer sharing processes* (or *processes* for short) which contain and generalize all classical methods. Then we derive various properties of these processes and the means for comparing them.

A proportional representation problem arises when, being given a set E of, say, p states, a measure μ of their populations, and a number n of seats, we look for an integer valued measure ν over E , of total mass n and proportional to μ , that fulfills

$$|\nu| = \sum_{e \in E} \nu(e) = n \quad \text{and} \quad \nu(e) = \lambda \mu(e) \quad (e \in E) \quad \text{for some } \lambda.$$

* Received by the editors January 16, 1989; accepted for publication (in revised form) June 15, 1989.

† Département de Mathématiques et Informatique Appliquées, Université Paul Valéry, Montpellier, France.

Except for very rare values of μ and n , the exact solution of these equations is not integer valued. Thus we must content ourselves with an *approximately proportional* solution. In this paper we give two different and nonequivalent definitions for *approximately proportional* (the reader is referred to Guilbaud [6] for a general and exciting discussion of the role of the adverb *approximately* in mathematics and in various applied domains).

Our first definition of approximate proportionality is expressed in terms of classical optimization theory: we define a *representation criterion* and look for a minmax or max-min sharing with respect to it. This model gives rise to the notion of *fair sharing processes*. In § 1 we give conditions for existence and unicity of such optimal sharings and derive a general algorithm for their computation. In § 2 we introduce two natural families of representation criteria and hence of fair sharing processes, namely *additive* and *multiplicative* processes. These families contain and generalize all previously known methods.

Each classical method usually has been given a particular algorithmic definition. Such definitions entail a serious risk of confusion since people do not always realize that different algorithms can give rise to the same method ([2], [12], [13]). In § 2 we relate several of these classical algorithms to our definition and generalize them to the whole families of additive or multiplicative processes.

Our second definition of approximate proportionality is expressed (§ 3) in geometrical terms: we look for a tessellation (or convex partition) of the simplex of positive measures with total mass 1. This gives rise to the notion of *mixed process*. Additive and multiplicative processes are subclasses of mixed processes and most of their features appear much easier to grasp under this geometric form.

In § 4 we define a series of elementary properties that look desirable for a process and characterize the mixed processes possessing each of them. In § 5, introducing a powerful comparison tool on measure spaces allows us to define and characterize some extra properties. The same comparison tool allows us to formalize in § 6 the notion of *bias* and state several results concerning the comparison of processes.

Finally, we suggest a series of directions for possible extensions of this work and introduce the notion of *random sharing process*.

In order to clarify exposition, all demonstrations are collated in an appendix.

1. Fair sharing. The very idea of proportional sharing suggests that if two measures of population, μ and μ' , say, over the same set of states $\mathbf{E}$ are proportional to one another, we should associate with both of them the same integer sharing ν of n seats. Thus we can assume that every population measure is normalized, and we put $\mu \in \Sigma_1$, the space of real measures over $\mathbf{E}$ fulfilling $\mu(\mathbf{E}) = 1$. In electoral applications it seems natural to assume that μ belongs to the simplex Σ_1^+ of positive measures. Nevertheless, we shall see that some sharing processes extend to the whole space Σ_1 .

An integer sharing ν of n is any element of the space Σ_n^Z of integer measures over $\mathbf{E}$ fulfilling $\nu(\mathbf{E}) = n$. In electoral applications, one is usually restricted to positive measures. This restriction, as well as more general ones, are taken into account by the notion of *threshold processes* developed at the end of this section.

We can now define an *integer sharing process*, or *process* for short, as any mapping P from some domain $\mathbf{D}$ in $\Sigma_1 \times \mathbf{Z}$ into Σ^Z such that for every $(\mu, n) \in \mathbf{D}$, $\nu = P(\mu, n)$ belongs to Σ_n^Z . Much like in physics, we shall refer to $\Sigma_1 \times \mathbf{Z}$ as *space-time*, Σ_1 being *space* and $\mathbf{Z}$ being *time*.

In fact, because of tie situations, a process P can be brought to associate with some (μ, z) in space-time several, rather than one, single fair sharing. Such situations prove to be rare, and in order to simplify our theory we deal with them separately.

The algorithmic definition of a sharing process proceeds from the choice of a *representation index* and a *fairness criterion*. The representation index ε will depend on

the population μ and the number of seats n . At any given point (μ, n) in space-time, $\varepsilon_{(\mu,n)}(e, k)$ will measure how well state e , with population $\mu(e)$, is represented by k seats among n . The fairness criterion will select a sharing which gives every state, as far as possible, the same quality of representation, i.e., the sharing ν which yields the best balanced set of values for the $\varepsilon_{(\mu,n)}(e, \nu(e))$'s.

Formally, a *representation index* ε associates with every point (μ, n) in space-time a mapping $\varepsilon_{(\mu,n)}$ from $\mathbf{E} \times \mathbf{Z}$ into a totally ordered set, fulfilling:

- (i) for any $e \in \mathbf{E}$, $k' < k$ implies $\varepsilon_{(\mu,n)}(e, k') < \varepsilon_{(\mu,n)}(e, k)$.
- (ii) for any $e, e' \in \mathbf{E}$ and $k \in \mathbf{Z}$ there exists $k' \in \mathbf{Z}$ such that $\varepsilon_{(\mu,n)}(e, k) < \varepsilon_{(\mu,n)}(e', k')$.

Property (i) means that the greater the number of seats, the better the representation. If in a given sharing of n seats, state e is given k and state e' is given k' of them, the relation $\varepsilon_{(\mu,n)}(e, k) < \varepsilon_{(\mu,n)}(e', k')$ means that this sharing favors state e' more than state e . Property (ii) expresses the fact that in such a situation it is always possible to obtain the converse by redistributing seats.

For instance, representation indices associated with greatest remainders method and greatest means method are, respectively, denoted by ε_0^0 and ε_0^1 (notations justified in § 2) and defined by:

$$\varepsilon_{0(\mu,n)}^0(e, k) = k - n\mu(e) \quad \text{and} \quad \varepsilon_{0(\mu,n)}^1(e, k) = k/n\mu(e).$$

In § 2 we consider a great variety of representation indices generalizing the above two.

Since we always utilize a representation index $\varepsilon_{(\mu,n)}$ at a fixed point (μ, n) in space-time, we eventually write ε instead of $\varepsilon_{(\mu,n)}$ for short.

A representation index being given, we can consider two *fairness criteria*, namely minmax and maxmin: Let

$$\varepsilon^*(\nu) = \max_{e \in \mathbf{E}} \varepsilon(e, \nu(e))$$

and

$$\varepsilon_*(\nu) = \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e)).$$

A sharing ν of $\Sigma_n^{\mathbf{Z}}$ is a *minmax* sharing with respect to representation index ε at point (μ, n) if $\varepsilon_*(\nu')$ takes its minimum value when $\nu' = \nu$. A sharing ν of $\Sigma_n^{\mathbf{Z}}$ is a *maxmin* sharing with respect to representation index ε at point (μ, n) if $\varepsilon^*(\nu')$ takes its maximum value when $\nu' = \nu$.

For instance, the greatest remainders method and the greatest means method consist in choosing minmax sharings with respect to their respective representation indices. Except for the greatest remainders index, minmax and maxmin criteria with respect to a given index ε are not equivalent. We nevertheless restrict our study to minmax criterion because of Lemma 1.

LEMMA 1. ν is minmax with respect to ε if and only if ν is maxmin with respect to ε' defined by:

$$\varepsilon'(e, k) = \varepsilon(e, k+1), \quad e \in \mathbf{E}, \quad k \in \mathbf{Z}.$$

This result immediately follows from the following lemma.

LEMMA 2. ν is minmax with respect to ε if and only if

$$(1) \quad \max_{e \in \mathbf{E}} \varepsilon(e, \nu(e)) \leq \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e) + 1).$$

ν is maxmin with respect to ε if and only if

$$\min_{e \in E} \varepsilon(e, \nu(e)) \geq \max_{e \in E} \varepsilon(e, \nu(e) - 1).$$

In the demonstration of this lemma we may make use of the *exchange operator* $X(e', e'')$ which transforms a sharing ν into the sharing $\nu' = X(e', e'')(\nu)$ obtained by transferring a seat from e' to e'' :

$$\begin{aligned} \nu'(e') &= \nu(e') - 1, \\ \nu'(e'') &= \nu(e'') + 1, \end{aligned}$$

and

$$\nu'(e) = \nu(e) \quad \text{for every } e \neq e', e''.$$

From now on we call a minmax sharing with respect to ε a *fair sharing*, and we can prove the following theorem.

THEOREM 1. *For any representation index ε and at any point (μ, n) in space-time there exists a fair sharing ν of n with respect to ε .*

Demonstration of Theorem 1 essentially consists in displaying an algorithm for finding a solution to (1). This algorithm proceeds as follows: let ν be an arbitrary element of Σ_n^Z and let e' and e'' be two elements of E fulfilling:

$$(2) \quad \varepsilon(e', \nu(e')) = \max_{e \in E} \varepsilon(e, \nu(e)) \quad \text{and} \quad \varepsilon(e'', \nu(e'')) = \min_{e \in E} \varepsilon(e, \nu(e) + 1).$$

Then either $\varepsilon(e', \nu(e')) \leq \varepsilon(e'', \nu(e''))$ holds and ν is fair, or the converse is true and we replace ν by $X(e', e'')(\nu)$. This algorithm always provides a fair sharing after a finite number of iterations (due to condition (ii)).

Except in case of ties this fair sharing is unique. Ties occur when, for a given sharing ν , inequality (1) comes to an equality. We can prove the following lemma.

LEMMA 3. *If $\max_{e \in E} \varepsilon(e, \nu(e)) = \min_{e \in E} \varepsilon(e, \nu(e) + 1) = \alpha$ holds for a given sharing ν , then the class of tied fair sharings consists in ν and all the $X(e', e'')(\nu)$, where*

$$\varepsilon(e', \nu(e')) = \alpha \quad \text{and} \quad \varepsilon(e'', \nu(e'')) + 1 = \alpha.$$

Thus we can define the *fair sharing process* P^e as the process that associates with any point (μ, n) in space-time the corresponding fair sharing $\nu = P^e(\mu, n)$ (or possibly the class of tied fair sharings).

Let us illustrate the above definition by an example with $n = 6$ seats to be distributed between $p = 3$ states a, b , and c , among which the distribution of populations is

$$\mu(a) = 0.41, \quad \mu(b) = 0.27 \quad \text{and} \quad \mu(c) = 0.22.$$

Table 1 gives the values of greatest remainders and greatest means indices for the first positive values of k .

$\nu = (2, 2, 2)$ is minmax with respect to ε^0 : It is the solution of the greatest remainders method. As a matter of fact we have (Fig. 1(a)):

$$\max_{e \in E} \varepsilon_0^0(e, \nu(e)) = \varepsilon_0^0(c, 2) = 0.44 \leq 0.60 = \varepsilon_0^0(a, 3) = \min_{e \in E} \varepsilon_0^0(e, \nu(e) + 1).$$

The same sharing is also maxmin with respect to ε^0 :

$$\min_{e \in E} \varepsilon_0^0(e, \nu(e)) = \varepsilon_0^0(a, 2) = -0.40 > -0.56 = \varepsilon_0^0(c, 1) = \max_{e \in E} \varepsilon_0^0(e, \nu(e) - 1).$$

TABLE 1
Values of two representation indices in an example with $p = 3$ states
and $n = 6$ seats.

States	a	b	c	a	b	c
3	0.60	0.96	1.46	1.25	1.47	1.92
2	-0.40	-0.04	0.44	0.83	0.98	1.28
1	-1.40	-1.04	-0.56	0.42	0.49	0.64
0	-2.40	-2.04	-1.56	0	0	0
$\varepsilon(e, k)$	ε_0^0 (greatest remainders)			ε_0^1 (greatest means)		

On the other hand the minmax, with respect to ε^1 , and thus the fair sharing for the greatest means method, is $\nu' = (3, 1, 2)$ since we have (Fig. 1 (b)):

$$\max_{e \in \mathbf{E}} \varepsilon_0^1(e, \nu'(e)) = \varepsilon_0^1(a, 3) = 1.25 \leq 1.28 = \varepsilon_0^1(c, 2) = \min_{e \in \mathbf{E}} \varepsilon_0^1(e, \nu'(e) + 1).$$

But $(3, 1, 2)$ is not maxmin with respect to ε^1 since:

$$\min_{e \in \mathbf{E}} \varepsilon_0^1(e, \nu'(e)) = \varepsilon_0^1(c, 1) = 0.64 < 0.83 = \varepsilon_0^1(a, 2) = \max_{e \in \mathbf{E}} \varepsilon_0^1(e, \nu'(e) - 1).$$

The maxmin fair sharing with respect to ε^1 is $(2, 2, 2)$:

$$\min_{e \in \mathbf{E}} \varepsilon_0^1(e, \nu(e)) = \varepsilon_0^1(a, 2) = 0.83 > 0.64 = \varepsilon_0^1(c, 1) = \max_{e \in \mathbf{E}} \varepsilon_0^1(e, \nu(e) - 1).$$

At this stage it should be stressed that, even if we usually choose some real function as our representation index ε , we only exploit the order structure of its values. A consequence is that many real representation indices boil down to the same fair process. Let us say that two indices ε and ε' are *equivalent* if (at any point of space-time) we have:

$$\varepsilon(e, k) \leq \varepsilon(e', k') \Leftrightarrow \varepsilon'(e, k) \leq \varepsilon'(e', k').$$

Then obviously $P^{\varepsilon'} = P^\varepsilon$. For example, if ε is a real index and f is an increasing real function, $\varepsilon' = f \circ \varepsilon$ is equivalent to ε . In particular, this accounts for the fact that, as mentioned above, minmax and maxmin criteria are equivalent in the case of greatest remainders index: we know by Lemma 1 that maxmin criterion with $\varepsilon(e, k) = k - n\mu(e)$ is equivalent to minmax criterion with $\varepsilon'(e, k) = \varepsilon(e, k + 1) = (k + 1) - n\mu(e) = \varepsilon(e, k) + 1$, which is an increasing function of $\varepsilon(e, k)$.

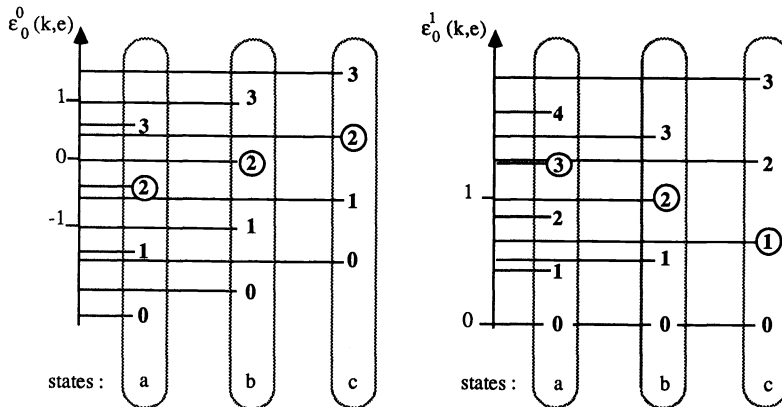


FIG. 1. Fair sharings with respect to greatest remainders (a) and greatest means (b) indices.

We can also consider replacing the order relation over the set of values of ε by the dual (or inverse) order. The new minmax criterion is merely the ancient maxmin criterion, and thus equivalent to a minmax criterion with the original order and ε' instead of ε . This is what happens in the case of a real index when replacing, for instance, ε by $-\varepsilon$.

Electoral applications of sharing processes impose restrictions upon the values of $\nu(e)$. First, any number of seats is always supposed to be positive. In addition, in various circumstances the legislator imposes minimum or maximum values: for instance, every state should have at least one representative, or no list should have as much as one half of the seats, etc.

Such restrictions are taken into account by the definition of *threshold processes*: Let τ and τ' denote two integer measures over E . We look for a sharing with respect to thresholds (τ, τ') , that is, an element of

$$\Sigma_n^Z(\tau, \tau') = \{ \nu \in \Sigma_n^Z \mid \tau(e) \leq \nu(e) \leq \tau'(e), \forall e \in E \}.$$

Of course the total masses of τ and τ' should fulfill

$$\tau(E) \leq n \leq \tau'(E).$$

Then an element ν of $\Sigma_n^Z(\tau, \tau')$ is a fair *sharing with respect to thresholds* τ and τ' if it fulfills, instead of (1), the following inequality:

$$\text{Max} \{ \varepsilon(e, \nu(e)) \mid \tau(e) \leq \nu(e) \} \leq \text{Min} \{ \varepsilon(e, \nu(e) + 1) \mid \nu(e) \leq \tau'(e) \}.$$

It is easy to modify our algorithm of Theorem 1 by applying the same restrictions, and thus to prove that, for every representation index ε , there exists a fair sharing with respect to thresholds (τ, τ') .

2. Additive and multiplicative fair sharing processes. There are two great families of representation indices, and hence of fair sharing processes. We call them *additive* and *multiplicative* indices or processes. The prototype of additive indices is that of greatest remainders method, $\varepsilon_0^0(e, k) = k - n\mu(e)$. Every additive index is a difference between some functions of the number of seats k and the *ideal share* $n\mu(e)$. The prototype of multiplicative indices is that of greatest means method, $\varepsilon_0^1(e, k) = k/n\mu(e)$. Every multiplicative index is a ratio of such functions.

Any additive or multiplicative representation index is defined by the choice of a particular *scale*. A *scale* is an increasing mapping s from $\mathbf{Z}$ into $\mathbf{R}$ fulfilling:

$$\lim_{z \rightarrow +\infty} s(z) = +\infty \quad \text{and} \quad \lim_{z \rightarrow -\infty} s(z) = -\infty.$$

With every scale s is associated (Fig. 2) a *rounding procedure*, that is, an increasing mapping $[]_s$ from $\mathbf{R}$, into $\mathbf{Z}$, defined by

$$[x]_s = z \Leftrightarrow s(z) \leq x < s(z+1).$$

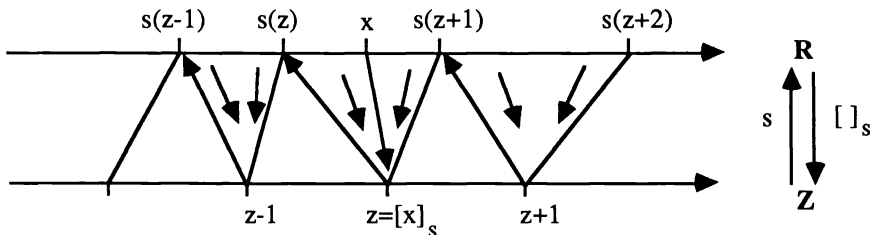


FIG. 2. A scale s and the associated rounding procedure $[]_s$.

The scale utilized in greatest remainders and greatest means methods is identity, and the associated rounding procedure is the integer part.

The *additive representation index* associated with a scale s is

$$\varepsilon_s^0(\mu(e), z) = s(z) - s(n)\mu(e).$$

The *multiplicative representation index* associated with a scale s is

$$\varepsilon_s^1(\mu(e), z) = s(z)/s(n)\mu(e).$$

The fair sharing processes associated with ε_s^0 is the *additive process* P_s^0 .

The fair sharing processes associated with ε_s^1 is the *additive process* P_s^1 .

Any additive index ε_s^0 can be extended to the whole space Σ_1 of positive or negative measures. It follows that an additive process P_s^0 is defined over $\Sigma_1 \times \mathbf{Z}$. The same is not true in the multiplicative case: in order for ε_s^1 to fulfill condition (i) in the definition of a representation index, μ should be a positive measure and $s(n)$ a positive number. Thus P_s^1 is only defined over

$$\Sigma_1^+ \times \{n \in \mathbf{Z} \mid s(n) > 0\}.$$

From now on we always assume $s(n) > 0$. Mixed processes introduced in § 3 are defined over a space of measure that is intermediate between Σ_1 and Σ_1^+ .

A major difference between additive and multiplicative indices is that the latter depend upon time when the former do not. As a matter of fact, since $s(n)$ is positive, ε_s^1 is equivalent to $\varepsilon_s^1(e, k) = k/\mu(e)$. This difference will prove to entail serious consequences when we compare the properties of additive and multiplicative processes (§§ 4 and 5).

We shall in particular make use of the family of *affine scales* s_r ($r \in \mathbf{R}$) defined by $s_r(z) = z + r$. We shall denote by $[]_r$ and P_r^i ($i = 0, 1$) the associated affine rounding procedure and affine fair sharing processes. Here are some classical examples of couples $(s_r, []_r)$:

$$\begin{aligned} s_0(z) &= z & [x]_0 &= \text{integer part of } x; \\ s_{-1/2}(z) &= z - \frac{1}{2} & [x]_{-1/2} &= \text{closest integer to } x; \\ s_{-1}(z) &= z - 1 & [x]_{-1} &= \text{least integer greater than } x. \end{aligned}$$

Except for P_0^0 , the greatest remainders method, no additive method has ever been used or described in the literature. Most classical multiplicative processes are associated with scales s fulfilling $z - 1 \leq s(z) \leq z$. Balinski and Young describe this family in [2] under the name of *divisor methods*. Less than ten of them have actually been utilized, and all but two are, in fact, affine processes. The last two are associated with two members of a subfamily of rather sophisticated scales making use of generalized means: $s(z) = [z(z-1)]^{1/2}$ (associated with the geometric mean) for Hill's method and $s(z) = (z^2 - z)/(z - \frac{1}{2})$ (associated with the harmonic mean) for Dean's method (cf. [2]).

Most classical methods have been discovered several times and described by different algorithms whose equivalence seldom has been noted. This is why they usually bear several names (Table 2).

In Theorem 1, we derived an algorithm for the computation of fair sharing processes. Balinski and Young [2], after Huntington [8], derived another algorithm that works for every multiplicative process. By reformulating it, we can extend it to additive processes. It basically proceeds as follows: If the chosen scale is identity, that is, $s = s_0$, the values $n\mu(e)$ constitute the *ideal shares* (also called *quotas*). They usually are not integers, and

TABLE 2
Classical methods with their scales and their various appellations.

Scales	Additive process	Multiplicative processes
affine:		
$s(n) = n$	Greatest remainders, Hamilton, Vinton	Greatest means, Jefferson, Greater divisors, d'Hondt, Hagenbach-Bischoff
$s(n) = n - \frac{1}{2}$		Sainte-Laguë, Webster, Greatest fractions, Odd numbers
$s(n) = n - 0.6$		Condorcet
$s(n) = n - 1$		Adams, Least divisors
$s(n) = n + 1$		Imperiali
non affine:		
$s(n) = n(n - 1)/(n^{1/2} - \frac{1}{2})$		Dean, Harmonic mean
$s(n) = [n(n - 1)]^{1/2}$		Hill, Geometric mean, Huntington, Equal proportions

it would look fair to replace them by their integer parts $[n\mu(e)]_0$. But the sum of these integers is usually less than n . Then one can modify the ideal shares, either additively or multiplicatively, so that the sum of the rounded modified shares be exactly n .

Greatest remainders method P_0^0 is equivalent to searching for a real γ such that

$$\sum_{e \in E} [\gamma + n\mu(e)]_0 = n,$$

and allocating to each $e \in E$ $\nu(e) = [\gamma + n\mu(e)]_0$ seats.

Similarly, greatest average method P_0^1 is equivalent to searching for a real γ such that

$$\sum_{e \in E} [\gamma n\mu(e)]_0 = n,$$

and allocating to each $e \in E$ $\nu(e) = [\gamma n\mu(e)]_0$ seats.

If we replace identity by an arbitrary scale s , the same idea leads to searching, by trial and error, for an *adder* γ^0 (in the additive case) or a *multiplier* γ^1 (in the multiplicative case) such that $\Sigma[\gamma^0 + s(n)\mu(e)]_s$ or $\Sigma[\gamma^1 s(n)\mu(e)]_s$ be exactly n , and we can prove the following theorem.

THEOREM 2. *For any scale s and any point (n, μ) in space-time there exist real numbers γ^0 and γ^1 such that*

$$P_s^0(\mu, n)(e) = [\gamma^0 + s(n)\mu(e)]_s \quad \text{and} \quad P_s^1(\mu, n)(e) = [\gamma^1 s(n)\mu(e)]_s.$$

In the multiplicative case, our *multiplier* γ is proportional to the inverse of Balinski and Young's *divisor* (Balinski and Young [2]). It is worth noting that the adder (or multiplier) does not have a definite value. In fact, there exists an interval in which every value is a possible adder (or multiplier).

Another classical algorithm, initially derived by Jefferson, then by d'Hondt for the greatest average method, can be extended to the whole class of multiplicative processes. Contrary to the above ones, it is a sequential algorithm, first attributing the first seat, then the second, and so on, without ever questioning the previous attributions. As a matter of fact, if $\nu = P_s^1(\mu, n)$ and e' and e'' fulfill (2), we have for every e in E :

$$\varepsilon(e, \nu(e)) \leq \varepsilon(e', \nu(e')) \leq \varepsilon(e'', \nu(e'')) + 1 \leq \varepsilon(e, \nu(e) + 1).$$

Hence $P_s^1(\mu, n + 1)$ is the sharing ν' defined by

$$\nu'(e) = \nu(e), \quad e \neq e'' \text{ and } \nu'(e'') = \nu(e'') + 1.$$

Thus from any particular fair sharing $P_s^1(\mu, n)$ we can sequentially compute the fair sharing $P_s^1(\mu, n')$ of any integer n' greater than n . The same holds for $n' < n$ (as long as $s(n') > 0$), $P_s^1(\mu, n - 1)$ being the sharing ν'' defined by

$$\nu''(e) = \nu(e), \quad e \neq e' \text{ and } \nu'(e') = \nu(e') - 1.$$

This sequential algorithm can be described as follows in the case $n' > n$: first arrange the values $\varepsilon(e, k)$, $e \in E$, $k > \nu(e)$ in increasing order, then successively attribute one extra seat to each state occurring in this list until the n' seats are attributed. In the case $n' < n$, operate similarly by suppressing one seat to each state occurring in the list obtained according to the decreasing order of the values $\varepsilon(e, k)$, $e \in E$, $k < \nu(e)$. A possible initial value for this algorithm is the uniform sharing of $n = p[0]_s$ seats (where $p = |E|$ is the number of states). As a matter of fact, the following lemma holds.

LEMMA 4. *If $n = p[0]_s$ fulfills $s(n) > 0$, the fair sharing $P_s^i(\mu, n)$ of n seats is, for any μ , the uniform sharing ν defined by $\nu(e) = [0]_s$ for every e in E .*

In the particular case of an affine scale s_r , $[0]_r = [-r]_0$, thus $s_r(p[0]_r) > 0 \Leftrightarrow r < 0$. Most classical affine scales fulfill $-1 \leq r < 0$. In this case our sequential algorithm boils down to the d'Hondt algorithm: arrange the quantities $\varepsilon_r^1(e, k)$, $e \in E$, $k > 0$, in increasing order, and give list e one seat at every occurrence of a quantity concerning e among the first n elements of this sequence.

Existence of such a sequential algorithm ensues from the fact that multiplicative indices do not depend on time. On the other hand, additive indices depend on time and no sequential algorithm can be provided for an additive process.

3. Geometric definition of mixed processes. In the first two sections, additive and multiplicative processes have been given an algebraic definition. In the present section, they are given a geometric description that allows us to merge both families into a wider family called *mixed processes* and eventually to formalize and analyse their properties (§§ 4 and 5).

A *tessellation* of $\Sigma_+^\dagger$ is a family $T = \{T(i), i \in I\}$ of disjoint open cells $T(i)$ such that the union of their closures is $\Sigma_+^\dagger$.

A fair sharing process P^e determines at any time $n \in \mathbb{Z}$ a tessellation

$$T^e = \{T^e(\nu), \nu \in \Sigma_n^Z\}$$

of $\Sigma_+^\dagger$ where the cell $T^e(\nu)$ is defined by:

$$T^e(\nu) = \{\mu \in \Sigma_+^\dagger \mid P^e(\mu, n) = \nu\}.$$

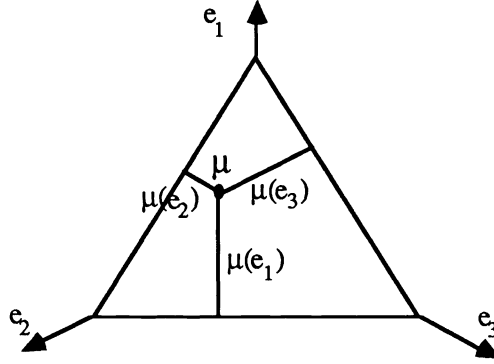
In order to visualize these tessellations, when $n = 3$, we make use of the classical triangular representation of $\Sigma_+^\dagger$ (Fig. 3).

Two neighboring cells are associated with two sharings ν' and ν'' differing only by one seat: there are two states e' and e'' such that $\nu'' = X(e', e'')(\nu')$ (Fig. 4).

The boundary between $T^e(\nu')$ and $T^e(\nu'')$ is constituted of points μ of space where ν' and ν'' are tied fair sharings. The junction point of three mutually neighboring cells is a point μ at which there are three tied fair sharings. All of this is easily generalized to the case $p > 3$ but it cannot be easily represented graphically.

At any point of the boundary between $T^e(\nu')$ and $T^e(\nu'')$ the following equation holds:

$$\varepsilon(e', \nu(e') + 1) = \varepsilon(e'', \nu(e'')).$$

FIG. 3. Triangular representation of Σ_1^+ .

If we take for ε an additive or multiplicative representation index, we obtain a linear equation. The boundary between two neighboring cells is thus a line (when $p = 3$), or more generally a hyperplane in Σ_1^+ . This is expressed by the following theorem.

THEOREM 3. *The cell $T_s^0(\nu)$ associated with ν by additive process P_s^0 is defined by inequalities:*

$$I_s^0(\nu, e', e'') = s(\nu(e') + 1) - s(n)\mu(e') - s(\nu(e'')) + s(n)\mu(e'') \geq 0, \quad e', e'' \in E.$$

The cell $T_s^1(\nu)$ associated with ν by multiplicative process P_s^1 is defined by inequalities:

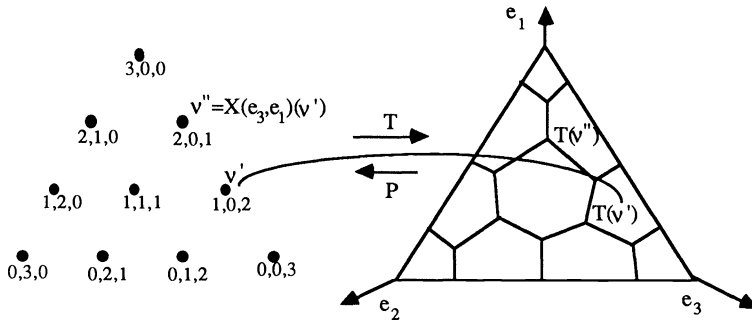
$$I_s^1(\nu, e', e'') = s(\nu(e') + 1)\mu(e'') - s(\nu(e''))\mu(e') \geq 0, \quad e', e'' \in E.$$

Let us denote by $H_s^i(\nu, e', e'')$ the hyperplane separating $P_s^i(\nu)$ and $P_s^i(\nu')$ where $\nu' = X(e', e'')(\nu)$. Its equation is $I_s^i(\nu, e', e'') = 0$. Figure 5 shows, in the case where $p = 3$, $H_s^0(\nu, e', e'')$ and $H_s^1(\nu, e', e'')$ for fixed values of s and ν , $e' = e_3$ and $e'' = e_1$. Their intersection N is a subhyperplane defined by

$$\mu(e') = s(\nu(e') + 1)/s(n); \quad \mu(e'') = s(\nu(e''))/s(n).$$

The equation of $H_s^0(\nu, e', e'')$ is of the type $\mu(e') - \mu(e'') = \text{constant}$, thus this hyperplane is parallel to the symmetry hyperplane S defined by $\mu(e') - \mu(e'') = 0$.

The equation of $H_s^1(\nu, e', e'')$ is of the type $\mu(e')/\mu(e'') = \text{constant}$, thus this hyperplane contains subhyperplane O defined by $\mu(e') = \mu(e'') = 0$.

FIG. 4. A tessellation T associated with a sharing process P when $n = 3$, $p = 3$.

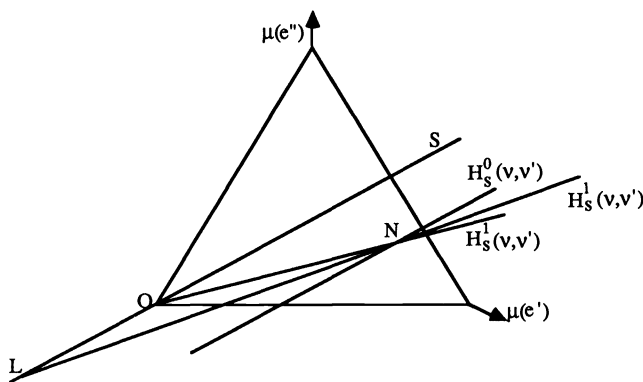


FIG. 5. Separating hyperplanes.

Every cell $T_s^i(\nu)$ is a convex polytope. Instead of defining it by the equations of its faces, we can define it by its extremal points. The coordinates of such points are extremal (maximal or minimal) values of coordinates of points in the cell. The following corollary of Theorem 3 gives these extremal values in the case of an affine process P_r^i .

COROLLARY 1. *Every point μ in the cell $T_r^0(\nu)$ ($r \in \mathbf{R}$) fulfills the inequalities:*

$$[p(\nu(e) - 1) + r + 1]/p(n + r) < \mu(e) < [p(\nu(e) + 1) + r - 1]/p(n + r) \quad \text{for every } e \in \mathbf{E}.$$

Every point μ in the cell $T_r^1(\nu)$ ($r \in \mathbf{R}$) fulfills the inequalities:

$$(\nu(e) + r)/(pr + n + p - 1) < \mu(e) < (\nu(e) + r + 1)/(pr + n + 1) \quad \text{for every } e \in \mathbf{E}.$$

We give in the Appendix the formulas for extremal values in the general case of a nonaffine process. They are much more intricate because extremal values of a coordinate $\mu(e)$ not only depend on $\nu(e)$ but also on the values $\nu(e')$ taken by ν on the other coordinates. The fact that, in the affine case, the extremal values depend only on $\nu(e)$ allows us to write the following corollary.

COROLLARY 2. *If $\nu = P_r^0(\mu, n)$, then*

$$\mu(e) < [pk - p + r - 1]/p(n + r) \Rightarrow \nu(e) < k,$$

$$\mu(e) > [pk + p + r - 1]/p(n + r) \Rightarrow \nu(e) > k.$$

If $\nu = P_r^1(\mu, n)$, then

$$\mu(e) < (k + r)/(pr + n + p - 1) \Rightarrow \nu(e) < k,$$

$$\mu(e) > (k + r + 1)/(pr + n + 1) \Rightarrow \nu(e) > k.$$

Consequences of these inequalities are extensively studied in Petit and T  rouanne [13].

Classes Π^0 of additive processes and Π^1 of multiplicative processes are particular cases of class $\Pi^\lambda = \{P_s^\lambda \mid \lambda \in [0, 1]\}$ of *mixed processes* with parameter $\lambda \in [0, 1]$. The *mixed process* P_s^λ is formally defined as a barycenter between processes P_s^0 and P_s^1 associated with the same scale s :

$$P_s^\lambda = \lambda P_s^1 + (1 - \lambda) P_s^0.$$

More precisely, hyperplane $H_s^\lambda(\nu, e', e'')$ separating cells $T_s^\lambda(\nu)$ and $T_s^\lambda(X(e', e'')(\nu))$ is the barycenter $\lambda H_s^1(\nu, e', e'') + (1 - \lambda)H_s^0(\nu, e', e'')$ and its equation is:

$$I_s^\lambda(\nu, e', e'') = \lambda I_s^1(\nu, e', e'') + (1 - \lambda)I_s^0(\nu, e', e'') = 0, \quad e, e' \in E.$$

This hyperplane contains subhyperplane N (Fig. 5) and sub-hyperplane L of S , defined by:

$$\mu(e) = \mu(e') = (\lambda - 1)/\lambda.$$

Theorem 3 and its corollaries are easily extended to mixed processes, even though formulas look somewhat more complex (see the Appendix).

It should be noted that a mixed process is a fair sharing process only if $\lambda = 0$ or $\lambda = 1$. As a matter of fact, if $\lambda \in]0, 1[$ variables cannot be separated in equation $I_s^\lambda(\nu, e', e'')$; therefore, this equation cannot be put under the form $\varepsilon(e', \nu(e') + 1) - \varepsilon(e'', \nu(e''))$.

Figure 6 shows the tessellations of Σ_1^+ provided by P_0^0 when $n = 3$ (heavy lines) and $n = 4$ (light lines). These tessellations extend to the whole space Σ_1 .

In fact T_0^0 is, for any given n , the Voronoi tessellation (Ahuja and Schachter [1]) associated with the lattice G of points of Σ_1 obtained by projecting Σ_n^Z onto it. A point $G(\nu)$ of this lattice has for coordinates:

$$G(\nu)(e) = \nu(e)/n.$$

This accounts for the well known fact that the sharing yielded by the greatest remainders method is a solution to the least squares problem associated with the classical Euclidean structure of Σ_1 (Burt and Harris [5]). This result can be extended to every affine additive process P_r^0 : the coordinates of the point $G_r(\nu)$ of the corresponding lattice are:

$$G_r(\nu)(e) = (\nu(e) + r/p)/(n + r).$$

Figure 7(a) shows the tessellations of Σ_1^+ by P_0^1 when $n = 3$ (heavy lines) and $n = 4$ (light lines). Figure 7(b) (heavy lines) represents the mixed process $P_{-4/5}^{4/5}$ when $n =$

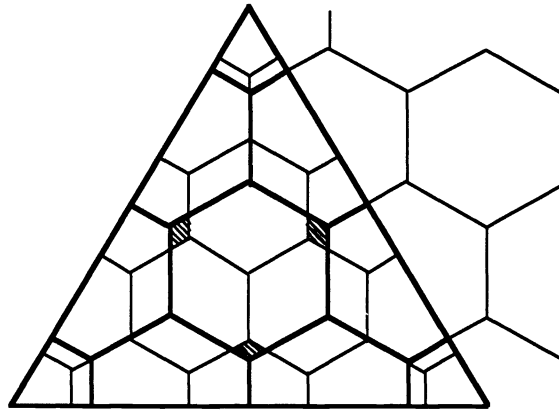


FIG. 6. Tessellations $T_0^0(3)$ (heavy lines) and $T_0^0(4)$ (faint lines).

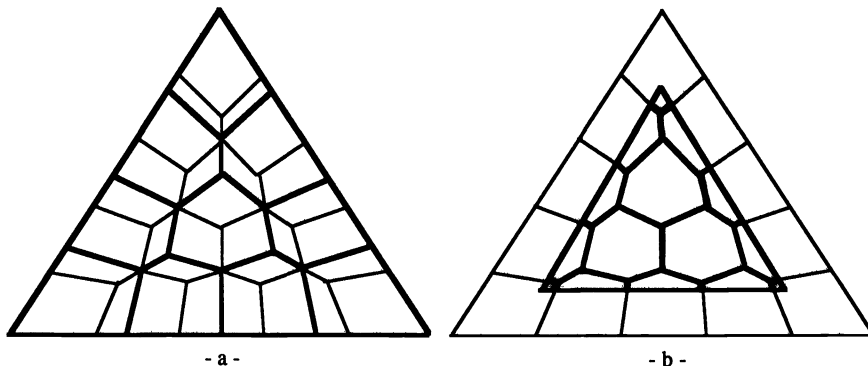


FIG. 7. Tessellations $T_0^1(3)$ (heavy lines) and $T_0^1(4)$ (faint lines) of Σ_1 (a) and tessellation $P_{-4/5}^{4/5}(4)$ of Σ_1 (heavy lines) and $\Sigma^{4/5}$ (faint lines) (b).

4. Every mixed process P_s^λ extends out of Σ_1 to the convex

$$\Sigma^\lambda = \{ \mu \in \Sigma_1 \mid \mu(e) \geq (\lambda - 1)/\lambda \forall e \in \mathbf{E} \}.$$

$\Sigma^{4/5}$ is indicated in light lines in Fig. 7(b).

A series of spatio-temporal transformations (homotheties in Σ_1 , translations in $\mathbf{Z}$) operate into the space of affine processes. Let us denote by A_α the homothety whose centre is the uniform distribution μ_0 over $\mathbf{E}$ ($\mu_0(e) = 1/p$ for every e in $\mathbf{E}$) and whose coefficient is $\alpha > 0$:

$$A_\alpha(\mu) = \alpha\mu + (1 - \alpha)\mu_0.$$

Additive affine processes are deduced from one another by means of the following homotheties.

PROPOSITION 1.

$$P_r^0(\mu, n) = P_{r'}^0(A_\alpha(\mu), n) \Leftrightarrow \alpha = (n + r)/(n + r').$$

In the same way, every mixed affine process P_r^λ can be expressed as the application of a multiplicative process to Σ^λ : an illustration is given by tessellations in faint lines in Figs. 7(a) and 7(b), which are identical.

PROPOSITION 2.

$$P_r^\lambda(\mu, n) = P_r^1(A_\alpha(\mu), n) \Leftrightarrow \alpha = \lambda/[\lambda + p(1 - \lambda)] \quad \text{and} \quad r' = [r + n(\lambda - 1)]/\lambda.$$

Finally any affine multiplicative process P_r^1 with $r \leq -1$ or $r > 0$ can be expressed in terms of an affine multiplicative process $P_{r'}^1$ with $-1 < r' \leq 0$. This result accounts for the fact that if P_r^1 yields a given tessellation of Σ_1^+ at a given time n , and if time increases by a multiple of p , say, kp , then P_{r-k}^1 yields the same tessellation and attributes everywhere and to every $e \in \mathbf{E}$ k seats in addition to those attributed by P_r^1 .

PROPOSITION 3.

$$P_{r-k}^1(\mu, n + kp) = P_r^1(\mu, n) + (k, k, \dots, k).$$

4. First properties of mixed processes: regularity and conditional independence. We have seen that there exists a great variety of integer sharing processes, and it is easy to check that examples give rise to very different results (Balinski and Young [2]; Lacheny, Petit, and Térouanne [9], [10]; Petit and Térouanne [12]). The choice of a convenient process in any particular situation must thus be guided by the knowledge of the different

properties of each process and by the choice of the properties that look most desirable. We define in this section a series of such properties and characterize the mixed processes that possess each one. No one process can possess all properties.

The first desirable property is that P should treat all elements of $\mathbf{E}$ in the same way: A process P is said to be *symmetric* if it is invariable under the action of the group $G(\mathbf{E})$ of permutations over $\mathbf{E}$:

$$P(\sigma\mu, n) = \sigma P(\mu, n) \quad \forall \sigma \in G(\mathbf{E}).$$

In some nonelectoral applications of proportional integer sharing processes, we can look for nonsymmetric processes. This is, for instance, the case when an a priori integer function η over $\mathbf{E}$ is given and when we look for a sharing ν proportional to μ that fulfills $\sum_{e \in \mathbf{E}} \eta(e) \nu(e) = n$, instead of $\sum_{e \in \mathbf{E}} \nu(e) = n$. However, this is beyond our present scope, and we have the following proposition.

PROPOSITION 4. *Every mixed process is symmetric.*

It would appear desirable that, if population of state e is greater than that of state e' , e should receive at least as many seats as e' . A process P is said to be μ -*increasing* if the preorder induced over $\mathbf{E}$ by $\nu = P(\mu, n)$ is compatible with that induced by μ :

$$\mu(e') \leq \mu(e) \Rightarrow \nu(e') \leq \nu(e).$$

PROPOSITION 5. *Every mixed process is μ -increasing.*

The following property expresses the fact that if, in two ballots concerning the same lists, P yields the same sharing of n , if we merge the two ballots by adding up the numbers of votes for each list, P should again yield the same sharing of n . Since the distribution of votes so obtained is a barycenter of the original two distributions, this property is fulfilled by any *convex process*, where a process P is said to be *convex* if for every $n \in \mathbf{Z}$ the associated tessellation T is made of convex cells.

PROPOSITION 6. *Every mixed process is convex.*

All sharing processes actually used are additive or multiplicative processes and are thus convex. Balinski and Young derived, under the name of *quotas methods*, a family of nonconvex processes.

Contrary to the first four properties, the following ones (as well as those introduced in § 5) are not possessed by every mixed process: each of these properties is characteristic of one family, either the additive or the multiplicative processes.

The first of these properties is related with problems of coalitions and manipulations of votes in list ballots. We developed it (Petit and Térouanne [13]) as an answer to a question put forth by Guilbaud (private communication). Assume that μ represents the votes obtained by three lists e , e' , and e'' competing for n seats and that P attributes k seats to list e . If e' and e'' could redistribute their votes between themselves, without changing the score of e , the number of seats attributed to e by P could happen to be different from k . As a matter of fact it is clearly impossible for a symmetric and non-constant process P to associate with a fixed score $\mu(e)$ always the same number of seats $\nu(e)$, whatever the repartition of the remaining votes between the other lists. But some processes happen to produce, under such circumstances, much greater variations than others and therefore favour some kind of manipulation. It would appear desirable to have this variation reduced as much as possible. A process P is said to be *regular* if for every $n \in \mathbf{Z}$:

$$\mu'(e) = \mu(e) \Rightarrow |P(\mu', n)(e) - P(\mu, n)(e)| \leq 1.$$

Regularity of greatest remainders method is immediate if we consider its classical definition: $\mu(e)$ being given, $\nu(e)$ is either $[n\mu(e)]_0$, the integer part of $n\mu(e)$, or

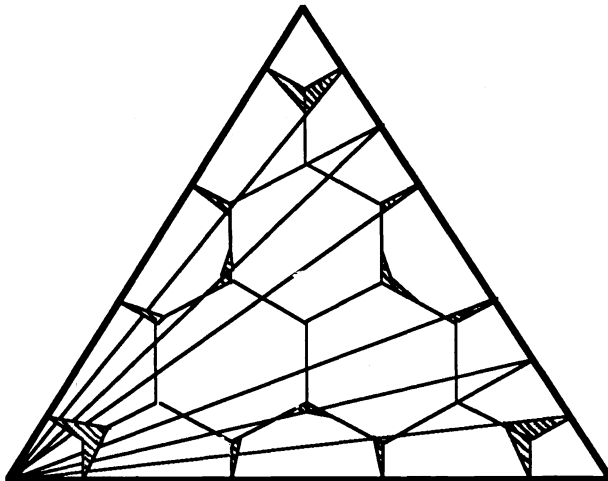


FIG. 8. Illustration of the violation of conditional independence by greatest remainders method.

$[n\mu(e)]_0 + 1$. Regularity of any affine additive process then follows from Proposition 1. We can prove that the following theorem holds.

THEOREM 4. *An affine mixed process P is regular if and only if it is an additive process.*

The following property deals with the problem of *reapportionment*: assume that the sharing of n seats among e , e' and e'' by P is $\nu = (\nu(e), \nu(e'), \nu(e''))$. Then it would be appropriate that the sharing of a $\nu(e) + \nu(e')$ seat between e and e' by the same process P should be $\nu' = (\nu(e), \nu(e'))$ (this property is linked to what Social Choice theoretists call *independence of irrelevant alternatives*). A process P is said to be *conditionally independent* if for every subset E' of E , every $n \in \mathbb{Z}$ and every $\mu \in \Sigma_1^+$ we have

$$P(\mu', n')(e) = P(\mu, n)(e) \quad \forall e \in E',$$

where μ' is the restriction of μ to E' :

$$\mu'(e) = \mu(e) / \mu(E') \quad \forall e \in E',$$

and n' is the total number of seats attributed to E' out of n by P :

$$n' = \sum_{e \in E'} P(\mu, n)(e).$$

THEOREM 5. *A mixed process is conditionally independent if and only if it is a multiplicative process.*

The above result follows from the fact that multiplicative indices depend upon time when additive ones do not. As a matter of fact, considering reapportionment between two states e and e' amounts to applying P to the projection μ' of μ onto $\Sigma_1^+(\{e, e'\})$. This projection fulfills $\mu'(e) = \rho$ and $\mu'(e') = 1 - \rho$ where $\mu(e)/\mu(e') = \rho/(1 - \rho)$. Let $\mu'_0 \in \Sigma_1^+(\{e, e'\})$ be a limit point of the cell corresponding to sharing ν' . The projection hyperplane H_0 going through μ'_0 and subhyperplane $O(\mu(e) = \mu'(e) = 0)$ delimitates into cell $T(\nu)$ a set of points that violate conditional independence (Fig. 8). This subset is empty if H is the corresponding boundary of $T(\nu)$, which is true if and only if P is a multiplicative process.

5. Spatio-temporal increasingness of mixed processes. We now consider a series of properties of *increasingness* with respect to order structures into space-time and integer

sharings space. A first notion of *temporal increasingness* is the following: if n increases, μ and P being fixed, the share of any given state should not decrease. Violation of this property is known as the *Alabama paradox*. Examples of such violations are easily built with any additive fair sharing process (Petit and Térouanne [12]). For instance, hatched parts in Fig. 6 contain points for which the sharing of three and four seats by P_0^0 , respectively, yield $(1, 1, 1)$ and $(0, 2, 2)$ (apart from a permutation of components). In the same figure for any mixed processes P_0^λ , these hatched parts would decrease as λ increases and only vanish when $\lambda = 1$ (Fig. 7(a)).

A weaker notion of *temporal increasingness* consists of saying that if n increases, μ and P being fixed, the cumulated shares of the greatest k states (or lists) should not decrease, whatever k .

In order to formalize this property and eventual properties of *spatial increasingness*, we first need to introduce an order relation into the space of real measures over an ordered set, the *cumulation order*:

Let E be a totally ordered set (the definition could be extended to pre-orders, that is to say orders admitting ties, but we do not consider ties here for the sake of simplicity). The *cumulation order* $\prec$ associated with order $\leq$ is defined in the space of measures over E by:

$$\mu' \prec \mu \Leftrightarrow \mu'(\{e'/e' \leq e\}) \leq \mu(\{e'/e' \leq e\}) \text{ for every } e \text{ in } E.$$

Figure 9 illustrates cumulation order: we denote elements of E by $e_1, e_2, \dots, e_p$, assuming $e_1 \leq e_2 \leq \dots \leq e_p$. Some authors call cumulation order the stochastic order. Curves in Fig. 9 bear different names in different domains: repartition functions, cumulated frequencies, Lorentz curves, concentration curves.

A process P is said to be *weakly temporally increasing* if for every $\mu \in \Sigma_1^+$:

$$n' < n \Rightarrow P(\mu, n') \prec P(\mu, n) \text{ for the order defined on } E \text{ by } e \leq e' \Leftrightarrow \mu(e') \leq \mu(e).$$

A process P is said to be *strongly temporally increasing* if for every $\mu \in \Sigma_1^+$:

$$n' < n \Rightarrow P(\mu, n')(e) \leq P(\mu, n)(e) \quad \forall e \in E.$$

This latter property is equivalent to:

$$n' < n \Rightarrow P(\mu, n') \prec P(\mu, n) \quad \text{for every order } \leq \text{ on } E.$$

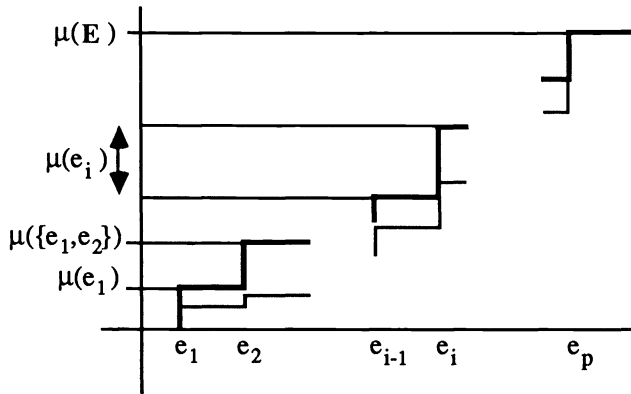


FIG. 9. Cumulation order: $\mu' \prec \mu$ means that the cumulated graph of μ' (faint lines) lies below that of μ (heavy lines).

THEOREM 6. *Every mixed process is weakly temporally increasing.*

A mixed process is strongly temporally increasing if and only if it is a multiplicative process.

Spatial increasingness deals with the behaviour of P when μ changes to μ' in space Σ_1 . We can associate with every such variation a total order $\leq$ into $\mathbf{E}$, $e \leq e'$ expressing the fact that e is less favoured than e' when going from μ to μ' . There are at least two means for doing that: compare the absolute variations $\mu'(e) - \mu(e)$ or compare the relative variations $\mu'(e)/\mu(e)$:

A process P is said to be *additively spatially increasing* if for every $n \in \mathbf{N}$, $\mu, \mu' \in \Sigma_1^+$, $P(\mu, n) \prec P(\mu', n)$ holds for the *additive order* defined on $\mathbf{E}$ by:

$$e \leq e' \Leftrightarrow \mu'(e) - \mu(e) \leq \mu'(e') - \mu(e').$$

A process P is said to be *multiplicatively spatially increasing* if for every $n \in \mathbf{N}$, $\mu, \mu' \in \Sigma_1^+$, $P(\mu, n) \prec P(\mu', n)$ holds for the *multiplicative order* defined on $\mathbf{E}$ by:

$$e \leq e' \Leftrightarrow \mu'(e)/\mu(e) \leq \mu'(e')/\mu(e').$$

Figure 10 illustrates both forms of spatial increasingness: μ' being a fixed element of Σ_1^+ , hatched areas A^0 and A^1 , respectively, represent the sets of measures μ fulfilling:

$$\mu'(e_1) - \mu(e_1) \leq \mu'(e_2) - \mu(e_2) \leq \mu'(e_3) - \mu(e_3) \quad \text{in the additive case (Fig. 10(a)),}$$

$$\mu'(e_1)/\mu(e_1) \leq \mu'(e_2)/\mu(e_2) \leq \mu'(e_3)/\mu(e_3) \quad \text{in the multiplicative case (Fig. 10(b)).}$$

Then let ν' denote the image of μ' by a given process P (Fig. 10(c)). Hatched area B represents the set of sharings ν fulfilling $\nu \prec \nu'$ for the order $e_1 \leq e_2 \leq e_3$. Then P is additively (respectively, multiplicatively) spatially increasing if and only if $P(A^0)$ (respectively, $P(A^1)$) is contained into B .

THEOREM 7. *A mixed process P is additively spatially increasing if and only if it is an additive process.*

A mixed process P is multiplicatively spatially increasing if and only if it is a multiplicative process.

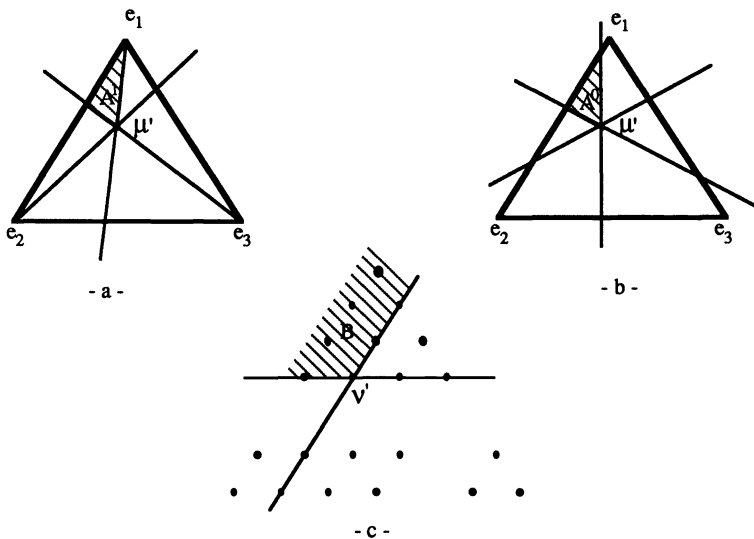


FIG. 10. *Spatial increasingness.*

6. Comparison of sharing processes. The class of sharing processes is endowed with a partial order structure, accounting for the fact that a given process P favors the little ones more than another process P' does, or equivalently P favors the big ones less than P' does. We shall denote this situation by $P \downarrow P'$. This relation, first described by Balinski and Young under the name of *bias* and also studied by Breny [4], can be formalized through the notion of cumulation order:

We say that a process P' favors the little ones less than P does, or $P' \downarrow P$, if at every point (μ, n) of space-time, $P'(\mu, n) \prec P(\mu, n)$ holds for the order defined over $\mathbf{E}$ by: $e' \leq e \Leftrightarrow \mu(e') \leq \mu(e)$.

Within the subfamily of affine processes, some families of processes are totally ordered with respect to this relation. This is expressed by Theorems 8 and 9; the first one generalizes some previous results (Balinski and Young [2]; Brény [4]).

THEOREM 8. For any $\lambda \in [0, 1]$, $P_r^\lambda \downarrow P_{r'}^\lambda \Leftrightarrow r \leq r'$.

THEOREM 9.

$$0 \leq r \Rightarrow P_r^1 \downarrow P_r^0, \quad r \leq -1 \Rightarrow P_r^0 \downarrow P_r^1.$$

It is easy to check with some examples that P_r^0 and P_r^1 are not comparable when r lies between 0 and -1 . All these results can be extended to nonaffine mixed processes.

7. Conclusion. Various problems are left unsolved in proportional integer sharing theory. For instance the problem of finding an axiomatic characterization of mixed processes. It is also possible to extend the notion of sharing process by considering random processes that associate with every point (μ, n) in space-time a probability distribution over Σ_n^Z . An example of random process is supplied by multinomial distribution:

$$P(\mu, n)(\nu) = n! [\Pi_e \nu(e)!] [\Pi_e \mu(e)^{\nu(e)}], \quad \nu \in \Sigma_n^Z, n \in \mathbf{N}.$$

This random process is related to P_0^1 by means of its *mode* (Guilbaud, private communication): Let ν^* , the *mode* of P , be defined by:

$$P(\mu, n)(\nu^*) = \sup_{\nu} P(\mu, n)(\nu).$$

This sharing fulfills the following inequalities:

$$\mu(e)/\mu(e') < (\nu^*(e) + 1)/\nu^*(e') \quad \text{for every } e, e' \text{ in } \mathbf{E}.$$

Thus P_0^1 is the deterministic process that replaces the multinomial distribution associated with (μ, n) by its mode. This result can be extended to any affine multiplicative process.

Appendix: proofs of main results.

Proof of Lemma 2. Let us first prove that (1) implies that ν is minmax with respect to ε . For any $\nu' \neq \nu$ there is some $e' \in \mathbf{E}$ such that $\nu'(e') > \nu(e')$. Then (1) and increasingness of ε entail:

$$\begin{aligned} \max_{e \in \mathbf{E}} \varepsilon(e, \nu(e)) &\leq \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e) + 1) \leq \varepsilon(e', \nu(e') + 1) \\ &\leq \varepsilon(e', \nu'(e')) \leq \max_{e \in \mathbf{E}} \varepsilon(e, \nu'(e)). \end{aligned}$$

Conversely, if ν is minmax with respect to ε and that (1) does not hold, the following holds for some e', e'' in $\mathbf{E}$:

$$\alpha = \max_{e \in \mathbf{E}} \varepsilon(e, \nu(e)) = \varepsilon(e', \nu(e')) > \varepsilon(e'', \nu(e'') + 1) = \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e) + 1).$$

Then $\nu' = X(e', e'')(\nu)$ fulfills:

$$\varepsilon(e', \nu'(e')) = \varepsilon(e', \nu(e')) - 1 < \varepsilon(e', \nu(e')) = \alpha,$$

$$\varepsilon(e'', \nu'(e'')) = \varepsilon(e'', \nu(e'')) + 1 < \alpha \quad \text{and}$$

$$\varepsilon(e, \nu'(e)) = \varepsilon(e, \nu(e)) < \alpha \quad \text{for any } e \neq e', e''.$$

Thus $\text{Max}_{e \in \mathbf{E}} \varepsilon(e, \nu'(e)) < \alpha$, which is impossible since ν is minmax.

The same kind of argument holds in the multiplicative case. $\square$

Proof of Theorem 1. Let $\nu \in \Sigma_n^{\mathbf{Z}}$ and let $H(\nu)$ be the subset of $\mathbf{Z} \times \mathbf{E}$ defined by:

$$H(\nu) = \{(e, k) \mid k \leq \nu(e) \quad \text{and} \quad \varepsilon(e, k) > \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e) + 1)\}.$$

$H(\nu)$ is a finite set because of condition (ii) in the definition of representation index ε . $H(\nu)$ is empty if and only if ν is fair with respect to ε . If $H(\nu)$ is not empty, e' and e'' being defined as in (2), $(e', \nu(e'))$ belongs to $H(\nu)$. Then we modify ν in $\nu' = X(e', e'')(\nu)$ and obtain $\text{Card}(H(\nu')) < \text{Card}(H(\nu))$. As a matter of fact:

$$\text{Max}_{e \in \mathbf{E}} \varepsilon(e, \nu'(e)) \leq \text{Max}_{e \in \mathbf{E}} \varepsilon(e, \nu(e)) \quad \text{and}$$

$$\min_{e \in \mathbf{E}} \varepsilon(e, \nu'(e) + 1) \geq \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e) + 1).$$

Then $(e'', k) \in H(\nu') \Rightarrow k \leq \nu'(e'') = \nu(e'') + 1$ and $\varepsilon(e'', k) > \min_{e \in \mathbf{E}} \varepsilon(e, \nu'(e) + 1) \geq \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e) + 1) = \varepsilon(e'', \nu(e'') + 1)$, which is impossible since ε is an increasing function of k ;

$(e, k) \in H(\nu')$ and $e \neq e'' \Rightarrow k \leq \nu'(e) \leq \nu(e)$ and $\varepsilon(e, k) > \min_{e \in \mathbf{E}} \varepsilon(e, \nu'(e) + 1) \geq \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e) + 1) \Rightarrow (e, k) \in H(\nu)$.

Thus $H(\nu)$ contains $H(\nu')$ and since $(e', \nu(e'))$ does not belong to $H(\nu)$, this inclusion is strict. Hence after a finite number of iterations of this operation, we obtain a fair sharing. $\square$

Proof of Lemma 3. Let $\nu \in \Sigma_n^{\mathbf{Z}}$ be a fair sharing with respect to ε fulfilling:

$$\text{Max}_{e \in \mathbf{E}} \varepsilon(e, \nu(e)) = \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e) + 1) = \alpha.$$

Let $\nu' \neq \nu$ be another element of $\Sigma_n^{\mathbf{Z}}$: $\nu(e') < \nu'(e')$ and $\nu(e'') > \nu'(e'')$ hold for some e', e'' in $\mathbf{E}$. Increasingness of ε implies

$$\varepsilon(e'', \nu'(e'')) + 1 \leq \varepsilon(e'', \nu(e'')) = \alpha = \varepsilon(e', \nu(e')) + 1 \leq \varepsilon(e', \nu'(e')).$$

Then if ν' is fair above, inequalities are equalities. $\square$

Proof of Theorem 2. Let $\nu \in \Sigma_n^{\mathbf{Z}}$ be a fair sharing with respect to ε and let γ fulfill:

$$\text{Max}_{e \in \mathbf{E}} \varepsilon(e, \nu(e)) \leq \gamma < \min_{e \in \mathbf{E}} \varepsilon(e, \nu(e) + 1).$$

Then if $\varepsilon = \varepsilon_s^0$ we have for every e in $\mathbf{E}$:

$$s(\nu(e)) - s(n)\mu(e) \leq \gamma < s(\nu(e) + 1) - s(n)\mu(e) \Leftrightarrow$$

$$s(\nu(e)) \leq \gamma + s(n)\mu(e) < s(\nu(e) + 1) \Leftrightarrow$$

$$[\gamma + s(n)\mu(e)]_s = \nu(e).$$

Similarly if $\varepsilon = \varepsilon_s^1$ we have for every e in $\mathbf{E}$:

$$s(\nu(e))/\mu(e) \leq \gamma < s(\nu(e) + 1)/\mu(e) \Leftrightarrow$$

$$s(\nu(e)) \leq \gamma \mu(e) < s(\nu(e) + 1) \Leftrightarrow$$

$$[\gamma \mu(e)]_s = \nu(e). \quad \square$$

Proof of Lemma 4. $k' \leq [0]_s < k \Rightarrow s(k') \leq 0 < s(k) \Rightarrow \varepsilon_s^1(e', k') \leq 0 < \varepsilon_s^1(e, k)$ for every e, e' in $\mathbf{E}$. Then if $n = p[0]_s$ the uniform sharing ν_0 defined by $\nu_0(e) = [0]_s$ for every e in $\mathbf{E}$ fulfills:

$$\max_{e \in \mathbf{E}} \varepsilon(e, \nu_0(e)) \leq 0 < \min_{e \in \mathbf{E}} \varepsilon(e, \nu_0(e) + 1). \quad \square$$

Proof of Corollary 1 of Theorem 3. We give the proof in the general case of a non-affine scale s , where Corollary 1 becomes:

Every point μ in the cell $T_s^0(\nu)$ fulfills the inequalities:

$$\mu_s^{0-}(\nu, e) < \mu(e) < \mu_s^{0+}(\nu, e) \quad \text{for every } e \in \mathbf{E}, \quad \text{where}$$

$$\mu_s^{0-}(\nu, e) = [(p-1)s(\nu(e)) + s(n) - \sum_{e' \neq e} s(\nu(e') + 1)]/ps(n) \quad \text{and}$$

$$\mu_s^{0+}(\nu, e) = [(p-1)s(\nu(e) + 1) + s(n) - \sum_{e' \neq e} s(\nu(e'))]/ps(n).$$

Every point μ in the cell $T_s^1(\nu)$ fulfills the inequalities:

$$\mu_s^{1-}(\nu, e) < \mu(e) < \mu_s^{1+}(\nu, e) \quad \text{for every } e \in \mathbf{E}, \quad \text{where}$$

$$\mu_s^{1-}(\nu, e) = s(\nu(e))/[s(\nu(e)) + \sum_{e' \neq e} s(\nu(e') + 1)] \quad \text{and}$$

$$\mu_s^{1+}(\nu, e) = s(\nu(e) + 1)/[s(\nu(e) + 1) + \sum_{e' \neq e} s(\nu(e'))].$$

Let $P_s^0(\mu) = \nu$ and $e \in \mathbf{E}$. Then, for any $e' \neq e$, $I_s^0(\nu(e'), \nu(e)) > 0$ holds, thus:

$$\sum_{e' \neq e} I_s^0(\nu(e'), \nu(e)) > 0 \Leftrightarrow$$

$$\sum_{e' \neq e} s(\nu(e') + 1) - s(n)\mu(e') - s(\nu(e)) + s(n)\mu(e) > 0 \Leftrightarrow$$

$$s(n)(p-1)\mu(e) > s(n)\sum_{e' \neq e} \mu(e') + s(\nu(e))(p-1) - \sum_{e' \neq e} s(\nu(e') + 1) \Leftrightarrow$$

$$s(n)p\mu(e) > s(n) + s(\nu(e))(p-1) - \sum_{e' \neq e} s(\nu(e') + 1) \Leftrightarrow$$

$$\mu(e) > \mu_s^{0-}(\nu, e) = [s(n) + s(\nu(e))(p-1) - \sum_{e' \neq e} s(\nu(e') + 1)]/s(n)p.$$

The same computation applied to inequalities $I_s^0(\nu(e), \nu(e')) > 0$ yields $\mu_s^{0+}(\nu, e)$. Similar arguments work in the multiplicative case.

We can state for a mixed process P_s^λ results similar to those of Theorem 3 and its Corollary: $T_s^\lambda(\nu)$ is defined by the inequalities $I_s^\lambda(\nu(e), \nu(e')) \geq 0$, $e, e' \in \mathbf{E}$ and every point μ in the cell $T_s^\lambda(\nu)$ fulfills the inequalities:

$$\mu_s^{\lambda-}(\nu, e) < \mu(e) < \mu_s^{\lambda+}(\nu, e), \quad \text{with}$$

$$\mu_s^{\lambda-} = [\lambda a + (1-\lambda)b]/[\lambda c + (1-\lambda)d], \quad \text{where we put:}$$

$$\mu_s^{1-} = a/c \quad \text{and} \quad \mu_s^{0-} = b/d, \quad \text{and the same for } \mu_s^{\lambda+}. \quad \square$$

Proof of Proposition 1. If $\mu' = A_\alpha(\mu)$ and $e, e' \in \mathbf{E}$, we have:

$$\mu' \in H_{r'}^0(\nu(e), \nu(e')) \Leftrightarrow \nu(e) + 1 - \nu(e') = (n+r')(\mu'(e) - \mu'(e')) = (n+r')\alpha(\mu(e) - \mu(e'))$$

and

$$\mu \in H_r^0(\nu(e), \nu(e')) \Leftrightarrow \nu(e) + 1 - \nu(e') = (n + r)(\mu(e) - \mu(e')).$$

The above two quantities are equal if and only if $\alpha = (n + r)/(n + r')$. $\square$

Proof of Proposition 2. $H_r^1(\nu(e), \nu(e'))$ contains the sub-hyperplane O defined by $\mu(e) = \mu(e') = 0$.

The image of O by A_α , defined by $\mu(e) = \mu(e') = (1 - \alpha)/p$, is confounded with the corresponding sub-hyperplane L of $H_r^\lambda(\nu(e), \nu(e'))$ defined by $\mu(e) = \mu(e') = (\lambda - 1)/\lambda$ if and only if $\alpha = 1 + p(1 - \lambda)/\lambda$.

$H_r^\lambda(\nu(e), \nu(e'))$ also contains the sub-hyperplane N defined by:

$$\mu(e) = [\nu(e) + 1 + r']/(n + r'); \mu(e') = [\nu(e') + r']/(n + r').$$

The image of $H_r^\lambda(\nu(e), \nu(e'))$ by A_α^{-1} is thus defined by:

$$\begin{aligned} & \mu(e)/\mu(e') \\ &= [\alpha^{-1}[\nu(e) + 1 + r']/(n + r') + (1 - \alpha)/p] / [\alpha^{-1}[\nu(e') + r']/(n + r') + (1 - \alpha)/p]. \end{aligned}$$

This image is confounded with $H_r^1(\nu(e), \nu(e'))$, defined by

$$\mu(e)/\mu(e') = [\nu(e) + 1 + r]/[\nu(e') + r]$$

if and only if $r = r' + [(n + r')(1 - \alpha^{-1})/p\alpha]$, which is equivalent to $r' = \lambda r + (\lambda - 1)n$. $\square$

Proof of Proposition 3. The value of $\varepsilon_{r-k}^1(\mu(e), \nu(e))$ at time $n + kp$ and that of $\varepsilon_r^1(\mu(e), \nu(e) - k)$ at time n are both equal to $(\nu(e) + r - k)/\mu(e)$. $\square$

Proof of Proposition 4. This result immediately follows from the symmetry of the minimax criterion defining a fair sharing. $\square$

Proof of Proposition 5. For additive and multiplicative processes, this result immediately follows from the fact that the representation indices $\varepsilon_s^i(\mu(e), z)$ are, for every fixed z , decreasing functions of $\mu(e)$ (under our hypothesis $s(n) > 0$). Proposition 2 extends the result to mixed processes. $\square$

Proof of Proposition 6. This result immediately follows from the geometric definition of mixed processes. $\square$

Proof of Theorem 4. Corollary 1 of Theorem 3 implies, for every $r \in \mathbf{R}$ and if $p > 2$:

$$\mu_r^{0-}(k) < \mu_r^{0+}(k-1) < \mu_r^{0-}(k+1) < \mu_r^{0+}(k),$$

which entails regularity of affine additive processes. On the other hand, for every non-additive affine mixed process P_r^λ it is possible to obtain, for certain values of n and k , $0 \leq \mu_r^{\lambda-}(k+1) < \mu_r^{\lambda+}(k-1) \leq 1$. If $\mu(e)$ is given a fixed value between $\mu_r^{\lambda-}(k+1)$ and $\mu_r^{\lambda+}(k-1)$, and that the other components $\mu(e')$ ($e' \neq e$) take all possible values, then $\nu(e)$ will take at least the values $k-1$, k and $k+1$, and P_r^λ is not regular. $\square$

Proof of Theorem 6. Existence of a sequential algorithm for multiplicative processes (§ 2) implies that every multiplicative process is strongly temporally increasing. Let us assume that a mixed process P violates this property at a given point μ when time increases from n to $n+1$. There are at least two elements of $\mathbf{E}$, e and e' , say, whose shares increase and one element, e'' , say, whose share diminishes. Then it is easy to check that $\mu(e'')$ is lesser than both $\mu(e)$ and $\mu(e')$, which implies $P(\mu, n) \prec P(\mu, n+1)$ for the order defined into $\mathbf{E}$ by $e \leq e' \Leftrightarrow \mu(e) \leq \mu(e')$. Since the same relation $P(\mu, n) \prec P(\mu, n+1)$ obviously holds at every point μ where strong temporal increasingness is not violated, P is weakly temporally increasing. $\square$

Proof of Theorem 7. Figure 10 illustrates the fact that a mixed process is additively (respectively, multiplicatively) spatially increasing if and only if the equations of separating hyperplanes assume the form $\mu(e) - \mu(e') = \text{constant}$ (respectively, $\mu(e)/\mu(e') = \text{constant}$), that is if and only if P is an additive (respectively, multiplicative) process (Fig. 5). $\square$

Remark. We have made extensive use in some proofs of a geometric point of view. It could be replaced by a more algebraic view. Here is, for instance, an algebraic proof of multiplicative spatial increasingness of a multiplicative process (other proofs would be quite similar): If P_s^1 is not multiplicatively spatially increasing, there exist $e, e' \in \mathbb{E}$ such that:

$$\begin{aligned} \mu'(e)/\mu(e) &< \mu'(e')/\mu(e') \quad \text{and} \\ \nu(e) &< \nu'(e) \text{ and } \nu'(e') < \nu(e'), \quad \text{where we put:} \\ \nu &= P_s^1(\mu) \text{ and } \nu' = P_s^1(\mu'). \end{aligned}$$

It follows, using P_s^1 definition:

$$s(\nu(e) + 1)/s(\nu(e')) > \mu(e)/\mu(e').$$

Hence since s increases:

$$s(\nu'(e))/s(\nu'(e') + 1) > \mu'(e)/\mu'(e'),$$

which contradicts the definition of P_s^1 .

Proof of Theorem 8. Since $\mathbf{R}$ is totally ordered and $r \neq r'$ implies $P_r^\lambda \neq P_{r'}^\lambda$, it is sufficient to prove that $r < r'$ implies $P_r^\lambda \not\prec P_{r'}^\lambda$.

Let us assume that $P_r^\lambda \not\prec P_{r'}^\lambda$ does not hold: then there exists $(\mu, n) \in \Sigma_1^+ \times \mathbf{N}$ such that $\nu(e) = P_r^\lambda(\mu, n) \prec P_{r'}^\lambda(\mu, n) = \nu'(e)$ does not hold for the order defined over $\mathbb{E}$ by $e_1 \leq e_2 \leq \dots \leq e_p \Leftrightarrow \mu(e_1) \geq \mu(e_2) \geq \dots \geq \mu(e_p)$.

It ensues that there exist $e, e' \in \mathbb{E}$ such that $\mu(e) < \mu(e')$, $\nu(e) < \nu'(e)$ and $\nu(e') > \nu'(e')$. Then we should have at point μ : $I_r^\lambda(\nu(e), \nu(e')) > 0$ and $I_{r'}^\lambda(\nu'(e), \nu'(e')) = -I_{r'}^\lambda(\nu'(e) - 1, \nu'(e') + 1) > 0$.

But $I_r^\lambda(k, k')$ is, for any fixed λ, μ , and r , an increasing function of k and a decreasing function of k' . Thus $I_r^\lambda(\nu(e), \nu(e')) \leq I_r^\lambda(\nu'(e) - 1, \nu'(e') + 1) < 0 < I_r^\lambda(\nu(e), \nu(e'))$.

But $I_{r'}^\lambda(k, k')$ is, for any fixed λ, μ, k and k' with $k < k'$, an increasing function of r .

Since P_r^λ is μ -increasing and $\mu(e) < \mu(e')$, $\nu(e) < \nu(e')$ holds.

Thus $r' \leq r$. $\square$

Proof of Theorem 9. Let us first consider the case $r \geq 0$ and let us assume that $P_r^1 \not\prec P_r^0$ be false. Then for some e, e' in $\mathbb{E}$ and at some point (μ, n) in space-time we should have:

$$\mu(e') < \mu(e), \quad \nu^0(e) > \nu^1(e) \quad \text{and} \quad \nu^0(e') < \nu^1(e'),$$

where

$$\nu^0(e) = P_r^0(\mu, n) \quad \text{and} \quad \nu^1(e) = P_r^1(\mu, n).$$

At point μ , the following inequality (written $I_r^0(\nu^0, e, e') > 0$ in Theorem 3) holds:

$$\nu^0(e') + 1 + r - (n + r)\mu(e') > \nu^0(e) + r - (n + r)\mu(e).$$

Besides, we know (Corollary 1) that:

$$\mu(e') < [p\nu^0(e') + p + r - 1]/p(n + r).$$

Since $r \geq 0 > -1/(p-1)$, we obtain:

$$[p\nu^0(e') + p + r - 1]/p(n+r) < [\nu^0(e') + 1 + r]/(n+r) \text{ and thus } \\ \nu^0(e') + 1 + r - (n+r)\mu(e') > 0.$$

If we multiply this quantity by $1/\mu(e')$ which is greater than $1/\mu(e)$ (which is itself positive) the result is greater than $1/\mu(e)$ times $\nu^0(e) + r - (n+r)\mu(e)$, that is to say:

$$[\nu^0(e') + 1 + r]/\mu(e') > [\nu^0(e) + r]/\mu(e).$$

This latter inequality contradicts the following (written $I_r^1(\nu^1, e, e') > 0$ in Theorem 3):

$$[\nu^1(e) + 1 + r]\mu(e') > [\nu^1(e') + r]\mu(e).$$

As a matter of fact $\nu^1(e) + 1 \leq \nu^0(e)$ and $\nu^1(e') \geq \nu^0(e') + 1$ imply:

$$[\nu^0(e) + r]\mu(e') > [\nu^0(e') + 1 + r]\mu(e).$$

Let us now consider the case $r \leq -1$ and let us assume that $P_r^0 \setminus P_r^1$ is false. Then for some e, e' in E and at some point (μ, n) in space-time we should have:

$$\mu(e) < \mu(e'), \quad \nu^0(e) > \nu^1(e) \quad \text{and} \quad \nu^0(e') < \nu^1(e'),$$

where

$$\nu^0(e) = P_r^0(\mu, n) \quad \text{and} \quad \nu^1(e) = P_r^1(\mu, n).$$

At point μ , the following inequality (written $I_r^0(\nu^0, e', e) > 0$ in Theorem 3) holds:

$$(n+r)\mu(e) - [\nu^0(e) + r] > (n+r)\mu(e') - [\nu^0(e') + 1 + r].$$

Besides, we know (Corollary 1) that:

$$\mu(e) > [p\nu^0(e) - p + r + 1]/p(n+r).$$

Since $r \leq -1$ we obtain:

$$[p\nu^0(e) - p + r + 1]/p(n+r) > [\nu^0(e) + r]/(n+r)$$

and thus

$$(n+r)\mu(e) - [\nu^0(e) + r] > 0.$$

If we multiply this quantity by $1/\mu(e)$, which is greater than $1/\mu(e')$ (which is itself positive), the result is greater than $1/\mu(e')$ times $(n+r)\mu(e') - [\nu^0(e') + 1 + r]$, that is to say:

$$[\nu^0(e') + 1 + r]/\mu(e') > [\nu^0(e) + r]/\mu(e).$$

This latter inequality contradicts the following (written $I_r^1(\nu^1, e, e') > 0$ in Theorem 3):

$$[\nu^1(e) + 1 + r]\mu(e') > [\nu^1(e') + r]\mu(e).$$

As a matter of fact $\nu^1(e) + 1 \leq \nu^0(e)$ and $\nu^1(e') \geq \nu^0(e') + 1$ imply:

$$[\nu^0(e) + r]\mu(e') > [\nu^0(e') + 1 + r]\mu(e).$$

□

REFERENCES

- [1] N. AHUJA AND B. J. SCHACHTER, *Pattern Models*, John Wiley, New York, 1977.
- [2] M. L. BALINSKI AND H. P. YOUNG, *Fair Representation, Meeting the Ideal of One Man One Vote*, Yale University Press, New Haven, CT, 1982.
- [3] G. BIRKHOFF, *House monotone apportionment scheme*, Proc. Nat. Acad. Sci. U.S.A., 73 (1976), pp. 1–4.
- [4] H. BRENLY, *Quelques propriétés générales des systèmes sériels de représentation proportionnelle*, Math. et Sci. Hum., 94 (1986), pp. 33–44.
- [5] O. R. BURT AND C. C. HARRIS, *Apportionment of the US House of Representatives: a minimum range, integer solution, allocation problem*, Oper. Res., 11 (1963), pp. 648–652.
- [6] G. TH. GUILBAUD, *Leçons d'à peu près*, Christian Bourgois, Paris, 1985.
- [7] V. D'HONDT, *Système pratique et raisonné de représentation proportionnelle*, Murquardt, Bruxelles, 1882.
- [8] E. V. HUNTINGTON, *The apportionment of representatives in Congress*, Trans. Amer. Math. Soc., 30 (1928), pp. 85–110.
- [9] A. LACHENY, J. L. PETIT, AND E. TEROUANNE, *Vous avez dit proportionnelle?*, Math. et Sci. Hum., 94 (1986), pp. 5–32.
- [10] ———, *Une étude des méthodes proportionnelles sur l'exemple des élections législatives françaises de 1986*, Math. Inf. et Sci. Hum., 102 (1988), pp. 35–57.
- [11] J. P. MAYBERRY, *Quota methods for congressional apportionment are still non-unique*, Proc. Nat. Acad. Sci. USA, 75 (1978), pp. 3537–3539.
- [12] J. L. PETIT AND E. TEROUANNE, *A stormy assembly: electoral paradoxes*, Theory and Decision, 22 (1987), pp. 271–284.
- [13] ———, *Proportional methods, extremal values and manipulability*, Eur. J. Pol. Res., 16 (1988), pp. 339–356.
- [14] G. SAARI, *Apportionment methods and the House of Representatives*, Amer. Math. Monthly, 85 (1978), pp. 792–802.
- [15] A. SAINTE-LAGÜE, *La représentation proportionnelle et la méthode des moindres carrés*, C.R.A.S., 151 (1910), pp. 377–378.
- [16] J. W. STILL, *A class of new methods for congressional apportionment*, SIAM J. Appl. Math., 37 (1979), pp. 401–418.

BIT SERIAL MULTIPLICATION IN FINITE FIELDS*

MUZHONG WANG† AND IAN F. BLAKE‡

Abstract. Bit serial multiplication schemes for hardware implementation of arithmetic in a finite field of characteristic two are considered. In addition, certain aspects of polynomial bases in finite fields are investigated.

Key words. finite field arithmetic, digital circuitry

AMS(MOS) subject classification. 12C15

1. Introduction. In recent years the implementation of error correction and cryptographic systems using finite field arithmetic has become more prevalent. The realization of arithmetic operations in such an algebraic structure has received attention and several interesting approaches have been developed. For example, using a normal basis of F_{2^m} over F_2 results in simple addition and exponentiation, and the hardware realization of multiplication can be simplified by choosing the correct basis [6]. The work reported here extends the approach of Berlekamp [1] and Morii and Kasahara [5] to an efficient realization of multiplication, using a polynomial basis, that may be useful for hardware implementation.

2. Bit serial multiplication. We consider only multiplication of elements in F_{2^m} . Let $\alpha \in F_{2^m}$ be such that $\{1, \alpha, \alpha^2, \dots, \alpha^{m-1}\}$ is a basis of F_{2^m} over F_2 . Let $\{\beta_0, \beta_1, \dots, \beta_{m-1}\}$ be the dual basis so that

$$\text{Tr}(\alpha^i \beta_j) = \delta_{ij}, \quad i, j = 0, 1, 2, \dots, m-1$$

where $\text{Tr}(\cdot)$ is the trace function from F_{2^m} to F_2 and δ_{ij} is the Kronecker delta function, equal to 1 if i is equal to j and zero otherwise. We use a prime to denote a coordinate with respect to the dual basis so that for $x \in F_{2^m}$

$$x = \sum_{i=0}^{m-1} x_i \alpha^i = \sum_{i=0}^{m-1} x'_i \beta_i.$$

Notice that $x'_j = \text{Tr}(\alpha^j x)$. Throughout, we let the minimal polynomial of α be $f(z) = \sum_{i=0}^m f_i z^i$, a monic polynomial of degree m . It follows that the j th coordinate of αx in the dual basis is given by

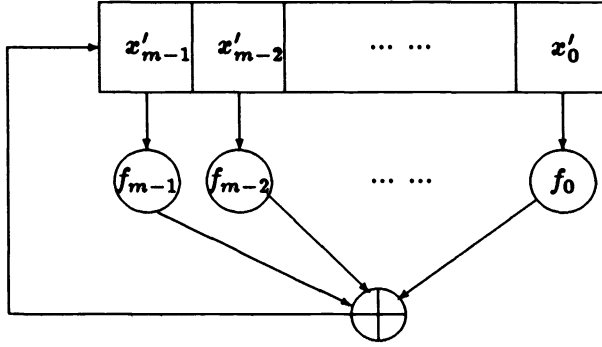
$$(\alpha x)'_j = \text{Tr}(\alpha^j \cdot \alpha x) = x'_{j+1}, \quad j = 0, 1, 2, \dots, m-2$$

and

$$\begin{aligned} (\alpha x)'_{m-1} &= \text{Tr}(\alpha^{m-1} \cdot \alpha x) \\ &= \sum_{i=0}^{m-1} f_i \text{Tr}(\alpha^i x) \\ &= \sum_{i=0}^{m-1} f_i x'_i. \end{aligned}$$

* Received by the editors June 13, 1988; accepted for publication May 11, 1989.

†‡ Department of Electrical Engineering, University of Waterloo, Waterloo, Canada N2L 3G1. This work was supported in part by Natural Sciences and Engineering Research Council of Canada grant 7382.


 FIG. 1. Multiplication by α .

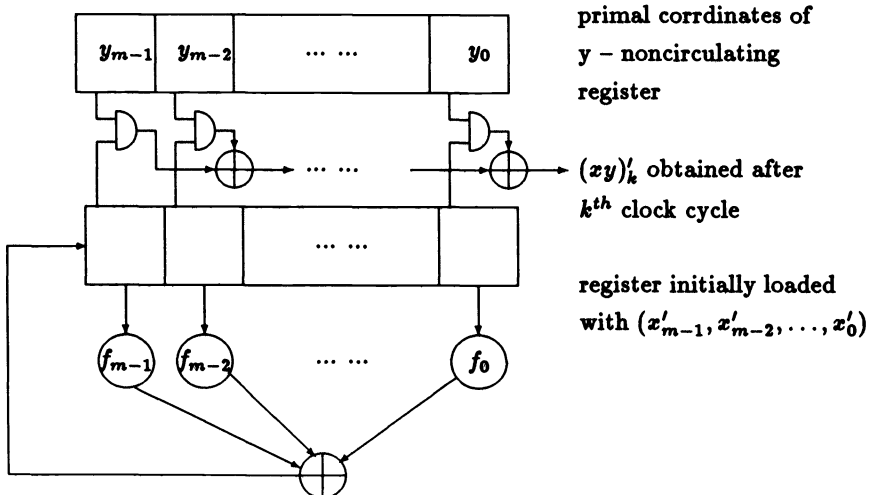
Thus multiplication of x by α can be achieved by the shift register shown in Fig. 1. If the shift register is initially loaded with $(x'_{m-1}, x'_{m-2}, \dots, x'_0)$, after k clock cycles it will contain the coordinates of $\alpha^k x$ with respect to the dual basis.

To multiply two elements $x, y \in F_{2^m}$, $y = \sum_{i=0}^{m-1} y_i \alpha^i$, consider

$$\begin{aligned} (xy)'_k &= \text{Tr}(\alpha^k xy) = \text{Tr}\left(\alpha^k x \sum_{i=0}^{m-1} y_i \alpha^i\right) = \sum_{i=0}^{m-1} y_i \text{Tr}(\alpha^{i+k} x) \\ &= \sum_{i=0}^{m-1} y_i (\alpha^k x)'_i \end{aligned}$$

which can be realized by the shift register arrangement of Fig. 2. This is the bit serial multiplication (BSM) scheme of Berlekamp [1], as described in [4]. Note that the result is obtained with respect to the dual coordinate system.

In the interesting paper of Morii and Kasahara [5], it is observed that it is not necessary to use the dual basis to achieve efficient multiplication. They proceeded as follows. For a fixed nonzero element $\beta \in F_{2^m}$ and arbitrary element $x \in F_{2^m}$, define the


 FIG. 2. Bit serial multiplication of x and y .

quantities $\tilde{x}_i = \text{Tr}(\beta \alpha^i x)$, $i = 0, 1, \dots, 2m-2$ and denote $\tilde{\underline{x}} = (\tilde{x}_0, \tilde{x}_1, \dots, \tilde{x}_{m-1})^t$. Note that if $\underline{x} = (x_0, x_1, \dots, x_{m-1})^t$ where $x = \sum_0^{m-1} x_i \alpha^i$, then $\tilde{\underline{x}} = T\underline{x}$ where

$$(1) \quad T = (\varepsilon_{ij}), \quad \varepsilon_{ij} = \text{Tr}(\beta \alpha^{i+j}), \quad i, j = 0, 1, \dots, m-1.$$

Note that the tilde quantities play the role of the dual coordinates in the previous BSM scheme.

Consider the multiplication of $x = \sum_0^{m-1} x_i \alpha^i$ and $y = \sum_0^{m-1} y_i \alpha^i$ to give $w = \sum_0^{m-1} w_i \alpha^i$ with vectors $\underline{y}$, $\tilde{\underline{y}}$, $\underline{w}$, and $\tilde{\underline{w}}$ defined in a similar manner. Then

$$\begin{aligned} \text{Tr}(\beta \alpha^l w) &= \tilde{w}_l \quad (l = 0, 1, \dots, m-1) \\ &= \text{Tr}(\beta \alpha^l xy) \\ &= \sum_j y_j \text{Tr}(\beta \alpha^{l+j} x) \\ &= \sum_{j=0}^{m-1} y_j \tilde{x}_{l+j}, \quad l = 0, 1, \dots, m-1 \end{aligned}$$

resulting in the matrix equation

$$(2) \quad \begin{pmatrix} \tilde{x}_0 & \tilde{x}_1 & \cdots & \tilde{x}_{m-1} \\ \tilde{x}_1 & \tilde{x}_2 & \cdots & \tilde{x}_m \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{x}_{m-1} & \tilde{x}_m & \cdots & \tilde{x}_{2m-2} \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{m-1} \end{pmatrix} = \begin{pmatrix} \tilde{w}_0 \\ \tilde{w}_1 \\ \vdots \\ \tilde{w}_{m-1} \end{pmatrix}.$$

This equation can be realized by the shift register of Fig. 3. In this form the relationship with the bit serial multiplication scheme of Fig. 2 is clearly seen. In the previous scheme the lower register is initially loaded with dual coordinates $\underline{x}' = (x'_{m-1}, x'_{m-2}, \dots, x'_0)$ where

$$\underline{x}' = T' \underline{x} \quad \text{and} \quad T' = (\varepsilon'_{ij}), \quad \varepsilon'_{ij} = \text{Tr}(\alpha^{i+j}), \quad i, j = 0, 1, \dots, m-1.$$

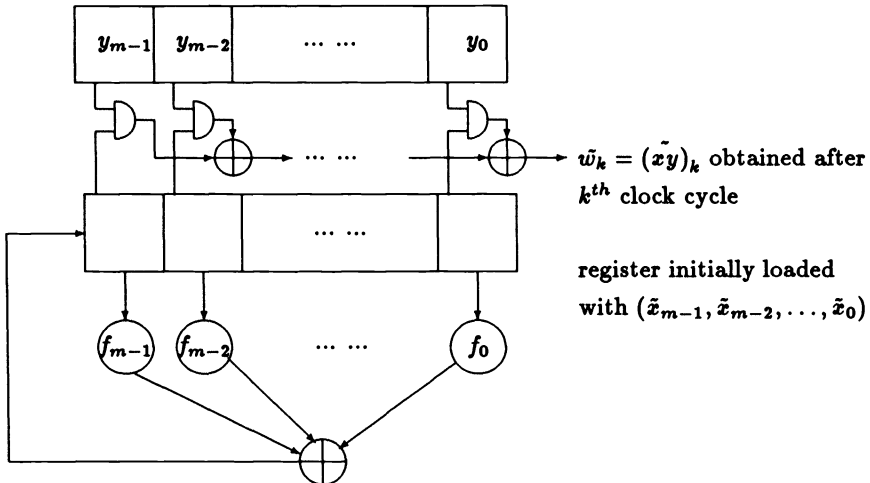


FIG. 3. Generalized bit serial multiplication of x and y .

In the scheme of Fig. 3 the register is initially loaded with $\underline{\tilde{x}} = T\underline{x}$ where T is given in (1). The point of [5] and this note is to choose β in such a manner that the matrix T is "simple."

An interesting example of this is given in [5] where it is shown that when the minimal polynomial of the primitive element α is a trinomial then β can be chosen so that T is a back circulant permutation matrix (one "1" per row and column realized as a back circulant matrix) and $T^{-1} = T$. Thus $\underline{x}$ is obtained from $\underline{\tilde{x}}$ by a cyclic shift of a certain number of positions and is particularly simple to implement. It is not necessary, however, that α be primitive as stated in [5]. To show this, it is sufficient to observe that if $\{1, \alpha, \alpha^2, \dots, \alpha^{m-1}\}$ is a basis of F_{2^m} over F_2 then the matrix

$$\begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha & \alpha^2 & \cdots & \alpha^{2^{m-1}} \\ \alpha^2 & (\alpha^2)^2 & \cdots & (\alpha^2)^{2^{m-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{m-1} & (\alpha^{m-1})^2 & \cdots & (\alpha^{m-1})^{2^{m-1}} \end{pmatrix}$$

is nonsingular (e.g., [3, p. 119]) and hence there exists a unique β such that

$$t_{0i} = \text{Tr}(\beta\alpha^i) \quad i = 0, 1, \dots, m-1, \quad t_{0i} \in F_2$$

for any given $(t_{00}, t_{01}, \dots, t_{0,m-1})$. If the minimal polynomial of α is $x^m + x^k + 1$, then choosing the first row of T matrix so that

$$t_{0,k-1} = 1 \quad t_{0,j} = 0 \quad \text{for } j \neq k-1$$

yields a permutation back circulant matrix, without the necessity of α being primitive as assumed in [5]. It is now clear that the coordinates of the product $w = xy$ with respect to the primal basis $\{1, \alpha, \dots, \alpha^{m-1}\}$ can be recovered from the output of the circuit of Fig. 3 by a simple permutation.

The relationship between bases and permutation matrices is developed a little further. It is clear that a basis $A = \{1, \alpha, \dots, \alpha^{m-1}\}$ cannot be self-dual. For a fixed nonzero element $\beta \in F_{2^m}$, $B = \{\beta, \beta\alpha, \dots, \beta\alpha^{m-1}\}$ is also a basis and we make the following definition.

DEFINITION. The basis $C = \{\gamma_1, \gamma_2, \dots, \gamma_m\}$ is said to be *permutation dual* to the basis $D = \{\eta_1, \eta_2, \dots, \eta_m\}$ if the matrix $(\text{Tr}(\gamma_i\eta_j))$ is a permutation matrix.

If D is permutation dual to C , then permuting the elements of D yields the actual dual of C and so the concept is only interesting when there is a natural ordering on both C and D .

LEMMA 1. *There exists an element $\beta \in F_{2^m}$ such that B is permutation dual to A if and only if the minimal polynomial of α is a trinomial.*

Proof. Suppose the minimal polynomial of α is the trinomial $x^m + x^k + 1$, $1 \leq k \leq m-1$ (it must be of degree m since A is assumed to be a basis). As above, choose $\beta \in F_{2^m}$ so that $\text{Tr}(\beta\alpha^{k-1}) = 1$, $\text{Tr}(\beta\alpha^i) = 0$, $0 \leq i \leq m-1$. It follows immediately that the matrix T of (1) is a back circulant matrix and hence B is permutation dual to A . Notice that in this case the basis

$$B' = \{\beta\alpha^{k-1}, \beta\alpha^{k-2}, \dots, \beta\alpha, \beta, \beta\alpha^{m-1}, \dots, \beta\alpha^m\},$$

an ordered set, is the dual basis to A .

Now suppose that B is permutation dual to A . Then for some k , $1 \leq k \leq m-1$, $\text{Tr}(\beta\alpha^{k-1}) = 1$ and so $\text{Tr}(\beta\alpha^i) = 0$, $0 \leq i \leq m-1$, $i \neq k-1$. It follows that the matrix T is a back circulant matrix. For the entry on the k th row, $\text{Tr}(\alpha^k\beta\alpha^i)$, note that i must be $m-1$ for a nonzero entry since, if i is less than $m-1$, the matrix T would have at

least one column with more than one nonzero entries, contradicting the assumption that T is a permutation matrix, i.e., that B is permutation dual to A . Now suppose the minimal polynomial of α is $f(x) = \sum_{i=0}^m f_i x^i$ and consider the sequence generated by the associated shift register with initial state corresponding to the first row of T . At successive clock cycles, the rows of T are generated. This is only possible if $f_0 = f_k = 1, f_i = 0, i \neq 0, k$, i.e., if the minimal polynomial of α is the trinomial $x^m + x^k + 1$, as was required to show.

The lemma illustrates, to some extent, the approaches of [1] and [5]. It is shown in [5] that even when such an irreducible trinomial does not exist, it is often possible to choose a $\beta \in F_{2^m}$ so that the matrix T has a fairly simple form, compared to that of the BSM scheme, but the approach is ad hoc. Such schemes are referred to as generalized bit serial multiplication (GBSM) schemes in [5]. The purpose of this note is to give an alternate form for the matrix T which always works and results in an efficient multiplication scheme. Although not as attractive perhaps as when T is permutation matrix, it has the advantage of always being applicable and possessing regular structure for hardware implementation. For the sake of differentiation from BSM and GBSM, we refer to this scheme as modified bit serial multiplication (MBSM).

3. Modified bit serial multiplication. Consider the shift register of Fig. 4 (Fig. 1 revised), initially loaded with $\underline{x} = (0, 0, \dots, 0, 1)$ and let the sequence so generated be $\{b_n\}$ so that $b_0 = \dots = 0, b_{m-1} = 1, b_m = f_{m-1}, b_{m+1} = f_{m-2} + f_{m-1}b_m, \dots, b_{i+1} = \sum_{j=m-1} b_j f_{j-i+(m-1)}, i = m-1, m, \dots, 2m-2$. Let the i th row of the matrix T be the contents of the shift register after the i th clock cycle, $i = 0, 1, \dots, m-1$ so that

$$(3) \quad T = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 1 & b_m \\ 0 & 0 & 0 & \cdots & 1 & b_m & b_{m+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & b_m & \cdots & b_{2m-5} & b_{2m-4} & b_{2m-3} \\ 1 & b_m & b_{m+1} & \cdots & b_{2m-4} & b_{2m-3} & b_{2m-2} \end{pmatrix}.$$

Notice that for $f(z)$ an irreducible polynomial there always exists a $\beta \in F_{2^m}$ so that this matrix T is the matrix of (1). An explicit formula for this β can be obtained by choosing $\beta \in F_{2^m}$ so that $(\text{Tr}(\beta), \text{Tr}(\beta\alpha), \dots, \text{Tr}(\beta\alpha^{m-1})) = (0, 0, \dots, 0, 1)$, which

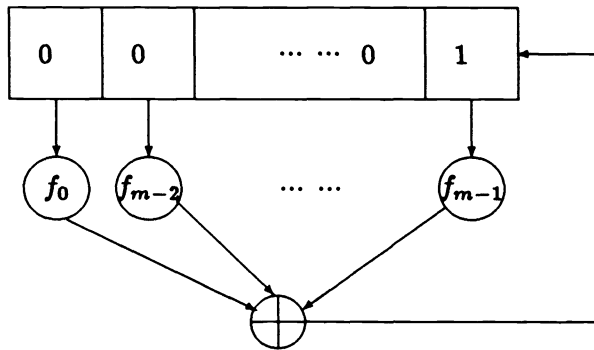


FIG. 4. Generation of $\{b_n\}$.

is equivalent to solving the matrix equation

$$\begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha & \alpha^2 & \cdots & \alpha^{2^{m-1}} \\ \alpha^2 & (\alpha^2)^2 & \cdots & (\alpha^2)^{2^{m-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{m-1} & (\alpha^{m-1})^2 & \cdots & (\alpha^{m-1})^{2^{m-1}} \end{pmatrix} \begin{pmatrix} \beta \\ \beta^2 \\ \vdots \\ \beta^{2^{m-1}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}.$$

The matrix is of the van der Monde type and can be shown to have determinant 1 [3, p. 117]. The required solution is

$$\begin{aligned} \beta &= \begin{vmatrix} 0 & 1 & \cdots & 1 \\ 0 & \alpha^2 & \cdots & \alpha^{2^{m-1}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & (\alpha^{m-1})^2 & \cdots & (\alpha^{m-1})^{2^{m-1}} \end{vmatrix} \div \begin{vmatrix} 1 & 1 & \cdots & 1 \\ \alpha & \alpha^2 & \cdots & \alpha^{2^{m-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{m-1} & (\alpha^{m-1})^2 & \cdots & (\alpha^{m-1})^{2^{m-1}} \end{vmatrix} \\ &= \prod_{\substack{i,j=1 \\ i < j}}^{m-1} (\alpha^{2^i} + \alpha^{2^j}) \bigg/ \prod_{\substack{i,j=0 \\ i < j}}^{m-1} (\alpha^{2^i} + \alpha^{2^j}) \\ &= \left\{ \prod_{j=1}^{m-1} (\alpha + \alpha^{2^j}) \right\}^{-1} \\ &= f'(\alpha)^{-1} \end{aligned}$$

where the well-known formula for the determinant of van der Monde matrix has been used.

LEMMA 2. *Over the ring of integers, and hence over F_2 ,*

$$(4) \quad T^{-1} = \begin{pmatrix} -f_1 & -f_2 & \cdots & -f_{m-2} & -f_{m-1} & 1 \\ -f_2 & -f_3 & \cdots & -f_{m-1} & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ -f_{m-1} & 1 & \cdots & 0 & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix}.$$

Proof. For convenience let $A = T = (a_{ik})$ and $B = T^{-1} = (b_{ik})$ $i, k = 1, 2, \dots, m$. Thus

$$a_{ik} = b_{k+i-2}, \quad k = m+1-i, m+2-i, \dots, m$$

$$b_{jk} = -f_{i+k-1}, \quad k = 1, 2, \dots, m-i$$

for $i = 1, 2, \dots, m$ where f_m is defined as -1 and $b_{m-1} = 1$. Proof is by straightforward verification that $AB = I$. Consider the (i, j) element of AB

$$\begin{aligned} (AB)_{ij} &= \sum_{k=1}^m a_{ik} b_{kj} = \begin{cases} -\sum_{k=m+1-i}^{m+1-j} b_{k+i-2} f_{k+j-1} & \text{for } i \geq j \\ 0 & \text{for } j > i \end{cases} \\ (AB)_{ij} &= -\{b_{m-1} f_{m+j-i} + b_m f_{m+j-i+1} + \cdots + b_{m+i-j-1} f_m\}, \quad i \geq j \\ &= b_{m-1+i-j} - \{b_{m-1} f_{m+j-i} + \cdots + b_{m-2+i-j} f_{m-1}\}, \quad i \geq j \\ &= \begin{cases} 0 & \text{if } i > j; \\ 1 & \text{if } i = j. \end{cases} \end{aligned}$$

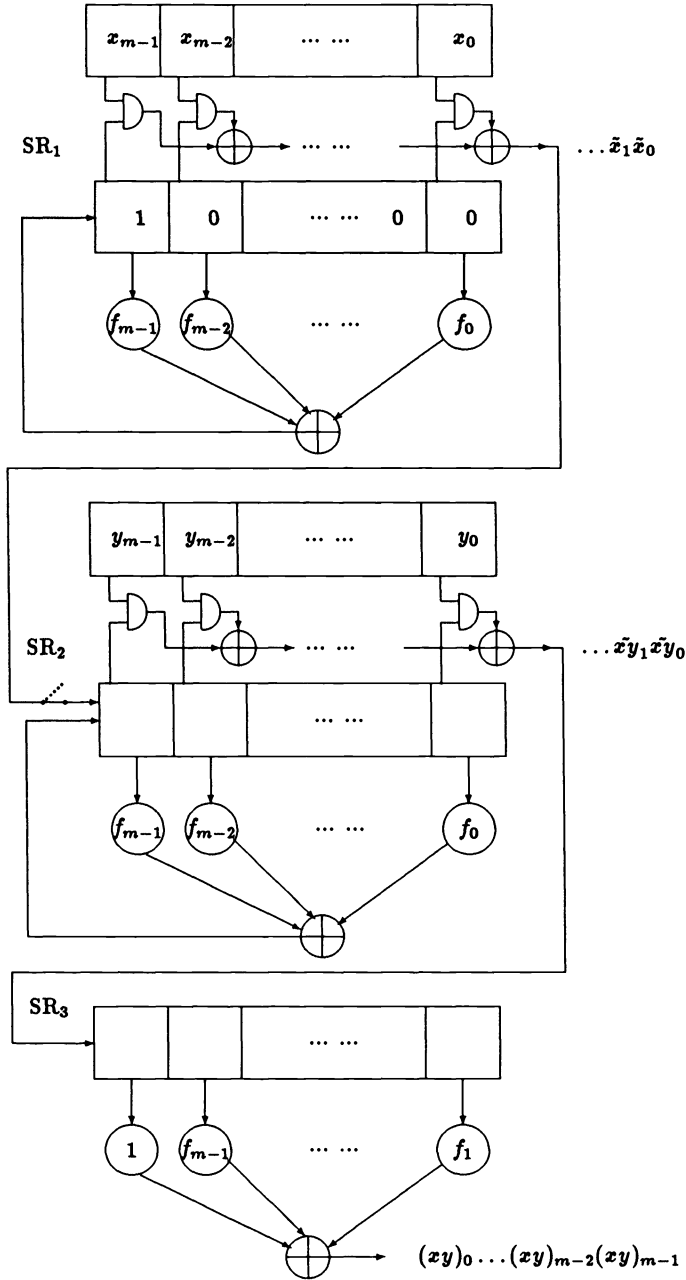


FIG. 5. Modified bit serial multiplication of x and y .

Using this matrix T , and its inverse, and the notations of the previous section, the multiplication of x and y can be realized conceptually by the shift register circuits of Fig. 5. After m clock cycles the uppermost circuit SR_1 has produced and loaded $\tilde{x}$ into SR_2 at which point the switch is opened. After a further m clock cycles the coordinates of xy , with respect to the primal basis, appear serially in reverse order.

4. On the dual of a polynomial basis. It is well known [2] that the dual of a polynomial basis $\{1, \alpha, \dots, \alpha^{m-1}\}$, where the minimal polynomial of $\alpha \in F_{2^m}$ over F_2 is $f(x)$, is given by

$$(5) \quad \left\{ \frac{g_0}{f'(\alpha)}, \frac{g_1}{f'(\alpha)}, \dots, \frac{g_{m-1}}{f'(\alpha)} \right\}$$

where

$$f(x) = (x + \alpha) \left(\sum_{i=0}^{m-1} g_i x^i \right), \quad g_i \in F_{2^m}, \quad g_{m-1} = 1.$$

This result also follows readily from the results of the previous section, as shown below.

As in (1) let

$$\begin{aligned} T &= \text{Tr} \{ (1, \alpha, \dots, \alpha^{m-1})^t (\beta, \beta\alpha, \dots, \beta\alpha^{m-1}) \} \\ &= (\text{Tr} (\beta\alpha^{i+j})). \end{aligned}$$

For $\beta \neq 0$, $\beta \in F_{2^m}$, T is nonsingular and so

$$I = \text{Tr} \{ (1, \alpha, \dots, \alpha^{m-1})^t (\beta, \beta\alpha, \dots, \beta\alpha^{m-1}) T^{-1} \}$$

implying that the dual basis to $\{1, \alpha, \dots, \alpha^{m-1}\}$ is

$$(\beta, \beta\alpha, \dots, \beta\alpha^{m-1}) T^{-1}.$$

If β is chosen as $f'(\alpha)^{-1}$ then T is given by (3) and T^{-1} by (4), and so the dual basis is

$$(6) \quad f'(\alpha)^{-1} \left(\sum_{i=1}^m f_i \alpha^{i-1}, \sum_{i=2}^m f_i \alpha^{i-2}, \sum_{i=3}^m f_i \alpha^{i-3}, \dots, 1 \right).$$

Now, by definition

$$(7) \quad f(x) = \sum_{i=0}^m f_i x^i = (x + \alpha) \sum_{i=0}^{m-1} g_i \alpha^i, \quad g_{m-1} = 1.$$

Recursively identifying coefficients in (7) gives

$$g_{i-1} = \sum_{j=i}^m f_j \alpha^{j-i} \quad i = 1, 2, \dots, m$$

which identifies the basis (6) and (5).

5. Comments. A modification of the BSM of [1] and GBSM of [5] has been given and some properties of these schemes developed. It would appear that when the matrix T can be realized as a permutation, the resulting circuit is most efficient both in terms of gate count and cycle times. Such a permutation representation is not always available and in such cases the modified version presented in § 3 may be preferable to realizing the T , T^{-1} and multiplication matrix of (2) by arrays of AND gates. In some cases the choice of multiplication technique may depend on the computation of which it is part.

Some of the results shown here are of independent interest and many are true for the more general case of F_{q^m} over F_q . Because of the connection with hardware implementations, they were only shown here for the case of characteristic two.

REFERENCES

- [1] E. R. BERLEKAMP, *Bit-serial Reed-Solomon encoders*, IEEE Trans. Inform. Theory, 28 (1982), pp. 869–874.
- [2] R. LIDL AND H. NIEDERREITER, *Finite Fields*, Addison-Wesley Publishing Co., Reading, MA, 1983.
- [3] F. S. MACWILLIAMS AND N. J. A. SLOANE, *The Theory of Error Correcting Codes*, North Holland, New York, 1977.
- [4] R. J. McELIECE, *Finite Fields for Computer Scientists and Engineers*, Kluwer Academic Publishers, Boston, 1987.
- [5] M. MORII AND M. KASAHARA, *Efficient bit-serial multiplier*, IEEE Trans. Inform. Theory, (1987), submitted.
- [6] R. C. MULLIN, I. M. ONYSZCHUK, AND S. A. VANSTONE, *Optimal normal bases in $GF(p^n)$* , Discrete Appl. Math., 22 (1988/89), pp. 149–161.

CALCUL SAALSCHÜTZIEN DES PARTITIONS ET DES DÉRANGEMENTS COLORÉS*

JIANG ZENG†

Résumé. Deux formules de nature combinatoire déjà établies par des méthodes analytiques sont démontrées combinatoirement. Elles fournissent le calcul des coefficients de linéarisation de produits de polynômes orthogonaux (Hermite, Laguerre, Charlier, Krawtchouk, Meixner). D'autre part, l'une d'elles se spécialise en les formules de Pfaff–Saalschütz et de Dixon.

Abstract. Two formulas of combinatorial nature already derived by means of analytical methods are proved combinatorially. They provide the calculation of the linearization coefficients of products of orthogonal polynomials (Hermite, Laguerre, Charlier, Krawtchouk, Meixner). On the other hand, one of them specializes into the two formulas of Pfaff–Saalschütz and Dixon.

Key words. Pfaff–Saalschütz identity, colored derangements, linearization coefficients, exceedances, cycles, colored partitions, compound alphabets

AMS(MOS) subject classifications. 05A15, 05A17, 33A15

1. Introduction. Notons $(a)_n$ les factorielles montantes

$$(a)_0 = 1 \quad \text{et} \quad (a)_n = a(a+1) \cdots (a+n-1) \quad (n \geq 1)$$

et

$${}_3F_2 \left(\begin{matrix} a_1, a_2, a_3 \\ b_1, b_2 \end{matrix}; x \right) = \sum_{n \geq 0} \frac{(a_1)_n (a_2)_n (a_3)_n}{(b_1)_n (b_2)_n} \frac{x^n}{n!}$$

la série hypergéométrique avec trois paramètres au numérateur et deux paramètres au dénominateur. L'identité suivante, dans laquelle n est un entier positif

$$(1) \quad {}_3F_2 \left(\begin{matrix} a, b, -n \\ c, 1+a+b-c-n \end{matrix}; 1 \right) = \frac{(c-a)_n (c-b)_n}{(c)_n (c-a-b)_n},$$

est traditionnellement appelée formule de Pfaff–Saalschütz (cf. [As], [Bai]). Elle apparaît dans la littérature sous différentes formes équivalentes et en particulier comme (cf. [Ca-Fo], [Ge-St], [Go])

(2)

$$\binom{n_1}{k-n_3} \binom{n_2}{k-n_1} \binom{n_3}{k-n_2} = \sum_{s \geq 0} \frac{s!}{(s-n_1)! (s-n_2)! (s-n_3)! (k-s)! (n_1+n_2+n_3-k-s)!},$$

qui par sommation sur k et l'introduction d'une nouvelle variable γ donne [Ma], [Go]:

$$(3) \quad \sum_{k \geq 0} \binom{n_1}{k-n_3} \binom{n_2}{k-n_1} \binom{n_3}{k-n_2} \gamma^k = \sum_{s \geq 0} \frac{s! (1+\gamma)^{n_1+n_2+n_3-2s}}{(s-n_1)! (s-n_2)! (s-n_3)! (n_1+n_2+n_3-2s)!} \gamma^s.$$

* Received by the editors August 22, 1988; accepted for publication (in revised form) May 19, 1989. Ce travail a été accompli avec le soutien du P.R.C. mathématique et informatique.

† Département de Mathématique, Université Louis-Pasteur, 7 Rue René-Descartes, F-67084 Strasbourg Cedex, France. Current address, School of Mathematics, The Institute for Advanced Study, Princeton, New Jersey 08540.

On remarque que cette dernière formule est aussi équivalente à une transformation quadratique de Whipple (cf. [Er, p. 190, (2)] ou [Bai, p. 97, Ex. 4, (iv)]), comme cité dans [Go] et [Ge-St].

Le calcul des coefficients de linéarisation des polynômes de Meixner et de Krawtchouk [Ze1] exige l'évaluation du second membre de (3) lorsque $s!$ est remplacé par la factorielle montante $(\beta)_s$. Le premier membre devient alors un polynôme en les deux variables β et γ , qu'on note $(1/(n_1!n_2!n_3!))D(\mathbf{n}; \beta, \gamma)$ pour des raisons de normalisation. Dans [Ze1] on a montré par des méthodes relativement élaborées que $D(\mathbf{n}; \beta, \gamma)$ est la fonction génératrice de certaines permutations, appelées *dérangements colorés*, par un certain poids.

Pour définir ces dérangements colorés, on se donne une suite de trois entiers positifs $\mathbf{n} = (n_1, n_2, n_3)$, puis on forme l'ensemble

$$C_{\mathbf{n}} = \{(1, 1), \dots, (1, n_1), (2, n_1 + 1), \dots, (2, n_1 + n_2), \\ (3, n_1 + n_2 + 1), \dots, (3, n_1 + n_2 + n_3)\}.$$

Il est commode de dire que le premier terme i du couple $a = (i, j)$ est la *couleur* de a et d'écrire $c(a) = i$. Un *dérangement coloré* est alors défini comme étant une permutation π de $C_{\mathbf{n}}$ telle que $c(a) \neq c(\pi(a))$ pour tout $a \in C_{\mathbf{n}}$. On note ensuite $\text{cyc } \pi$ le *nombre de cycles* de π et on désigne par $\text{exc } \pi$ le nombre de ses *excédances*, c'est-à-dire le nombre des éléments a de $C_{\mathbf{n}}$ tels que $c(a) < c(\pi(a))$. Si $\mathcal{D}_{\mathbf{n}}$ désigne l'ensemble de tous les dérangements colorés de $C_{\mathbf{n}}$, on considère le polynôme générateur:

$$D(\mathbf{n}; \beta, \gamma) = \sum_{\pi \in \mathcal{D}_{\mathbf{n}}} \beta^{\text{cyc } \pi} \gamma^{\text{exc } \pi}.$$

Le but principal de cet article est de montrer par des techniques entièrement combinatoires le théorème suivant.

THÉORÈME. *On a*

$$(4) \quad D(\mathbf{n}; \beta, \gamma) = \sum_{s \geq 0} \frac{n_1!n_2!n_3!(1+\gamma)^{n_1+n_2+n_3-2s}}{(s-n_1)!(s-n_2)!(s-n_3)!(n_1+n_2+n_3-2s)!} \gamma^s (\beta)_s.$$

On remarque que lorsque $\beta = 1$, le calcul du premier membre de (4) devient évident, à savoir

$$(5) \quad D(\mathbf{n}; 1, \gamma) = n_1!n_2!n_3! \sum_{k \geq 0} \binom{n_1}{k-n_3} \binom{n_2}{k-n_1} \binom{n_3}{k-n_2} \gamma^k,$$

et on retrouve la formule (3). La formule (4) apparaît ainsi comme une β -extension de la formule de Pfaff-Saalschütz. Comme elle est obtenue, en fait, en évaluant une fonction génératrice de dérangements colorés, on dit qu'on a effectué le *calcul saalschützien* de ces structures.

On peut aussi évaluer, par des techniques plus sophistiquées, le polynôme $D(\mathbf{n}; \beta, \gamma)$ lorsque (n_1, n_2, n_3) est remplacé par une suite de $m \geq 1$ entiers positifs $\mathbf{n} = (n_1, \dots, n_m)$. En effet, la fonction génératrice de $D(\mathbf{n}; \beta, \gamma)$ ainsi obtenu admet une expression explicite à l'aide des fonctions symétriques élémentaires $e_2, \dots, e_m$ des variables $x_1, \dots, x_m$ (cf. [Ze1]):

$$(6) \quad \sum_{\mathbf{n}} D(\mathbf{n}; \beta, \gamma) \frac{x_1^{n_1}}{n_1!} \dots \frac{x_m^{n_m}}{n_m!} \\ = [1 - \gamma e_2 - \gamma(1+\gamma)e_3 - \dots - \gamma(1+\gamma + \dots + \gamma^{m-2})e_m]^{-\beta}$$

où la sommation est faite sur toutes les suites $\mathbf{n} = (n_1, \dots, n_m)$ de m entiers positifs. Il est facile de voir que le théorème ci-dessus est équivalent au cas $m = 3$ de (6). Malheureusement, la méthode présentée ici ne peut s'étendre à (6). Il est à noter que lorsque $\mathbf{n} = (1, \dots, 1)$ le polynôme $D(\mathbf{n}; \beta, \gamma)$ se réduit au polynôme générateur des *dérangements* de $\{1, 2, \dots, m\}$, noté $d(m; \beta, \gamma)$, et la fonction génératrice correspondant à (6) est:

$$(7) \quad \sum_{m \geq 0} d(m; \beta, \gamma) \frac{x^m}{m!} = \left[1 - \sum_{l \geq 2} (\gamma + \dots + \gamma^{l-1}) \frac{x^l}{l!} \right]^{-\beta}.$$

Récemment, Gessel [Ge2] a donné une démonstration directe de (7) dans le cas où $\beta = \gamma = 1$ en combinant sa *décomposition de crochet* et un résultat dû à Désarménien [De]. En fait, d'après les remarques faites dans l'addendum de [Ge2] on peut démontrer (7) directement lorsque $\gamma = 1$ et β *quelconque*. Cependant, les méthodes de Gessel ne semblent pas permettre d'établir l'identité (7) lorsque γ est *quelconque*, même dans le cas $\beta = 1$.

Enfin, l'étude des coefficients de linéarisation des polynômes de Charlier [Ze1] fait apparaître également une formule analogue à (4), qui repose cette fois non plus sur les dérangements, mais sur des partitions dites colorées (cf. Proposition 1). Dans la § 2, on démontre cette formule par un simple comptage. Les trois sections suivantes sont consacrées à l'évaluation de $D(\mathbf{n}; \beta, \gamma)$. La dernière section est constituée de remarques et d'exemples.

2. Partitions colorées. Considérons une partition π de C_n . Si chacun de ses blocs ne contient que des éléments de couleur différente, on dit que π est une *partition colorée*. On note $\text{bloc } \pi$ le nombre de blocs de π et $\mathcal{PCS}(\mathbf{n})$ l'ensemble des partitions colorées sans points isolés, c'est-à-dire sans blocs réduits à un élément. Soit

$$\text{PCS}(\mathbf{n}; a) = \sum_{\pi} a^{\text{bloc } \pi} \quad (\pi \in \mathcal{PCS}(\mathbf{n}))$$

la fonction génératrice de $\mathcal{PCS}(\mathbf{n})$ suivant le nombre de blocs.

PROPOSITION 1. On a

$$\text{PCS}(\mathbf{n}; a) = \sum_{s \geq 0} \frac{n_1! n_2! n_3! a^s}{(s - n_1)! (s - n_2)! (s - n_3)! (n_1 + n_2 + n_3 - 2s)!}.$$

Démonstration. On dit qu'une partie B de C_n est de couleur $c = [1^i 2^j 3^k]$, si B contient i, j et k éléments de couleur 1, 2 et 3, respectivement. Soit π une partition de C_n . On note $n_c \pi$ le nombre de blocs de π qui sont de couleur c . Si π est une partition colorée sans points isolés, ses blocs ne peuvent avoir que l'une des couleurs suivantes: [12], [13], [23], [123]. Si, de plus, elle a s blocs, les relations suivantes sont satisfaites:

$$\begin{aligned} n_{12}\pi + n_{13}\pi + n_{123}\pi &= n_1, & n_{12}\pi + n_{23}\pi + n_{123}\pi &= n_2, \\ n_{13}\pi + n_{23}\pi + n_{123}\pi &= n_3, & n_{12}\pi + n_{13}\pi + n_{23}\pi + n_{123}\pi &= s. \end{aligned}$$

D'où

$$(8) \quad n_{12}\pi = s - n_3, \quad n_{13}\pi = s - n_2, \quad n_{23}\pi = s - n_1, \quad n_{123}\pi = n_1 + n_2 + n_3 - 2s.$$

Il est facile de voir qu'il existe

$$(9) \quad \frac{n_1! n_2! n_3!}{(n_{12}\pi)! (n_{13}\pi)! (n_{23}\pi)! (n_{123}\pi)!}$$

partitions dans $\mathcal{PCS}(\mathbf{n})$ qui ont le même nombre de blocs de chaque couleur que π . La formule à démontrer résulte donc de (8) et (9). $\square$

3. Alphabets composés. Soit $w = (a_1, b_1)(a_2, b_2) \cdots (a_r, b_r)$ un mot dont les lettres sont des éléments de C_n . On dit que $[a_1 a_2 \cdots a_r]$ est la *couleur* de w . Le mot w est dit être un *segment*, si $r \geq 2$ et si a_1 est plus petit que chaque a_i pour $i \geq 2$. On appelle *lettre composée* tout mot ayant l'une des cinq formes suivantes: $(1, a)(2, b)$; $(1, a)(3, c)$; $(2, b)(3, c)$; $(1, a)(2, b)(3, c)$; $(1, a)(3, c)(2, b)$; où $1 \leq a \leq n_1$, $n_1 + 1 \leq b \leq n_1 + n_2$, $n_1 + n_2 + 1 \leq c \leq n_1 + n_2 + n_3 = n$. Ces cinq lettres composées sont, par définition, de couleur respective [12], [13], [23], [123], [132]. Soit $w = w_1 w_2 \cdots w_s$ le produit de juxtaposition des lettres composées $w_1, w_2, \dots, w_s$. Si ce produit est un réarrangement de $(1, 1)(1, 2) \cdots (3, n)$, on dit que l'ensemble $\{w_1, w_2, \dots, w_s\}$ est un *alphabet composé* de C_n .

Par exemple, l'ensemble

$$\Omega = \{(1, 2)(2, 5)(3, 10); (1, 1)(2, 8); (2, 4)(3, 9); (1, 3)(3, 12)(2, 7); (2, 6)(3, 11)\}$$

est un alphabet composé de C_n avec $\mathbf{n} = (3, 5, 4)$.

La proposition suivante se démontre de la même façon que la Proposition 1. On l'énoncera donc sans démonstration.

PROPOSITION 2. *Le nombre d'alphabets composés de C_n , de cardinal s , dont exactement k éléments sont de couleur [123], est égal à:*

$$\frac{n_1! n_2! n_3!}{(s - n_1)!(s - n_2)!(s - n_3)!(n_1 + n_2 + n_3 - 2s - k)!k!}.$$

4. Dérangements colorés. On se propose de construire une bijection qui associe à tout couple (Ω, σ) , où Ω est un alphabet composé de C_n et σ une permutation de Ω , une permutation π de C_n ayant certaines propriétés (cf. énoncé de la Proposition 3). Pour cela, on part d'un tel couple (Ω, σ) et l'on définit π en décrivant ses cycles un à un. On note a_1 le plus petit élément de C_n (par rapport à l'ordre lexicographique). Naturellement, $a_1 = (1, 1)$ si $n_1 \geq 1$. Soit w_1 la lettre composée de Ω , dont la lettre initiale est a_1 et soit $\tau = (w_1 w_2 \cdots w_t)$ le cycle de σ contenant w_1 . On définit $\delta(\tau)$ comme étant le produit de juxtaposition:

$$\delta(\tau) = w_1 w_2 \cdots w_t.$$

Par exemple, si $a_1 = (1, 1)$ et

$$\tau = (\{(1, 1)(2, 5)(3, 7)\} \{(1, 3)(2, 6)\} \{(2, 4)(3, 8)\} \{(1, 2)(3, 9)\}),$$

alors

$$\delta(\tau) = (1, 1)(2, 5)(3, 7)(1, 3)(2, 6)(2, 4)(3, 8)(1, 2)(3, 9).$$

Comme a_1 est la plus petite lettre dans $\delta(\tau)$, on peut, de plus, factoriser $\delta(\tau)$ en produit de segments dont les lettres initiales ont la même couleur que a_1 .

Dans l'exemple précédent, on a la factorisation:

$$\delta(\tau) = (1, 1)(2, 5)(3, 7)|(1, 3)(2, 6)(2, 4)(3, 8)|(1, 2)(3, 9).$$

Soit $\delta(\tau) = s_1 s_2 \cdots s_r$ la factorisation en segments s_i ($1 \leq i \leq r$) ainsi obtenue. Comme tous les w_i ($1 \leq i \leq t$) sont des lettres composées, chaque s_i ($1 \leq i \leq r$) a forcément l'une des couleurs suivantes:

$$[23], [12(23)^j], [13(23)^j], [123(23)^j], [132(23)^j] \quad (j \geq 0).$$

On réarrange maintenant chaque segment s_i en appliquant l'opérateur R défini comme suit:

- (i) si s_i est de l'une des couleurs $[23]$, $[13(23)^j]$, $[123(23)^j]$, alors $R(s_i) = s_i$;
- (ii) si s_i est de couleur $[12(23)^j]$ et $s_i = (1, x)(2, y)u$, avec u de couleur $[(23)^j]$, alors $R(s_i) = (1, x)u(2, y)$;
- (iii) si s_i est de couleur $[132(23)^j]$ et $s_i = (1, x)(3, y)(2, z)u$, avec u de couleur $[(23)^j]$, alors $R(s_i) = (1, x)(3, y)u(2, z)$.

On étend ensuite cet opérateur R au mot $\delta(\tau)$ en posant:

$$R(\delta(\tau)) = R(s_1)R(s_2) \cdots R(s_r).$$

Dans l'exemple précédent, on a:

$$\begin{aligned} R(\delta(\tau)) &= R((1, 1)(2, 5)(3, 7))R((1, 3)(2, 6)(2, 4)(3, 8))R((1, 2)(3, 9)) \\ &= (1, 1)(2, 5)(3, 7) \quad (1, 3)(2, 4)(3, 8)(2, 6) \quad (1, 2)(3, 9). \end{aligned}$$

Le cycle τ' de π contenant a_1 est simplement défini comme le cycle:

$$\tau' = (R(s_1)R(s_2) \cdots R(s_r)).$$

Dans l'exemple ci-dessus, on a:

$$\tau' = ((1, 1)(2, 5)(3, 7)(1, 3)(2, 4)(3, 8)(2, 6)(1, 2)(3, 9)).$$

On inaugure ensuite un nouveau cycle de π en partant du plus petit élément de l'ensemble C_n privé de tous les éléments apparaissant dans τ' . On construit un second cycle de π et ainsi de suite. On voit facilement que l'application $(\Omega, \sigma) \mapsto \pi$ est bijective.

Notons $|\Omega|$ (respectivement, $|\Omega_{123}|$) le nombre des lettres composées (respectivement, des lettres composées de couleur $[123]$) dans Ω .

PROPOSITION 3. *La permutation π définie par l'algorithme précédent est un dérangement de C_n et satisfait les propriétés:*

$$\text{cyc } \pi = \text{cyc } \sigma \quad \text{et} \quad \text{exc } \pi = |\Omega| + |\Omega_{123}|.$$

Démonstration. On remarque d'abord que chaque $R(s_i)$ ($1 \leq i \leq r$) est encore un segment, ayant la même lettre initiale que s_i . De plus, $R(s_i)$ est de l'une des couleurs suivantes:

$$[23], [1(23)^j], [13(23)^j], [123(23)^j], [13(23)^j2] \quad (j \geq 0).$$

Il en résulte que l'image par π de chaque élément de C_n est un élément de couleur différente; ce qui montre que π est un dérangement. D'autre part, chaque cycle de σ correspond exactement à un cycle de π . D'où $\text{cyc } \pi = \text{cyc } \sigma$.

De plus, pour évaluer le nombre d'excédances de π , on remarque que chaque excédance de π portée par le cycle τ' provient de deux lettres consécutives ab du mot $R(\delta(\tau))$ telles que $c(a) < c(b)$. Or d'après la définition de R , un tel couple (a, b) apparaît aussi dans le mot $\delta(\tau)$ et vice versa. D'autre part, la juxtaposition de deux lettres composées $w_i w_{i+1}$ ne crée pas de nouvelles excédances. En effet, si x (respectivement, y) est la dernière (respectivement, première) lettre de w_i (respectivement, w_{i+1}), on a $c(x) = 2$ ou 3 et $c(y) = 1$ ou 2 .

Enfin, chaque lettre composée de l'une des couleurs $[12]$, $[13]$, $[23]$, $[132]$ contient exactement une excédance et celle de couleur $[123]$ en contient deux. On en déduit que le nombre d'excédances de π est égal au nombre de lettres composées de Ω , les lettres de couleur $[123]$ étant comptées deux fois. Par conséquent,

$$\text{exc } \pi = |\Omega| + |\Omega_{123}|.$$

□

La Proposition 3 implique immédiatement l'identité

$$(10) \quad \sum_{\pi \in \mathcal{D}_{\mathbf{n}}} \beta^{\text{cyc } \pi} \gamma^{\text{exc } \pi} = \sum_{\Omega \in \Omega_{\mathbf{n}}} \sum_{\sigma \in \mathfrak{S}(\Omega)} \beta^{\text{cyc } \sigma} \gamma^{|\Omega| + |\Omega_{123}|},$$

où $\Omega_{\mathbf{n}}$ désigne l'ensemble des alphabets composés de $C_{\mathbf{n}}$ et $\mathfrak{S}(\Omega)$ désigne l'ensemble des permutations de Ω .

5. Démonstration du théorème. Soit $\Omega_{s,k}$ l'ensemble des alphabets composés ayant s lettres composées dont exactement k sont de couleur [123]. D'après (10), on a:

$$(11) \quad D(\mathbf{n}; \beta, \gamma) = \sum_{s \geq 0} \sum_{k \geq 0} \sum_{\Omega \in \Omega_{s,k}} \sum_{\sigma \in \mathfrak{S}(\Omega)} \beta^{\text{cyc } \sigma} \gamma^{|\Omega| + |\Omega_{123}|}.$$

Or pour chaque alphabet composé Ω de $\Omega_{s,k}$ on a:

$$(12) \quad \sum_{\sigma} \beta^{\text{cyc } \sigma} \gamma^{|\Omega| + |\Omega_{123}|} = \gamma^{k+s} \sum_{\pi} \beta^{\text{cyc } \pi} = \gamma^{k+s} (\beta)_s \quad (\sigma \in \mathfrak{S}(\Omega)).$$

Par ailleurs, d'après la Proposition 2, le cardinal de $\Omega_{s,k}$ est égal à:

$$(13) \quad \frac{n_1! n_2! n_3!}{(s-n_1)!(s-n_2)!(s-n_3)!(n_1+n_2+n_3-2s-k)!k!}.$$

En reportant (12) et (13) dans (11), on trouve:

$$\begin{aligned} D(\mathbf{n}; \beta, \gamma) &= \sum_{s \geq 0} \sum_{k \geq 0} \frac{n_1! n_2! n_3! \gamma^{s+k} (\beta)_s}{(s-n_1)!(s-n_2)!(s-n_3)!(n_1+n_2+n_3-2s-k)!k!} \\ &= \sum_{s \geq 0} \frac{n_1! n_2! n_3! (1+\gamma)^{n_1+n_2+n_3-2s}}{(s-n_1)!(s-n_2)!(s-n_3)!(n_1+n_2+n_3-2s)!} \gamma^s (\beta)_s. \end{aligned}$$

Ceci achève la démonstration du théorème. $\square$

6. Remarques. D'après (5), la démonstration ci-dessus se réduit à celle de (3) lorsque $\beta = 1$. Une autre démonstration combinatoire de (3) a été donnée implicitement par Foata [Fo, (10.2.8)]. Par ailleurs, l'arbitre nous a fait remarquer que MacMahon [Ma] a démontré (3) en appliquant son Master Theorem, donc on pourrait aussi déduire une démonstration combinatoire de (3) du travail de Cartier et Foata [Ca-Fo] ou de celui de Gessel [Ge1].

Les techniques esquissées ci-dessus permettent également de fournir des démonstrations combinatoires de certaines autres transformations de fonctions hypergéométriques liées directement à l'identité de Pfaff-Saalschütz. Par exemple, si l'on munit tout couple $(a, \pi) \in \mathcal{C}_{\mathbf{n}} \times \mathcal{D}_{\mathbf{n}}$ du poids: $W((a, \pi)) = (-1)^{\chi(a)} \gamma^{\text{exc } \pi}$ où $\chi(a) = 1$ si $c(a) > c(\pi(a))$ ou 0 dans le cas contraire. Le polynôme générateur correspondant de $\mathcal{C}_{\mathbf{n}} \times \mathcal{D}_{\mathbf{n}}$ est alors

$$\begin{aligned} CD(\mathbf{n}, \gamma) &= \sum_{(a, \pi)} W((a, \pi)) \quad ((a, \pi) \in \mathcal{C}_{\mathbf{n}} \times \mathcal{D}_{\mathbf{n}}) \\ &= \sum_{\pi \in \mathcal{D}_{\mathbf{n}}} [\text{exc } \pi - (n_1 + n_2 + n_3 - \text{exc } \pi)] \gamma^{\text{exc } \pi}. \end{aligned}$$

En évaluant $\mathcal{CD}(\mathbf{n}; \gamma)$ d'après (5) et (10), respectivement, on obtient après avoir simplifié

le facteur $n_1!n_2!n_3!$:

$$(14) \quad \sum_{k \geq 0} (2k - n_1 - n_2 - n_3) \binom{n_1}{k - n_3} \binom{n_2}{k - n_1} \binom{n_3}{k - n_2} \gamma^k \\ = \sum_{s \geq 0} \frac{(2s - n_1 - n_2 - n_3) \gamma^s (1 - \gamma) (1 + \gamma)^{n_1 + n_2 + n_3 - 2s - 1}}{(s - n_1)! (s - n_2)! (s - n_3)! (n_1 + n_2 + n_3 - 2s)!}.$$

Elle est équivalente à une transformation de ${}_4F_3$ très-bien balancée de Bailey [Bai, p. 97 (6)].

Notons, d'autre part, qu'en faisant $\gamma = -1$ dans (4), on trouve:

$$(15) \quad D(\mathbf{n}; \beta, -1) = \begin{cases} \frac{(-1)^s n_1! n_2! n_3! (\beta)_s}{(s - n_1)! (s - n_2)! (s - n_3)!} & \text{si } n_1 + n_2 + n_3 = 2s, \\ 0 & \text{si } n_1 + n_2 + n_3 \text{ est impair.} \end{cases}$$

En faisant maintenant $\beta = 1$ dans (15) et en tenant compte de (5), on trouve une formule due à l'origine à Dixon [Bai]:

$$\sum_{k \geq 0} (-1)^k \binom{n_1}{k - n_3} \binom{n_2}{k - n_1} \binom{n_3}{k - n_2} \\ = \frac{(-1)^s s!}{(s - n_1)! (s - n_2)! (s - n_3)!} \quad \text{si } n_1 + n_2 + n_3 = 2s, \\ 0 \quad \text{si } n_1 + n_2 + n_3 \text{ est impair.}$$

Cette formule a été utilisée par Gillis, Jedwab, et Zeilberger [Gi-Je-Ze] dans le calcul combinatoire des coefficients de linéarisation des polynômes de Legendre.

Pour terminer, signalons que la formule (2) a été q -généralisée par Zeilberger [Zei] lorsque $n_1 + n_2 + n_3$ est *paire*. Une légère modification de sa démonstration donne celle du cas général, comme signalé dans [Ze2]. La formule ainsi obtenue est équivalente à l'identité de q -Pfaff-Saalschütz (cf. [Bai]). Enfin, un q -analogue de (3) peut se déduire facilement de celui de (2).

REFERENCES

- [As] R. ASKEY, *Orthogonal Polynomials and Special Functions*, CBMS-NSF Regional Conference Series in Applied Mathematics 21, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1975.
- [Bai] W. N. BAILEY, *Generalized Hypergeometric Series*, Cambridge University Press, Cambridge, 1935.
- [Ca-Fo] P. CARTIER AND D. FOATA, *Problèmes combinatoires de permutations et réarrangements*, Lecture Notes in Mathematics 85, Springer-Verlag, Berlin, 1969.
- [De] J. DÉARMÉNIEU, *Une autre interprétation du nombre des dérangements*, Actes 8^e Séminaire Lotharingien de Combinatoire, Publ. I.R.M.A. Strasbourg, 229/S-08, 1984, pp. 11-16.
- [Er] W. N. ERDÉLYI, W. MAGNUS, F. OBERHETTINGER, AND F. G. TRICOMI, *Higher Transcendental Functions*, Vol. 1, McGraw-Hill, New York, 1953.
- [Fo] D. FOATA, *Étude algébrique de certains problèmes d'analyse combinatoire et du calcul des probabilités*, Publ. Inst. Statist. Univ. Paris, 14 (1965), pp. 81-241.
- [Ge1] I. GESSEL, *A combinatorial proof of the multivariable Lagrange inversion formula*, J. Combin. Theory Ser. A, 45 (1987), pp. 187-195.
- [Ge2] ———, *A coloring problem*, Amer. Math. Monthly, to appear.

- [Ge-St] I. GESSEL AND D. STANTON, *Short proofs of Saalschütz's and Dixon's theorems*, J. Combin. Theory Ser. A, 38 (1985), pp. 87–90.
- [Gi-Je-Ze] J. GILLIS, J. JEDWAB, AND D. ZEILBERGER, *A combinatorial interpretation of the integral of the product of Legendre polynomials*, SIAM J. Math. Anal., 19 (1988), pp. 1455–1461.
- [Go] I. J. GOOD, *Proofs of some "binomial" identities by means of MacMahon's "Master Theorem"*, Proc. Cambridge Philos. Soc., 58 (1962), pp. 161–162.
- [Ma] P. A. MACMAHON, *Collected Papers*, Vol. 1, MIT Press, Cambridge, MA, 1978.
- [Ze] D. ZEILBERGER, *A q -Foata proof of the q -Saalschütz identity*, European J. Combin., 8 (1987), pp. 461–463.
- [Ze1] J. ZENG, *Linéarisation des produits de polynômes de Charlier, Meixner et Krawtchouk*, SIAM J. Math. Anal., submitted.
- [Ze2] ———, *Pfaff-Saalschütz revisited*, J. Combin. Theory Ser. A., 51 (1989), pp. 141–143.

ADVANTAGEOUS PROPERTIES OF DUAL TRANSHIPMENT POLYHEDRA *

I. ADLER† AND S. COSARES‡

Abstract. The dual linear programs for the transshipment problem over a directed graph, $G = \{N, E\}$, are shown to have polyhedra with properties that make them well suited to vertex visiting solution techniques, like the simplex method. In particular, nondegenerate cases are shown to have feasible regions with considerably fewer extreme points than the feasible sets for primal problems. The adjacency structure of feasible bases is also shown to be quite favorable. In fact, the Hirsch Conjecture is valid when the network is complete. A dual-based simplex method for cases of the transshipment problem, which include the shortest paths problem, is presented that finds an optimal solution in no more than $\text{Min}(|E| - |N| + 1, |N||N - 1|/2)$ pivots.

Key words. linear programming, network flows, hirsch conjecture

AMS(MOS) subject classification. 09C08

1. Introduction. Consider a directed graph, G , having a node set,

$$N = \{0, 1, \dots, m\}$$

and an edge set, $E = \{e_k = (i, j) \mid \text{edge } k \text{ is directed from } i \text{ to } j, k = 1, \dots, n\}$. Let the vector $b \in \mathbb{R}^m$ represent a demand for flow at nodes $1, \dots, m$. When $b_i < 0$, we say that there is a supply at node i . Let c_{ij} represent the cost of sending one unit of flow across edge (i, j) . The transshipment problem is defined as the task of finding an assignment of flow to the edges that satisfies the demand at each node at minimum total cost.

A commonly used linear programming (LP) formulation for the transshipment problem is:

$$\begin{aligned} \text{TP: Min } & \sum_{(i,j) \in E} c_{ij}x_{ij} \\ \text{subject to: } & \sum_{i:(i,j) \in E} x_{ij} - \sum_{k:(j,k) \in E} x_{jk} = b_j \quad j = 1, \dots, m \\ & x_{ij} \geq 0. \end{aligned}$$

Its dual LP, (DTP), is formulated as follows:

$$\begin{aligned} \text{DTP: Max } & \sum_{i=1}^m b_i y_i \\ \text{subject to: } & y_j - y_i \leq c_{ij} \quad (i,j) \in E \\ & y_0 = 0. \end{aligned}$$

A constraint for node 0 is not included in the primal formulation since it would be redundant. For consistency, we set the (dummy) dual variable y_0 equal to 0. Let $A \in$

* Received by the editors July 18, 1988; accepted for publication (in revised form) June 14, 1989. This research was partially funded by United States Office of Naval Research contract N00014-87-K-0202.

† Department of Industrial Engineering and Operations Research, University of California, Berkeley, California 94720.

‡ Bell Communications Research, Piscataway, New Jersey 08854.

$\mathbb{R}^{m \times n}$ be the node-arc incidence matrix for G , without node 0. In particular,

$$a_{ik} = \begin{cases} -1 & \text{if } e_k \text{ is directed out of node } i \\ +1 & \text{if } e_k \text{ is directed into node } i \\ 0 & \text{otherwise.} \end{cases} \quad i = 1, \dots, m, \quad k = 1, \dots, n$$

The feasible region for TP can be represented by the system, $Ax = b$, $x \geq 0$, where A has row rank m and in each column there is no more than one $+1$ and no more than one -1 .

In [Tard85], an algorithm is presented which solves the transshipment problem in $O(n^2 m^3 \log m)$ arithmetic operations, thus resolving the long standing question of whether there exists an algorithm solving the problem in a number of steps that is bounded by a polynomial in n and m and is independent of the values of b_i and c_j , (i.e., a *strongly polynomial algorithm*). The method can be described as a primal-based algorithm since the iterates generated by it satisfy the constraints of TP. A dual framework to the algorithm has been developed in [Fuji85] and improved upon in [GaTa86] giving an $O(m^4 \log m)$ algorithm that solves the problem by way of the dual. This represents a significant theoretical improvement over the primal algorithm since n is of order m^2 in dense networks.

A Simplex method for the Transshipment problem is described in [Cunn76]. Here, the method for avoiding cycling under degeneracy by visiting only *strongly feasible bases* is introduced. (See [Cunn79] for a description of its theoretical underpinnings.) In [Orl85], the author shows that, in contrast to more general cases, the simplex method using Dantzig's pivot rule solves instances of TP having upper bounds on the variables in no more than $O[(u)mn^2 \log n]$ pivots, when solving the problem by way of the primal formulation, (where u represents the largest upper bound on a variable). For the dual formulation of the problem, Orlin ([Orl84]) presents a Simplex method that requires no more than $O(m^3 \log n)$ pivots, giving a strongly polynomial vertex visiting algorithm.

In an attempt to explain why the dual formulation appears to be better suited to vertex visiting methods, like the simplex method, considerable attention has been paid to linear programs for the dual of the transportation and assignment problems (DTAP). These problems arise from a transshipment problem over a bipartite graph with node sets N_1 and N_2 and edge set $\{(i, j) \mid i \in N_1, j \in N_2\}$. (In the assignment problem, $|N_1| = |N_2|$.) It had been shown that the polyhedra associated with DTAPs have considerably fewer vertices than those of the associated primal problems ([BaRu84]). The polyhedra have also been shown to satisfy the Hirsch Conjecture ([Bal84]). Coupled with these findings was the development of dual based methods for the problems having rather good performance guarantees. Goldfarb ([Gold85]) and Balinski ([Bal86]) give simplex algorithms solving the assignment problem that require no more than $O(|N_1|^3)$ operations. Similar techniques have been used to solve special cases of the Transportation problem in $O(|N_1|^2 |N_2| + |N_1| |N_2|^2)$ time ([KLS87]).

In the following sections, we analyze more general cases of the transshipment problem and show that the dual formulation has advantages over the primal for these problems as well. In § 2, we list some special properties of TP and DTP. We present a discussion about what happens when the node-arc incidence matrix A is a *Leontief Matrix*. An interesting case is the one in which A represents a complete directed graph, where between every pair of nodes i and j there exists a symmetric pair of edges (i, j) and (j, i) . In § 3, we show that the dual LPs for this case have considerably fewer basic feasible solutions than those of the primal, regardless of the values of b and c . In § 4, we comment on the adjacency structure of the feasible bases of DTP. We show that the *distance* between any

two extreme points is relatively small. In fact, the Hirsch Conjecture is shown to be valid for instances of DTP occurring over a complete directed graph. In § 5, we exploit these results to give a simplex method for cases of DTP which include the dual to the shortest paths problem, which require no more than

$$\text{Min } (|E| - |N| + 1, |N| |N - 1| / 2)$$

pivots.

2. Preliminaries. Let A_{*j} denote the j th column of A . For a set $S \subset \{1 \cdots n\}$, A_{*S} is defined as the submatrix of A consisting only of those columns with an index in S . A *basis* is defined as a set of m linearly independent columns of A . If A_{*S} is a basis, it is said to be *feasible* for TP if the unique (basic) solution to the system, $A_{*S}x_S = b$, is such that $x_S \geq 0$. This occurs if $A_{*S}^{-1}b$ is nonnegative. The basis A_{*S} is said to be feasible for DTP if the unique solution, y_S , to $y^T A_{*S} = c_S$, is such that

$$y_S^T A_{*j} \leq c_j, \quad \text{for all } j \notin S.$$

We say that the basis is *nondegenerate* if this set of inequalities holds strictly. If the basis is nondegenerate, the solution corresponds to a nondegenerate vertex of the polyhedron represented by the constraints of the problem. In such a case, no other basis could determine the solution y_S .

For the remainder of our discussion, we will assume that the vector c is such that there exists at least one dual feasible basis, i.e., that the network is connected and there are no directed cycles of negative total cost (so the optimal objective function value for TP is finite).

It is well known that a set of columns in A are linearly independent if the set of edges in the network associated with them do not form any cycles. So every collection of edges forming a spanning tree of the network corresponds to a basis. We will say a spanning tree is *primal feasible* or *dual feasible*, if it corresponds to a feasible basis for TP or DTP, respectively.

If T is a spanning tree, the subtree of T rooted at node i , T_i , has node set

$$N_{T_i} = \{j \in N \mid \text{the unique path in } T \text{ from } 0 \text{ to } j \text{ contains node } i\}.$$

A directed edge (i, j) in T is *downward* if the path from 0 to j in T contains node i and *upward* if it does not. A tree consisting solely of downward edges is called a *branching* rooted out of node 0. For a partition of the nodes into sets C and $\bar{C}$, where node 0 is in C , we define the edges that are directed from a node in C to a node in $\bar{C}$ to be the *forward edges* of the cut, $\{C, \bar{C}\}$, and the edges directed from $\bar{C}$ to C to be the *backward edges* of the cut $\{C, \bar{C}\}$.

When the matrix A has no more than one positive element in any column it is called a *Pre-Leontief Matrix*. The system $Ax = b$, $x \geq 0$ is called a *Pre-Leontief Substitution System* if, in addition, b is positive. When there is at least one solution to the system, we say that the system is a *Leontief Substitution System* (LSS) and A is a *Leontief Matrix*.

The properties of LSS's and their bases are discussed in great detail in [Dant55] and [Vein68]. LSS's have many characteristics not found in general LP's, which makes them worthy of special consideration.

We say that column j is in *substitution class* i , when the positive element of A_{*j} is in the i th row, and in *substitution class* 0, if $A_{*j} \leq 0$. The feasible bases for an LSS are such that there is exactly one member from each of Substitution Classes 1, $\cdots$, m . In addition, every feasible basis of an LSS has a nonnegative inverse, so if there is a solution for some $b > 0$, then there is a solution for any $b \geq 0$.

We find the following property particularly useful.

PROPOSITION 1. *If A is a Leontief Matrix, then $y^T A \leq c$, has a finite “maximal solution,” y^* , for all c for which the system is feasible. That is, every solution to the system, $\hat{y}$, is such that $\hat{y} \leq y^*$. Furthermore, the system $y^T A \geq c$, has a finite “minimal solution” for all c for which it is feasible.*

A proof of this result can be found in [CoVe72], where polyhedral sets with minimal (maximal) solutions are characterized.

If TP is feasible when $b > 0$, then A is a Leontief Matrix with a column of belonging to substitution class i if the associated edge is directed into node i . So the feasible bases for TP with $b > 0$ correspond to spanning trees with each node (except node 0) having in-degree one, which is a branching rooted out of node 0. If TP is feasible when $b < 0$, then the system, $(-A)x = -b$, $x \geq 0$ is an LSS, with substitution classes corresponding to the out-edges of nodes $1, \dots, m$, so feasible bases correspond to branchings rooted into node 0. Consequently, if the network contains at least one branching rooted out of (into) node 0, then, by Proposition 1, the system in DTP contains a maximal (minimal) solution.

3. The number of feasible bases of TP, DTP. In common worst-case analyses of vertex visiting algorithms, a relatively small number of feasible bases for a class of LP problems would indicate that the problems in the class are likely to be solved more quickly by the simplex method than more general problems.

Since the bases for transshipment problems must correspond to spanning trees in the network, we are able to provide the following bounds on the number of feasible bases for TP and its dual.

LEMMA 1. *The number of feasible bases for any nondegenerate instance of TP arising from a directed complete graph with $|N|$ nodes is exactly $|N|^{|N|-2}$.*

Proof. From Cayley’s Theorem, we know that an undirected complete graph with $|N|$ nodes contains $|N|^{|N|-2}$ spanning trees. Given one such tree, we find an associated feasible basis for TP as follows. If node i and node j are linked in the tree, and i is in the path from node 0 to node j , we choose for the basis the column of A corresponding to edge (i, j) if $\sum_{k \in N_{T_j}} b_k \geq 0$ or edge (j, i) if $\sum_{k \in N_{T_j}} b_k \leq 0$.

When $b > 0$, for instance, all edges in the tree are directed downward, which is a branching rooted out of node 0. If TP has degenerate bases, the sum of the demands at a subtree of some tree is equal to 0. We could associate more than one feasible basis with such a tree. $\square$

The number of spanning trees in the network which are dual feasible, it turns out, is considerably smaller than that of the primal, in fact, the following holds.

LEMMA 2. *The number of feasible bases for a nondegenerate, feasible instance of DTP arising from a directed complete graph with $|N|$ nodes can be as small as $|N|$.*

Proof. Since the graph is complete, it is guaranteed to contain a branching rooted into node 0 and a branching rooted out of node 0. So there are feasible bases corresponding to a minimal solution, y , and a maximal solution, $\bar{y}$. The polytope for DTP is contained within the region, $\{z \mid y \leq z \leq \bar{y}\}$. Any bounded, nondegenerate, m -dimensional polyhedron, such as this one, must contain at least $m + 1 = |N|$ affinely independent extreme points.

The tightness of the bound follows from the case of DTP where

$$c_{ij} = \begin{cases} 1 & j = i + 1 \text{ or } (i, j) = (m, 0) \\ M & \text{otherwise} \end{cases} \quad i, j = 0, \dots, m$$

where M is a number larger than m . Notice that, by summing the constraints associated with the set of edges, $\{(i, i+1), (i+1, i+2), \dots, (j-1, j)\}$, and the set, $\{(j, j+1), \dots, (m, 0), \dots, (i-1, i)\}$, we obtain implicit constraints

$$-m + (j - i - 1) \leq (y_j - y_i) \leq j - i$$

for every pair of nodes, i and j , where $i < j$. This implies that all but the $m + 1$ constraints associated with edges, $(0, 1), (1, 2), \dots, (m, 0)$ are redundant. The $m + 1$ basic feasible solutions for this case of DTP are enumerated as follows:

$$y_i^K = \begin{cases} i & i < K \\ -m + i - K & i \geq K \end{cases} \quad i = 1, \dots, m \quad K = 1, \dots, m+1. \quad \square$$

The number of feasible bases for DTP is different for differing values of the vector c . This is in contrast to the dual of transportation and assignment problems, (DTAPs). In [BaRu84], the authors show that the feasible bases of any nondegenerate instance over a complete bipartite graph, $B = \{N_1, N_2, (N_1 \times N_2)\}$, are in one-to-one correspondence with a set of *signature vectors*, $s \in \mathbb{Z}^{|N_1|}$, satisfying $s_i > 0$, and $\sum_{i=1}^{|N_1|} s_i = |N_1| + |N_2| - 1$. So the number of feasible bases is $\binom{|N_1| + |N_2| - 1}{|N_1| - 1}$.

THEOREM 1. *The tightest upper bound on the number of feasible bases for a nondegenerate, feasible instance of DTP arising from a directed complete graph with $|N|$ nodes is $\binom{2(|N| - 1)}{|N| - 1}$.*

Proof. Consider the following dual formulation of an instance of the assignment problem, where $N_1 = N_2 = \{0, \dots, m\}$:

$$\begin{aligned} \hat{D}: \quad & \text{Max } e^T u - e^T v \\ \text{subject to: } & u_j - v_i \leq c_{ij} \quad i \in N_1, \quad j \in N_2, \quad i \neq j \\ & u_j - v_i \leq 0 \quad i \in N_1, \quad j \in N_2, \quad i = j \\ & v_0 = 0. \end{aligned}$$

Independent of the values in c , the polytope associated with $\hat{D}$ has no more than $\binom{2m}{m} = \binom{2(|N| - 1)}{|N| - 1}$ vertices. One of the faces of the polytope,

$$F = \{(u, v) \mid u_j - v_j = 0, j = 0, \dots, m\},$$

is equivalent to the nondegenerate, feasibility set of DTP, thus the number of feasible bases for DTP is bound as stated.

The upper bound is tight when $c_{ij} = i(|N| - j)$, since a one-to-one correspondence can be drawn between the feasible bases for this case and a set of *degree vectors*, $d \in \mathbb{Z}^{m+1}$, satisfying $d_i \geq 0$, and $\sum_{i=0}^m d_i = |N| - 1$. There are $\binom{2(|N| - 1)}{|N| - 1}$ such vectors.

Let $d = (d_0, \dots, d_m)$ be some such degree vector. Notice that at least one element from this vector must be equal to 0. So let $d = \{d_S, d_T\}$, where $S = \{k \mid d_k > 0\}$ and $T = \{k \mid d_k = 0\}$.

Consider the following subset of the constraints from the instance of DTP:

$$\begin{aligned} y_j - y_i &\leq i(|N| - j) \quad i \in S, \quad j \in T \\ y_0 &= 0. \end{aligned}$$

Notice that this set of constraints resembles that of the dual formulation of some instance of the Transportation problem (DTAP), which is known to be nondegenerate since it is equivalent to a *cross-free* case, as described in [BaRu84]. Notice also that d_S constitutes

a valid signature vector for this DTAP. Let $z = \{z_S, z_T\}$ be the unique basic solution associated with this signature.

If z also satisfies the remaining inequalities from the instance of DTP, we can conclude that it is a basic feasible solution to the instance at hand. If the constraints are satisfied with strictness, then z is a unique, nondegenerate solution corresponding with d .

The constraints that are binding at z correspond to edges that comprise a tree spanning nodes $0, \dots, m$. Since a tree is a connected subgraph, we know that the following is true.

For any $j \in T$ there exists an $\hat{i} \in S$ such that

$$\begin{aligned} z_j - z_{\hat{i}} &= \hat{i}(|N| - j), \\ z_k - z_{\hat{i}} &\leq \hat{i}(|N| - k) \quad \text{for all other } k \in T \end{aligned}$$

implying that, $z_k - z_j \leq \hat{i}(j - k)$. Since $0 \leq \hat{i}, j, k < |N|$, we can conclude that $z_k - z_j < j(|N| - k)$. $j, k \in T$.

Also, for any $i \in S$ there exists a $\hat{j} \in T$ such that

$$\begin{aligned} z_j - z_i &= i(|N| - \hat{j}), \\ z_j - z_h &\leq h(|N| - \hat{j}) \quad \text{for all other } h \in S \end{aligned}$$

so, $z_i - z_h \leq (h - i)(|N| - \hat{j})$. Since $0 \leq \hat{j}, i, h < |N|$, we see that

$$z_i - z_h < h(|N| - i). \quad i, h \in S.$$

And, for any pair $h \in S, k \in T$ there exist $\hat{i} \in S$ and $\hat{j} \in T$ such that

$$z_k - z_{\hat{i}} = \hat{i}(|N| - k), \quad z_j - z_h = h(|N| - \hat{j})$$

where

$$z_j - z_{\hat{i}} \leq \hat{i}(|N| - \hat{j})$$

so, $z_h - z_k \leq \hat{i}(k - \hat{j}) - h(|N| - \hat{j}) < k(|N| - h)$.

Therefore, the all of the constraints of this instance of DTP are satisfied; and z is one of $\binom{2(|N|-1)}{|N|-1}$ nondegenerate, basic feasible solutions. $\square$

COROLLARY. *The number of d -dimensional faces in the polytope for DTP arising from a directed complete graph with $|N|$ nodes, when feasible and nondegenerate, is no more than*

$$\binom{2(|N|-1)-d}{|N|-1} \binom{|N|-1}{d}.$$

Proof. The number of bounded faces of the polytope for an instance of DTAP has been established in [BaRu84]. This bound holds for DTP since its entire feasible region is on face F of the DTAP, $\hat{D}$, described above. When $c_{ij} = i(m + 1 - j)$, all of the vertices, hence all of the bounded faces of the polytope of $\hat{D}$ lie on F , so the bound is tight. $\square$

We wish to point out that the face, F , described in the proof, is the optimal face for problem $\hat{D}$, whenever c is such that DTP is feasible. (Notice that the value of the objective function must be less than zero whenever $u_j < v_j$ for some j .) Therefore, a (Phase I) feasible starting basis for DTP can be found by way of an efficient strongly polynomial simplex method by solving $\hat{D}$ using either the method of [Bal86] or [Gold85].

4. The “distance” between feasible bases of DTP. Two feasible bases, B and $\hat{B}$, are *adjacent* if they have $m - 1$ columns of A in common. To travel from feasible basis

B to an adjacent basis requires a *pivot*. To describe a pivot for DTP, consider the formulation with the addition of slack variables:

$$\begin{aligned} \text{DTP: Max } & \sum_{i=1}^m b_i y_i \\ \text{subject to } & y_j - y_i + z_{ij} = c_{ij} \quad (i, j) \in E \\ & y_0 = 0 \quad z_{ij} \geq 0. \end{aligned}$$

If (i, j) is in the tree T associated with feasible basis B then $z_{ij} = 0$. Increasing some z_{ij} from 0 corresponds to dropping (i, j) from T , which cuts the nodes into sets, C and $\bar{C}$ (where node 0 belongs to C).

If (i, j) is an upward edge, then $\bar{C}$ is the set of nodes in the subtree of T rooted at node i . To maintain feasibility, any increase in z_{ij} of Δ must be accompanied by an increase of Δ in each of y_k , $k \in \bar{C}$. This results in a $\Delta(\sum_{k \in \bar{C}} b_k)$ change in the objective function value.

Thus, the (possibly negative) *reduced cost* of dropping (i, j) from the tree is equal to the total demand for flow at the subtree rooted at node i .

As y_k , $k \in \bar{C}$ increases, the slack in the constraints associated with the forward edges of $\{C, \bar{C}\}$ decreases. If no forward edge exists, then Δ can be made arbitrarily large, without violating any constraints. Otherwise, Δ is limited by the slack in the constraints associated with these edges. When a constraint becomes binding, we obtain a new solution, $\hat{y}$. The now binding constraint is called the *entering* constraint, giving a new basis $\hat{B}$ and dual feasible tree $\hat{T}$.

If there is degeneracy, there may be more than one candidate for the entering constraint during a pivot. It is also possible that some candidate entering constraint is already binding, so a pivot may occur without changes in the values of y and z .

If (i, j) is a downward edge, a pivot can be similarly described. The reduced cost of dropping (i, j) from the tree is equal to the total supply at the nodes in the subtree rooted at node j . In a nondegenerate pivot, the entering edge is backward in the cut $\{C, \bar{C}\}$. Under degeneracy, however, there may exist candidate entering edges which are forward.

The *distance* between two feasible bases is defined to be the smallest number of pivots necessary to get from one to the other. If the distance between every pair of bases for an LP is relatively small, then there is reason to believe that the LP is fairly well suited to some vertex visiting method.

Suppose some instance of TP is feasible for $b > 0$. Then, as we have seen, there is a dual feasible basis B^* , which determines a maximal solution to DTP, y^* , and has an associated tree T^* , which is a branching rooted out of node 0.

For some dual feasible basis, B^k , with associated tree, T^k , and solution, y^k , let

$$S^k = \{0\} \cup \{i \in \{1, \dots, m\} \mid \text{there exists a path from 0 to } i \text{ in } T^k \cap T^*\}.$$

Note that $i \in S^k$ implies that $y_i^k = y_i^*$.

LEMMA 3. If $(i, j) \in T^k$ and $i \in S^k$ then $y_j^k = y_j^*$.

Proof. Since $(i, j) \in T^k$ and $i \in S^k$, $y_j^k - y_i^k = c_{ij} = y_j^k - y_i^*$. Since y^* is a feasible solution, $y_j^* - y_i^* \leq c_{ij}$. So, $y_j^k \geq y_j^*$. But since y_j^* is maximal, $y_j^k = y_j^*$. $\square$

LEMMA 4. If $(i, j) \in T^*$ and $y_j^k = y_j^*$ then $y_i^k = y_i^*$.

Proof. Since $(i, j) \in T^*$ and $y_j^k = y_j^*$, $y_j^* - y_i^* = c_{ij} = y_j^k - y_i^*$. Since y^k is a feasible solution, $y_j^k - y_i^k \leq c_{ij}$. So, $y_i^k \geq y_i^*$. But since y_i^* is maximal, $y_i^k = y_i^*$. $\square$

THEOREM 2. *Suppose B^* is a feasible basis for DTP arising from a network with $|N|$ nodes and $|E|$ edges. If B^* corresponds to a branching rooted out of node 0, then the distance between B^* and any other feasible basis, say B^0 , is bounded by*

$$\min \left(|E| - |N| + 1, \binom{|N|}{2} \right).$$

Proof. We construct a sequence of pivots that starts with B^0 and terminates with B^* which is no longer than the bound stated.

Suppose k pivots have been performed, giving dual feasible solution y^k and dual feasible tree, T^k . While $T^k \neq T^*$, choose as the *next edge* to drop from T^k any edge (i, j) such that exactly one of i or j is a member of S^k .

An edge (p, q) is chosen from the set of candidate entering edges, which is forward in the cut $\{C, \bar{C}\}$, giving a new tree, T^{k+1} , and a new feasible solution, y^{k+1} . (We know that forward (p, q) exists, else the branching T^* doesn't exist.)

There are three possible occurrences, shown in the following cases.

Case 1. $p \notin S^k$. Then, $S^{k+1} = S^k$. While this event does not contribute to the progress toward B^* , we will make assurances that (i, j) is not an entering edge in any subsequent pivots. (Notice that (i, j) could never enter again if it is an upward edge, since the only entering edges are either directed away from S^{k+1} or are disjoint from it.)

Case 2. $p \in S^k$ and $(p, q) \in T^*$. Then with respect to T^{k+1} , $q \in S^{k+1}$, so $|S^{k+1}| \geq |S^k| + 1$.

Case 3. $p \in S^k$ and $(p, q) \notin T^*$. In this case, (p, q) may have been an edge in a previous tree, so we would rather not allow this edge to enter. We choose to select a different entering edge that satisfies either Case 1 or Case 2. We will show, with respect to the new solution y^{k+1} , that such an edge exists.

From Lemma 3 we know that since (p, q) is a candidate entering edge, y_q^{k+1} must be equal to its terminal value y_q^* , which implies that the solution y^{k+1} is degenerate, (i.e., there is more than one basis corresponding to y^{k+1} and (p, q) is not the only allowable downward entering edge for the pivot). An alternative entering edge can be found as follows:

Set $h = q$.

Let (g, h) be the unique edge in T^* directed into node h . By Lemma 4, $y_g^{k+1} = y_g^*$. If $g \notin \bar{C}$, then (g, h) is an acceptable entering edge. Otherwise, set h equal to g and repeat.

Suppose, initially, that T^0 has no edges in common with T^* . By construction, edges belonging to T^* are never to be dropped. Since there are no more than $|E| - |N| + 1$ edges not belonging to T^* , and no edge can be dropped more than once, the distance between B^0 and B^* is bounded by $|E| - |N| + 1$.

For the case when the graph is dense, we should notice that if $|S^0| = s$, then the number of valid next edges is bounded by $t = |N| - s$. It is impossible for all t of the valid next edges to be dropped, without an occurrence of Case 2, (otherwise there would be no edges in T^* joining S^t to $\bar{S}^t$, which is impossible since T^* is a spanning tree).

So, if $|S^0| = 1$, (i.e., $S^0 = \{0\}$), there can be no more than $|N| - 1$ consecutive occurrences of Case 1. Hence, the total number of pivots, K , guaranteeing that $|S^K| = |N|$ is bounded by $|N| - 1 + |N| - 2 + \cdots + 1 = \binom{|N|}{2}$. $\square$

We point out that if $\hat{B}$ is a dual feasible branching rooted into node 0, it corresponds to the minimal solution of the system, $y^T A \leq c$. Since the feasible bases of the system, $y^T A \leq c$ are in one-to-one correspondence with the feasible bases of the system,

$y^T(-A) \leq c$, $\hat{B}$ could be found by a similar sequence of pivots as is used to find the basis for the maximal solution of $y^T(-A) \leq c$.

Also, for many transshipment problems, the choice of root node 0 is arbitrary. By renumbering the nodes, we see that the basis for a dual feasible branching rooted out of or into some node r , if one exists, is never more than $\min(|E| - |N| + 1, \binom{|N|}{2})$ pivots away from the other dual feasible bases.

COROLLARY. *If some feasible basis of DTP, say B^0 , has k columns of A in common with the basis for a dual feasible branching, then the distance between the two bases is bounded by $\min(|E| - |N| + 1, \binom{|N|-k}{2})$.*

Proof. Even if $S^0 = \{0\}$, there cannot be more than $|N| - k - 1$ valid next edges, since edges that are in T^* are never dropped. So the total number of pivots necessary to guarantee that $|S^K| = |N|$, is bounded by

$$|N| - k - 1 + |N| - k - 2 + \cdots + 1 = \binom{|N| - k}{2}. \quad \square$$

Nondegenerate polytopes of dimension d , being determined by k linear inequalities, are said to satisfy the Hirsch Conjecture if the distance between any pair of feasible bases is bounded by $k - d$. Cases of TP and DTP known to have polytopes satisfying the Hirsch Conjecture include the assignment problem ([Bal74]), special cases of the transportation problem ([BaRu74]), DTAPs over complete bipartite graphs ([Bal84]), the shortest path problem ([Saig69]), and more generally, TPs that are also LSSs ([Grin71]). The above results allow us to extend this list to include some other cases of DTP.

THEOREM 3. *The polytope for DTP arising from a directed complete graph with $|N|$ nodes satisfies the Hirsch conjecture. That is, the distance between any two feasible bases is bounded by $(|N| - 1)^2$.*

Proof. Let B^1 and B^2 be dual feasible bases. Since the graph is complete, TP is feasible when $b > 0$ and when $b < 0$, so DTP has feasible bases corresponding to branchings rooted out of and into node 0. By Theorem 2, the distance between B^1 and any of these bases is bounded by $\binom{|N|}{2}$.

Let e be an edge incident to node 0 in the tree, T^2 , associated with B^2 . If e is directed into (out of) node 0 then, by Lemma 3, T^2 shares edge e with some dual feasible branching, $\hat{B}$, rooted into (out of) node 0. By the Corollary of Theorem 2, the distance between B^2 and $\hat{B}$ is bounded by $\binom{|N|-1}{2}$. So the distance between B^1 and B^2 is bounded by $\binom{|N|}{2} + \binom{|N|-1}{2} = (|N| - 1)^2$. $\square$

5. The Simplex method for special cases of DTP. Having established that the feasible bases for the instance of DTP with n edges and $m + 1$ nodes are no more than m^2 pivots away from each other, we now attempt to determine just how well a simplex method is expected to perform on particular instances of the problem. Ideally, there would be a simple rule for choosing the entering and leaving variables for the pivots, which would follow the sequence of bases of minimum length, while improving the objective function at each step. For general LPs, there is no guarantee that such rules exist. (For DTP however, there is a rule which follows a sequence to an optimal basis containing no more than $O(m^3 \log m)$ pivots [Orl84].)

Let us divert our attention to a special case of the transshipment problem called the shortest (0 to all i) paths problem (SP), which is formulated as follows:

$$\begin{array}{ll} \text{SP: Min } c^T x & \text{DSP: Max } e^T y \\ \text{subject to } Ax = 1 & \text{subject to } A^T y \leq c \\ & x \geq 0 \end{array}$$

This problem has been shown to be easily solved by variants of Dijkstra's method ([Dijk59]). Until the time of this writing, the best simplex method requires $O(m^2 n \log m)$ pivots for the primal formulation and $O(m^3 \log m)$ pivots for the dual formulation, ([Orl85]). For this and other cases of TP and DTP we are able to prove the following theorem.

THEOREM 4. *When $b > 0$, as in, for example, the shortest paths problem, an optimal solution to DTP (DSP) can be found by the simplex method, starting from any feasible basis, after no more than $\min(|E| - |N| + 1, \binom{|N|}{2})$ pivots, by using Dantzig's Pivoting Rule.*

Proof. When $b > 0$, feasible bases corresponding to the maximal solution, y^* , are optimal. Starting from feasible basis, B , with corresponding tree, T , the result holds if Dantzig's rule chooses to drop an edge that is a valid next edge as is defined in the proof of Theorem 2. This would have to be the edge with the largest reduced cost. Recall that it will be acceptable to terminate with any dual feasible basis that corresponds to a branching rooted out of node 0.

As we have seen, the reduced cost of dropping an upward tree edge, (i, j) , is equal to the total flow demanded from the subtree rooted at i , which is positive when $b > 0$. (When all $b_k = 1$, the reduced cost is precisely the number of nodes in the subtree at node i .)

Let $S = \{0\} \cup \{i \in \{1, \dots, m\} \mid \text{The path from } 0 \text{ to } i \text{ in } T \text{ consists of downward edges.}\}$

Suppose that the edge $(i, j) \in T$ has the largest reduced cost. Edge (i, j) must be an upward edge, where node j belongs to S , otherwise the path in T from node 0 to j would have to contain some other upward edge, (g, h) . In that case, both of nodes i and j would be in the subtree of T , rooted at g , contradicting that (i, j) has the largest reduced cost.

Since all of the dropped edges are upward, we can always choose an entering edge that is forward if a branching exists. This edge satisfies either Case 1 or Case 2 as in the proof of Theorem 2, so (i, j) is a valid next edge. The Simplex method terminates when there are no more upward tree edges, so the terminal basis corresponds to a dual feasible branching, which is optimal. $\square$

When $b \geq 0$, feasible bases corresponding to the maximal solution, y^* , are also optimal. In this case, however, it is possible that some edge with the largest reduced cost is not a valid next edge. By modifying the rule so that the edge with the largest reduced cost, which also has one of its nodes belonging to the set S , is chosen, we could solve this case of TP with the same performance bound as above.

Notice that cases of DTP where $b \leq 0$ can be solved analogously. These cases can be posed as minimization problems where the next edge chosen is the one with the smallest reduced cost.

6. Remarks. The number of basic feasible solutions and the distance between them has been used as a yardstick on how well or how poorly we could expect a vertex visiting method to perform on a given problem. Since, when using a simplex method, each pivot must result in an improvement to the value of the objective function, (improvements of 0 are acceptable), other interesting measures for the suitability of an LP to the method can be introduced. They include the longest and shortest *monotonic distance* between feasible bases. This is, respectively, the length of the longest and shortest sequence of pivots between two feasible bases, when we insist on the monotonicity of some linear function. The simplex method could do no better than travel to the optimal basis along the shortest monotonic path and no worse than travel along the longest path, so these measures would be quite useful in the attempt to provide tighter bounds on the expected performance of a simplex method for these problems.

Similar measures have yet to be developed for judging the relative suitability of a particular LP to interior point methods. These are algorithms for which the iterates move within the interior of the feasible region, hence allowing for the possibility of shortcutting what would otherwise be long sequences of pivots. The addition of such measures make way for a better informed decision about which solution technique to use and whether to solve the primal or dual formulation of the problem at hand. It would be interesting to determine whether these measures, once developed, would also show a preference to solving the transshipment problem via DTP rather than TP.

We have noted that some of the results reported above can be attributed to the fact that the constraint matrix for the transshipment problem is pre-Leontief. Since this is also true for generalized transshipment problems, where the flow across each edge is scaled by some positive constant, we expect to find favorable results here as well. Preliminary investigations have shown that this is indeed the case.

Acknowledgment. The support of the United States Office of Naval Research is gratefully acknowledged.

REFERENCES

- [Bal74] M. L. BALINSKI, *On two special classes of transportation polytopes*, Math. Programming Stud., 1 (1974), pp. 43–58.
- [Bal84] ———, *The Hirsch conjecture for dual transportation polyhedra*, Math. Oper. Res., 9 (1984), pp. 629–633.
- [Bal86] ———, *A competitive (dual) simplex method for the assignment problem*, Math. Programming, 34 (1986), pp. 125–141.
- [BaRu74] M. L. BALINSKI AND A. RUSSAKOFF, *On the assignment polytope*, SIAM Rev., 16 (1974), pp. 516–525.
- [BaRu84] ———, *Faces of dual transportation polyhedra*, Math. Programming Stud., 22 (1984), pp. 1–8.
- [CoVe72] R. W. COTTLE AND A. F. VEINOTT, JR., *Polyhedral sets having a least element*, Math. Programming, 3 (1972), pp. 238–249.
- [Cunn76] W. H. CUNNINGHAM, *A network simplex method*, Math. Programming, 11 (1976), pp. 105–116.
- [Cunn79] ———, *Theoretical properties of the network simplex method*, Math. Oper. Res., 4 (1979), pp. 196–208.
- [Dant55] G. B. DANTZIG, *Optimal solution of a dynamic Leontief model with substitution*, Econometrica, 23 (1955), pp. 295–302.
- [Dijk59] E. W. DIJKSTRA, *A note on two problems in connexion with graphs*, Numer. Math., 1 (1959), pp. 269–271.
- [Fuji85] S. FUJISHIGE, *Capacity-rounding algorithm for the minimum-cost circulation problem: a dual framework of the Tardos algorithm*, Institute of Socio-Economic Planning, Report No. 254 (85-3), University of Tsukuba, Japan.
- [GaTa86] Z. GALIL AND E. TARDOS, *An $O(n^2 \log n(m + n \log n))$ minimum cost flow algorithm*, MSRI Technical Report, Berkeley, CA, January 1986.
- [Gold85] D. GOLDFARB, *Efficient dual simplex algorithms for the assignment problem*, Math. Programming, 33 (1985), pp. 187–203.
- [Grin71] R. C. GRINOLD, *The Hirsch conjecture in Leontief substitution systems*, SIAM J. Appl. Math., 21 (1971), pp. 483–485.
- [KLS87] P. KLEINSCHMIDT, C. W. LEE, AND H. SCHANNATH, *Transportation problems which can be solved by the use of Hirsch-paths for the dual problems*, Math. Programming, 37 (1987), pp. 153–168.
- [Orl84] J. B. ORLIN, *Genuinely polynomial simplex and non-simplex algorithms for the minimum cost flow problem*, Sloan Working Paper 1615-84, Sloan School of Management, Massachusetts Institute of Technology, 1984.
- [Orl85] ———, *On the simplex algorithm for networks and generalized networks*, Math. Programming Stud., 24 (1985), pp. 166–178.
- [Saig69] R. SAIGAL, *A proof of the Hirsch conjecture on the polyhedron of the shortest route problem*, SIAM J. Appl. Math., 17 (1969), pp. 1232–1238.
- [Tard85] E. TARDOS, *A strongly polynomial minimum cost circulation algorithm*, Combinatorica, 5 (1985), pp. 247–255.
- [Vein68] A. F. VEINOTT, JR., *Extreme points of Leontief substitution systems*, Linear Algebra Appl., 1 (1968), pp. 181–194.

HARMONIC ANALYSIS OF POLYNOMIAL THRESHOLD FUNCTIONS*

JEHOSHUA BRUCK†

Abstract. The analysis of linear threshold Boolean functions has recently attracted the attention of those interested in circuit complexity as well as of those interested in neural networks. Here a generalization of linear threshold functions is defined, namely, polynomial threshold functions, and its relation to the class of linear threshold functions is investigated.

A Boolean function is polynomial threshold if it can be represented as a sign function of a polynomial that consists of a polynomial (in the number of variables) number of terms. The main result of this paper is showing that the class of polynomial threshold functions (which is called PT_1) is strictly contained in the class of Boolean functions that can be computed by a depth 2, unbounded fan-in polynomial size circuit of linear threshold gates (which is called LT_2).

Harmonic analysis of Boolean functions is used to derive a necessary and sufficient condition for a function to be an S -threshold function for a given set S of monomials. This condition is used to show that the number of different S -threshold functions, for a given S , is at most $2^{(n+1)|S|}$. Based on the necessary and sufficient condition, a lower bound is derived on the number of terms in a threshold function. The lower bound is expressed in terms of the spectral representation of a Boolean function. It is found that Boolean functions having an exponentially small spectrum are not polynomial threshold. A family of functions is exhibited that has an exponentially small spectrum; they are called “semibent” functions. A function is constructed that is both semibent and symmetric to prove that PT_1 is properly contained in LT_2 .

Key words. bent functions, Boolean functions, circuit complexity, harmonic analysis, lower bounds, threshold functions

1. Introduction. A Boolean function $f(X)$ is a *threshold function* if

$$f(X) = \text{sgn}(F(X)) = \begin{cases} 1 & \text{if } F(X) > 0 \\ -1 & \text{if } F(X) < 0 \end{cases}$$

where

$$F(X) = \sum_{\alpha \in \{0,1\}^n} w_\alpha X^\alpha$$

and

$$X^\alpha \stackrel{\text{def}}{=} \prod_{i=1}^n x_i^{\alpha_i}.$$

Throughout this paper a *Boolean function* will be defined as $f: \{1, -1\}^n \rightarrow \{1, -1\}$; namely, 0 and 1 are represented by 1 and -1 , respectively. It is also assumed, without loss of generality, that $F(X) \neq 0$ for all X .

A *threshold gate* is a gate that computes a threshold function. It can be shown that any Boolean function can be computed by a single threshold gate if we allow the number of monomials in $F(X)$ to be as large as 2^n . Although such a result is not interesting by itself, it stimulates the following natural question: What happens when the number of monomials (terms) in $F(X)$ is bounded by a polynomial in n ?

The question can be formulated by defining a new complexity class of Boolean functions. This class, called PT_1 for *polynomial threshold* functions, is made of all the Boolean functions that can be computed by a single threshold gate in which the number

* Received by the editors July 18, 1988; accepted for publication (in revised form) May 19, 1989. This work was supported by the Air Force Office of Scientific Research.

† IBM Almaden Research Center, 650 Harry Road, K54/802, San Jose, California 95120-6099.

of monomials is bounded by a polynomial in n . The main goal of this paper is the study of this complexity class and its relations with other known complexity classes of Boolean functions.

More precisely, let $S \subseteq \{0, 1\}^n$; a Boolean function f is an S -threshold function if there exist integers that we call weights (the w_α 's) such that $f(X) = \text{sgn}(\sum_{\alpha \in S} w_\alpha X^\alpha)$. Hence, a Boolean function $f(X)$ is in PT_1 if there exists a set S , with $|S|$ bounded by a polynomial in n , such that $f(X)$ is S -threshold. Notice that there is no restriction on the size of the weights.

A related class of functions is the class of *linear threshold* functions [7], [11]. A Boolean function is *linear threshold* if it is S -threshold with S corresponding to the constant and linear terms. We define LT_1 to be the class of functions that are computable by a single linear threshold gate. The next step is to define complexity classes that relate to circuits. Define $LT_d(PT_d)$ to be the class of Boolean functions that can be computed by an unbounded fan-in polynomial size circuit of depth at most d which consists of linear (polynomial) threshold gates.

Some motivation: recently, there has been a considerable interest in the study of the computational model of bounded depth unbounded fan-in polynomial size circuits that consist of linear threshold gates [5], [12], [14], [17]. This interest follows from recent results in complexity of circuits [8], [13], [16] which indicate that MAJORITY (hence, linear threshold functions) cannot be computed by a bounded depth unbounded fan-in polynomial size circuit that consists of $\vee$, $\wedge$, NOT, and PARITY gates. Thus, the next natural step in the analysis is adding MAJORITY as a possible gate in the computational model. Notice that in the results in [5] the weights are bounded by a polynomial in n . To make the distinction from the case in which the weights are not bounded we put "hats." Namely, $\widehat{LT}_d$ and $\widehat{PT}_d$ correspond to circuits with bounded weights. Using this notation, a related result in [5] is:

$$\widehat{LT}_1 \subset \widehat{LT}_2 \subset \widehat{LT}_3.$$

In this context, the study of circuits of polynomial threshold functions can be viewed as study of a model in which a single gate is rather powerful. Namely, there is no "trivial" function that cannot be computed by a single gate. For example, PARITY, EXACT $_k$ (output -1 if and only if k of its inputs are -1) and the characteristic function of a linear subspace (code) [1], [2], [10] are all in PT_1 but none are in LT_1 . This fact suggests that the separation between the classes PT_1 and PT_2 is not going to be an easy problem. The other motivation for this work comes from the area of neural networks [1], [6], where a linear threshold element is the basic processing element. One of the reasons for the limited computational power of neural networks is the fact that every neuron (node) is not very powerful. One solution to that is to consider generalized networks in which every node is computing a polynomial threshold function. In this context, our result is a characterization of generalized neural networks with respect to ordinary neural networks.

The main result in the paper is a characterization of the power of PT_1 with respect to the hierarchy of circuits of linear threshold functions. We have:

$$LT_1 \subset PT_1 \subset LT_2,$$

which also implies that $PT_1 \subset PT_2$.

Clearly, $LT_1 \subset PT_1$ follows from the fact that PARITY is not in LT_1 . But, PARITY(X) = $\text{sgn}(x_1 x_2 \cdots x_n)$, hence it is in PT_1 . Proving that $PT_1 \subseteq LT_2$ is based on the observation that PARITY does not require the full strength of a depth 2 circuit of linear threshold elements and is described in § 2. To prove that this containment is proper, we

developed a technique for deriving lower bounds for the number of monomials in a threshold function. This technique is based on the spectral representation of a Boolean function. Most of the paper is devoted to the development of this technique and its applications.

In § 3 we review the subject of harmonic analysis of Boolean functions [9] and show that every Boolean function has a representation as a polynomial over the rationals and hence as a threshold function.

In § 4 we use the spectral representation of Boolean functions and derive a necessary and sufficient condition for a function to be an S -threshold function for a given S . We use this condition to show that the number of different S -threshold functions, for a given S , is at most $2^{(n+1)|S|}$. These results turn out to be a generalization of known results for linear threshold functions [3], [7], [11].

In § 5 we use the necessary and sufficient condition to derive a lower bound on the number of monomials in a threshold function. The lower bound is expressed in terms of the spectral representation of a Boolean function. We find that Boolean functions that have an exponentially small spectrum are not polynomial threshold.

In § 6 we exhibit a family of functions that has an exponentially small spectrum; we call them “semibent” functions. We construct a function that is both semibent and symmetric to prove that PT_1 is properly contained in LT_2 . Finally, we address some open problems.

2. Simulation of polynomial threshold functions. It is a well-known result that PARITY (as well as other symmetric functions) is in LT_2 [5], [12]. From this fact it follows that a polynomial threshold function can be simulated by a depth 3 circuit of linear threshold gates. The idea is to compute the monomials using depth 2 circuits and combine the monomials in the gate in the third layer. What we will show here is that depth 2 is enough.

THEOREM 2.1. $PT_1 \subseteq LT_2$.

Proof. The idea is to notice that PARITY does not require the full power of a depth 2 circuit of linear threshold gates. Actually, PARITY can be realized by a set of linear threshold elements in the first layer while, in the second layer, we need only to sum and add a constant to get the desired result. Namely, we do not have to use the threshold operation in the second layer.

Example. Let $f(X) = x_1 \oplus x_2$. Let

$$F_1(X) = -1 - x_1 - x_2 \quad \text{and} \quad F_2(X) = -1 + x_1 + x_2.$$

It can be verified that:

$$f(X) = 1 + \operatorname{sgn}(F_1(X)) + \operatorname{sgn}(F_2(X)).$$

Note that we are using the $\{1, -1\}$ representation instead of $\{0, 1\}$, respectively. The above is true in general for PARITY of n variables. In the general case we need up to $n + 1$ linear threshold gates in the first layer and again only summation and addition of a constant in the second layer. Using this observation a polynomial threshold function can be simulated by depth 2 linear threshold circuit in a way similar to that done with depth 3. $\square$

Proving containment of polynomial threshold functions in LT_2 turns out to be a very easy problem compared to the problem of proving that this containment is proper; the latter requires proving lower bounds. The rest of the paper is devoted to the development of a technique for getting lower bounds for polynomial threshold functions and to the application of this technique to getting separation results.

3. Polynomial representation of Boolean functions. In this section the representation of Boolean functions as polynomials over the field of rational numbers is presented.

DEFINITION. A *Hadamard matrix* of order m , denoted by H_m , is an $m \times m$ matrix of $+1$'s and -1 's such that:

$$(1) \quad H_m H_m^T = m I_m$$

where I_m is the $m \times m$ identity matrix. The above definition is equivalent to saying that any two rows of H are orthogonal.

Hadamard matrices of order 2^k exist for all $k \geq 0$. The so-called Sylvester construction is as follows [10]:

$$(2) \quad \begin{aligned} H_1 &= [1] \\ H_2 &= \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\ H_{2^{n+1}} &= \begin{bmatrix} H_{2^n} & H_{2^n} \\ H_{2^n} & -H_{2^n} \end{bmatrix}. \end{aligned}$$

DEFINITION. Given a Boolean function f of order n , p is a polynomial (with coefficients over the field of rational numbers) *equivalent* to f if and only if for all $X \in \{1, -1\}^n$:

$$f(X) = p(X).$$

As an example, let $f = x_1 \oplus x_2$; that is, f is the XOR function of two variables. It is easy to check that in the $\{1, -1\}$ representation $p(x_1, x_2) = x_1 x_2$. Notice that for every Boolean function f , the polynomial p is linear in each of its variables because $x^2 = 1$ for $x \in \{-1, 1\}$. It is known that every Boolean function has a unique representation as a polynomial [2], [9]. This representation is derived by using the Hadamard matrix, as described by the following theorem.

THEOREM 3.1. *Let f be a Boolean function of order n . Let p be a polynomial equivalent to f . Let A_{2^n} denote the vector of coefficients of p . Let P_{2^n} denote the vector of the 2^n values of p (and f). Then:*

1. *The polynomial p always exists and is unique.*
2. *The coefficients of p are computed as follows:*

$$A_{2^n} = \frac{1}{2^n} H_{2^n} P_{2^n}.$$

Proof (sketch). The proof is constructive. The idea is to compute A_{2^n} by solving a system of linear equations. $\square$

Example. Let f be the AND function of two variables,

$$(3) \quad f(x_1, x_2) = x_1 \wedge x_2.$$

By Theorem 3.1,

$$(4) \quad p(x_1, x_2) = \frac{1}{2} (1 + x_1 + x_2 - x_1 x_2).$$

The entries of the vector A are denoted by $\{a_\alpha \mid \alpha \in \{0, 1\}^n\}$ and called the spectral representation of a function. Note that a_α is the coefficient of X^α in the polynomial representation where $X^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$. Hence, every Boolean function can be written as:

$$f(X) = \sum_{\alpha \in \{0, 1\}^n} a_\alpha X^\alpha.$$

The above method is applicable not only to Boolean functions but also to any function of the form $f : \{1, -1\}^n \rightarrow \mathbb{R}$.

4. Necessary and sufficient conditions and their applications. We use the polynomial representation of Boolean functions presented in the previous section to derive a necessary and sufficient condition for a function to be an S -threshold function for arbitrary S . This result turned out to be a generalization of a known result for linear threshold functions [3], [7], [11].

Let $f(X) = \text{sgn}(F(X))$ be a threshold function. Without loss of generality, assume that $F(X) \neq 0$ for all $X \in \{1, -1\}^n$. The following simple lemma enables us to express the relation between $f(X)$ and $F(X)$ in a global way. That is, instead of having 2^n conditions we have only one.

LEMMA 4.1. *Let $f(X)$ be a Boolean function and let $F(X) \neq 0$ for all $X \in \{1, -1\}^n$; then*

$$(5) \quad f(X) = \text{sgn}(F(X)) \quad \forall X \in \{1, -1\}^n$$

if and only if

$$(6) \quad \sum_{X \in \{1, -1\}^n} |F(X)| = \sum_{X \in \{1, -1\}^n} f(X)F(X).$$

Proof. Suppose there exists an X_1 that violates (5); this implies $(F(X) \neq 0 \text{ for all } X)$ that

$$|F(X_1)| > f(X_1)F(X_1).$$

Hence, (6) is also not true because violation in (5) can only decrease the value on the right-hand side of (6). Clearly, if (5) is true so is (6). $\square$

LEMMA 4.2.

$$(7) \quad \sum_{X \in \{1, -1\}^n} X^\alpha = \begin{cases} 2^n & \text{if } \alpha = \text{all-0 vector} \\ 0 & \text{else.} \end{cases}$$

Proof. The proof follows from the fact that X^α corresponds to a row of a Sylvester type Hadamard matrix (see Theorem 3.1). $\square$

The necessary and sufficient condition follows from (6) by using the polynomial representation of a Boolean function.

THEOREM 4.1. *Fix $S \subseteq \{0, 1\}^n$. Let $F(X) = \sum_{\alpha \in S} w_\alpha X^\alpha$. Let $f(X)$, $X \in \{-1, 1\}^n$, be a Boolean function with spectral representation $\{a_\alpha | \alpha \in \{0, 1\}^n\}$. Then:*

$$(8) \quad f(X) = \text{sgn}(F(X)) \quad \forall X \in \{1, -1\}^n$$

if and only if

$$\sum_{X \in \{1, -1\}^n} |F(X)| = 2^n \sum_{\alpha \in S} w_\alpha a_\alpha.$$

Proof. By the Lemma 4.1 it is enough to show that

$$\sum_{X \in \{1, -1\}^n} f(X)F(X) = 2^n \sum_{\alpha \in S} w_\alpha a_\alpha.$$

We write $f(X)$ as a polynomial:

$$f(X) = \sum_{\alpha \in \{0, 1\}^n} a_\alpha X^\alpha$$

and find that the constant term of $f(X)F(X)$ is $\sum_{\alpha \in S} w_\alpha a_\alpha$. Hence, the result follows from Lemma 4.2. $\square$

Theorem 4.1 is interesting because it suggests that an S -threshold function is fully characterized by the set of spectral coefficients that correspond to S . To see the power of the condition we present two applications. The first application (Theorem 4.2) is a generalization of the known result that PARITY is not in LT_1 and the second application (Theorem 4.3) is an upper bound on the number of S -threshold functions.

THEOREM 4.2. *Let*

$$\text{PARITY}(X) = \begin{cases} -1 & \text{odd number of } -1\text{'s in } X \\ 1 & \text{otherwise.} \end{cases}$$

Then PARITY is not an S -threshold function for any S such that the all-1 vector is not in S , i.e., $(1, 1, \dots, 1) \notin S$.

Proof. Note that $\text{PARITY}(X) = x_1 x_2 \cdots x_n$; hence, for a set S that does not include the all-1 vector we have

$$\sum_{X \in \{1, -1\}^n} |F(X)| \neq 2^n \sum_{\alpha \in S} w_\alpha a_\alpha = 0.$$

Thus, the condition cannot be satisfied ($F(X) \neq 0$). $\square$

A nice interpretation of this result is as follows: the X^α form a basis for the space of Boolean functions. Hence, by the definition of a basis, an element of the basis cannot be represented as a linear combination of the other elements in the basis. What Theorem 4.2 is saying is that it is also impossible to represent an element of the basis as the *sign* of a linear combination of the other elements in the basis.

LEMMA 4.3. *Let f_1 and f_2 be Boolean functions with $\{a_\alpha^i \mid \alpha \in \{0, 1\}^n\}$, $i = 1, 2$, as their spectral representation, respectively. Assume that at least one of the functions is an S -threshold function. Then $f_1(X) = f_2(X)$ for all $X \in \{1, -1\}^n$ if and only if $a_\alpha^1 = a_\alpha^2$ for all $\alpha \in S$.*

Proof. Suppose $f_1 = f_2$. By the uniqueness of the spectral representation (Theorem 3.1) we get the “only if.” Assume without loss of generality that f_1 is an S -threshold function. Now suppose $a_\alpha^1 = a_\alpha^2$ for all $\alpha \in S$. By Theorem 4.1 and the assumption that f_1 is S -threshold we get that there exists a set of weights that satisfies (8) for both f_1 and f_2 . Hence, f_2 is also S -threshold and $F_1(X) = F_2(X)$ for all $X \in \{1, -1\}^n$. Thus, $f_1 = f_2$. $\square$

COROLLARY 4.1. *Consider the set $S \subseteq \{0, 1\}^n$. Let f_1 and f_2 be Boolean functions of n variables. If $a_\alpha^1 = a_\alpha^2$ for all $\alpha \in S$, then either both f_1 and f_2 are S -threshold or both are not S -threshold.*

One application of the above is counting the number of different S -threshold functions.

THEOREM 4.3. *Fix $S \subseteq \{0, 1\}^n$. There are at most $2^{(n+1)|S|}$ different S -threshold functions.*

Proof. It can be shown that for all $\alpha \in \{0, 1\}^n$, a_α can assume at most $(2^n + 1)$ different values. Hence, there are at most $2^{(n+1)|S|}$ different sets of $|S|$ spectral coefficients. Thus, the result follows from Corollary 4.1. $\square$

The above turns out to be a generalization of a known [7], [11] upper bound on the number of linear threshold functions for which $|S| = n + 1$.

5. Lower bounds. The necessary and sufficient condition that is derived above is used to derive lower bounds on the number of monomials in a threshold function, again, by using the spectral representation. Let $f(X) = \text{sgn}(F(X))$ be an S -threshold function, namely,

$$F(X) = \sum_{\alpha \in S} w_\alpha X^\alpha.$$

We want to find lower bounds for $|S|$ as a function of the spectral representation of $f(X)$.

LEMMA 5.1. *For all $\alpha \in S$:*

$$2^n |w_\alpha| \leq \sum_{X \in \{1, -1\}^n} |F(X)|.$$

Proof. First we prove the statement for α being the all-0 vector:

$$\begin{aligned} \sum_{X \in \{1, -1\}^n} |F(X)| &= \sum_{F(X) > 0} F(X) - \sum_{F(X) < 0} F(X) \\ &= \sum_{X \in \{1, -1\}^n} F(X) - 2 \sum_{F(X) < 0} F(X) \\ &\stackrel{(a)}{=} 2^n w_{00 \dots 00} - 2 \sum_{F(X) < 0} F(X) \\ &\geq 2^n w_{00 \dots 00}. \end{aligned}$$

Note that (a) follows from Lemma 4.2. The proof for arbitrary α follows from the fact that $|X^\alpha| = 1$; hence:

$$|F(X)| = |X^\alpha| |F(X)| = |X^\alpha F(X)|.$$

Hence, we can make any w_α be the constant term without changing the value of $|F(X)|$. If $w_\alpha < 0$ we take $-F(X)$ and get the result. $\square$

THEOREM 5.1. *Let $f(X) = \text{sgn}(\sum_{\alpha \in S} w_\alpha X^\alpha)$ be an S -threshold function and let $\{a_\alpha | \alpha \in \{0, 1\}^n\}$ be its spectral representation; then*

$$(i) \quad |S| \geq \left(\sum_{\alpha \in S} p_\alpha |a_\alpha| \right)^{-1}$$

$$(ii) \quad \geq \hat{a}^{-1}$$

where

$$p_\alpha = \frac{|w_\alpha|}{\sum_{\alpha \in S} |w_\alpha|}$$

and

$$\hat{a} = \max_{\alpha \in S} |a_\alpha|.$$

Proof. We first prove (i). By Theorem 4.1 and Lemma 5.1, for all $\alpha \in S$:

$$|w_\alpha| \leq \sum_{\alpha \in S} w_\alpha a_\alpha.$$

We sum the above inequality over all $\alpha \in S$ and get:

$$\begin{aligned} \sum_{\alpha \in S} |w_\alpha| &\leq |S| \sum_{\alpha \in S} w_\alpha a_\alpha \\ &\leq |S| \sum_{\alpha \in S} |w_\alpha| a_\alpha. \end{aligned}$$

Therefore, we get (i). For (ii), just note that $p_\alpha \geq 0$ and $\sum p_\alpha = 1$. $\square$

We summarize this section by the following corollary.

COROLLARY 5.1. *Fix any $\varepsilon > 0$. Let $f(X)$ be a Boolean function of n variables. If $|a_\alpha| \leq 2^{-\varepsilon n}$ for all $\alpha \in \{0, 1\}^n$, then, for n sufficiently large, $f(X)$ is not a polynomial threshold function.*

6. Separating by semibent functions. We use Corollary 5.1 to get separation results by looking at functions that have an exponentially small spectrum.

DEFINITION. A Boolean function $f(X)$ is called “bent” [4], [10], [15] if and only if $|a_\alpha| = 2^{-n/2}$ for all $\alpha \in \{0, 1\}^n$. Notice that bent functions are defined for even n only.

PROPOSITION 6.1. *The Inner Product Mode 2 (IP2) function, i.e.,*

$$f(X) = (x_1 \wedge x_2) \oplus (x_3 \wedge x_4) \oplus \cdots \oplus (x_{2n-1} \wedge x_{2n})$$

is a bent function.

Proof. See [10]. A sketch of an alternative proof: it can be proven by induction on n that IP2, when written as a vector, is actually an eigenvector (with eigenvalue $= 2^n$) of a Sylvester type Hadamard matrix of order 2^{2n} . Hence, $|a_\alpha| = 2^{-n}$ for all $\alpha \in \{0, 1\}^n$. $\square$

THEOREM 6.1. $PT_1 \subset PT_2$.

Proof. By Corollary 5.1 and Proposition 6.1 the function IP2 is not in PT_1 . But it is in PT_2 : the AND’s are computed in the first level and the XOR is computed by the gate in the second layer. $\square$

DEFINITION. Let $\varepsilon > 0$ be fixed. A Boolean function $f(X)$ is called “semibent” if for all $\alpha \in \{0, 1\}^n$, $|a_\alpha| \leq 2^{-\varepsilon n}$. Clearly, a bent function is also a semibent function.

THEOREM 6.2. $PT_1 \subset LT_2$.

Proof. The fact that $PT_1 \subseteq LT_2$ is proved in Theorem 2.1. To show that it is a proper containment we must find a function that is in LT_2 but not in PT_1 . Every symmetric function is in LT_2 [5], [12] and every semibent function is not in PT_1 (Corollary 5.1). Hence, a natural candidate for such a function will be a symmetric semibent function. Indeed, a symmetric semibent function exists for all n as stated in Proposition 6.2 below. $\square$

Consider the function:

$$(9) \quad f(X) = (x_1 \wedge x_2) \oplus (x_1 \wedge x_3) \oplus \cdots \oplus (x_{n-1} \wedge x_n).$$

Hence, $f(X)$ consists of the sum mod 2 of all the $\binom{n}{2}$ possible AND’s between pairs of variables. We call this function the Complete Quadratic (CQ) function. Clearly, CQ is a symmetric function.

PROPOSITION 6.2. *CQ is bent for n even, and semibent for n odd.*

Proof. Actually, we can compute the exact spectral representation of CQ for every n ; see the Appendix.

7. Concluding remarks. We proved that the class of polynomial threshold functions is strictly contained in the class of functions computable by depth-2 linear threshold circuits. The technique we are using is based on harmonic analysis (polynomial representation) of Boolean functions. One interesting future direction is to try and prove other (even known) results in circuit complexity using this technique. For example, it is possible to prove that $\widehat{LT}_2 \subset \widehat{LT}_3$ (see [5]) using those techniques [1]. However, the main open problem is to find the exact relation between the class $LT^0 = \cup_{d \in \mathcal{A}} LT_d$ and the well known class NC^1 . In [17] Yao proved separation in the monotonic version of LT^0 . Here, we were concentrating on the relation between subclasses of PT^0 and subclasses of LT^0 , with the goal of getting separation in LT^0 . In particular we have the following conjecture.

CONJECTURE. For all $d \in \mathcal{N}$: $LT_d \subset PT_d \subset LT_{2d}$.

Our main result is proving the above for $d = 1$. A possible way to prove the conjecture is by induction on d ; clearly, we did not succeed in pursuing this direction.

Appendix.

A. The spectrum of the Complete Quadratic function. The Complete Quadratic function is defined in § 6. Here we prove Proposition 6.2. We start by giving an equivalent definition of the function CQ.

PROPOSITION A.1.

$$\text{CQ}(X) = \begin{cases} 1 & \text{no. of } -1\text{'s in } X = 0 \text{ or } 1 \bmod 4 \\ -1 & \text{otherwise.} \end{cases}$$

Proof. Suppose there are m -1 's in X . Since a pair in (9) is -1 if and only if both variables are -1 , we have exactly $\binom{m}{2}$ pairs that are -1 . Hence, the value of $\text{CQ}(X)$ is determined by the evenness of $\binom{m}{2}$ and the result follows. $\square$

First we calculate the spectrum for the case when n is even.

PROPOSITION A.2. Let $\{a_\alpha \mid \alpha \in \{0, 1\}^n\}$ be the spectral representation of $\text{CQ}(X)$. Assume that n is even, then

$$|a_\alpha| = 2^{-n/2}, \quad \forall \alpha \in \{0, 1\}^n.$$

Proof. The proof is by induction on n . For $n = 2$ we have

$$\text{CQ}(x_1, x_2) = \frac{1}{2}(1 + x_1 + x_2 - x_1 x_2).$$

Assume this is true for n and show that the statement is true for $(n + 2)$. We use the same notation as in § 3, namely, P_{2^n} represents the vector with the values of CQ and A_{2^n} represents the vector of the spectral coefficients of CQ. Using Proposition A.1 it can be shown that $P_{2^{n+2}}$ can be expressed as a function of P_{2^n} :

$$P_{2^{n+2}} = \begin{bmatrix} P_{2^n} \\ \hat{P}_{2^n} \\ \hat{P}_{2^n} \\ -P_{2^n} \end{bmatrix}$$

where

$$\hat{P}_{2^n} = \hat{X}_{2^n} \circ P_{2^n}.$$

$\hat{X}_{2^n}$ is the vector representation of $\text{PARITY}(X) = x_1 x_2 \cdots x_n$ and “ $\circ$ ” is bitwise multiplication.

Hence, by Theorem 3.1

$$\begin{aligned} A_{2^{n+2}} &= \frac{1}{2^{n+2}} H_{2^{n+2}} P_{2^{n+2}} \\ &= \frac{1}{2^{n+2}} \begin{bmatrix} H_{2^n} & H_{2^n} & H_{2^n} & H_{2^n} \\ H_{2^n} & -H_{2^n} & H_{2^n} & -H_{2^n} \\ H_{2^n} & H_{2^n} & -H_{2^n} & -H_{2^n} \\ H_{2^n} & -H_{2^n} & -H_{2^n} & H_{2^n} \end{bmatrix} \begin{bmatrix} P_{2^n} \\ \hat{P}_{2^n} \\ \hat{P}_{2^n} \\ -P_{2^n} \end{bmatrix} \\ &= \frac{1}{2} \begin{bmatrix} \hat{A}_{2^n} \\ A_{2^n} \\ A_{2^n} \\ -\hat{A}_{2^n} \end{bmatrix} \end{aligned}$$

where $\hat{A}_{2^n}$ is the reflection of A_{2^n} . Hence, if the result is true for n it is also true for $n + 2$. $\square$

Example. Using the above recursive description of the spectrum of $CQ(X)$ we can calculate A_{16} from A_4 :

$$A_4^T = \frac{1}{2}(1, 1, 1, -1),$$

and

$$A_{16}^T = \frac{1}{4}(-1, 1, 1, 1, \quad 1, 1, 1, -1, \quad 1, 1, 1, -1, \quad 1, -1, -1, -1).$$

The above is true for n even. For n odd we have the following proposition.

PROPOSITION A.3. *Let $\{a_\alpha \mid \alpha \in \{0, 1\}^n\}$ be the spectral representation of $CQ(X)$. Assume that n is odd, then*

$$|a_\alpha| = 0 \text{ or } 2^{-(n-1)/2} \quad \forall \alpha \in \{0, 1\}^n.$$

The proof is similar to the even case. We use induction on n and can write the recursive description of the spectrum.

Example. Let $n = 3$ then

$$CQ(x_1, x_2, x_3) = \frac{1}{2}(x_1 + x_2 + x_3 - x_1 x_2 x_3).$$

Acknowledgment. I thank Professor J. W. Goodman for his helpful comments.

REFERENCES

- [1] J. BRUCK, *Computing with networks of threshold elements*, Ph.D. Thesis, Stanford University, Stanford, CA, March 1989.
- [2] J. BRUCK AND M. BLAUM, *Neural networks, error-correcting codes and polynomials over the binary n-cube*, IBM Research Report RJ 6003, 1987; IEEE Trans. Inform. Theory, September 1989.
- [3] C. K. CHOW, *On the characterization of threshold functions*, in Proc. Symp. on Switching Circuit Theory and Logical Design, New York, American Institute of Electrical Engineers, 1961, pp. 34–38.
- [4] J. F. DILLON, *Elementary Hadamard difference sets*, in Proc. 6th S-E Conf. Combinatorics, Graph Theory and Computing, (Utilitas Math. Winnipeg, 1975), pp. 237–249.
- [5] A. HAJNAL, W. MAASS, P. PUDLÁK, M. SZEGEDY, AND G. TURÁN, *Threshold circuits of bounded depth*, Proceeding of the 28th IEEE Symposium of Foundations of Computer Science, 1987, pp. 99–110.
- [6] J. J. HOPFIELD, *Neural networks and physical systems with emergent collective computational abilities*, in Proceedings of the National Academy of Sciences USA, Vol. 79, 1982, pp. 2554–2558.
- [7] S. HU, *Threshold Logic*, University of California Press, Berkeley and Los Angeles, California, 1965.
- [8] D. S. JOHNSON, *The NP-completeness column: An ongoing guide*, J. Algorithms, 7 (1986), pp. 289–305.
- [9] R. J. LECHNER, *Harmonic analysis of switching functions*, in Recent Development in Switching Theory, A. Mukhopadhyay, ed., Academic Press, New York, 1971.
- [10] F. J. MACWILLIAMS AND N. J. A. SLOANE, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam, New York, 1977.
- [11] S. MUROGA, *Threshold Logic and its Application*, John Wiley, New York, 1971.
- [12] I. PARBERRY AND G. SCHNITGER, *Parallel computation with threshold functions*, in: Structure in Complexity Theory, Lecture Notes in Computer Science, vol. 223, 1986, pp. 272–290.
- [13] A. A. RAZBOROV, *Lower Bounds for the Size of Circuits of Bounded Depth with Basis $\{\wedge, \odot\}$* , Math. Notes, 41 (1987), pp. 333–338.
- [14] J. H. REIF, *On threshold circuits and polynomial computation*, in Proc. 2nd Structure in Complexity Theory Conf., 1987, pp. 118–123.
- [15] O. S. ROTHBAUS, *On “bent” functions*, J. Combin. Theory, Ser. A, 20 (1976), pp. 300–305.
- [16] R. SMOLENSKY, *Algebraic methods in the theory of lower bounds for Boolean circuit complexity*, in 19th Symposium on Theory of Computing, 1987, pp. 77–82.
- [17] A. C. YAO, *Circuits and local computations*, in 21st Symposium on Theory of Computing, 1989.

QUASI-SYMMETRIC 3-DESIGNS AND ELLIPTIC CURVES*

A. R. CALDERBANK† AND PATRICK MORTON‡

Abstract. A quasi-symmetric t -design is a t -design with two block intersection sizes p and q (where $p < q$). Quasi-symmetric 3-designs are classified with $p = 1$. The only nontrivial examples are the 4-(23, 7, 1) Witt design, and its residual, a 3-(22, 7, 4) design. This proves a conjecture of Sane and Shrikhande. The method is to reduce the classification problem to that of finding all integer points on the elliptic curves $y^2 = x^3 - 11x^2 + 32x$ and $y^2 = x^3 - 4x + 4$.

Key words. 3-designs, quasi-symmetric, elliptic curves, Skolen's method

AMS(MOS) subject classifications. 05B05, 11G05

1. Quasi-symmetric designs. A t -(v, k, λ) design is a collection $\mathfrak{B}$ of subsets of a v -set (called *blocks*), such that every block contains k points, and any set of t points is in exactly λ blocks. In a t -design, let λ_i denote the number of blocks containing a given set of i points, with $0 \leq i \leq t$. The identities

$$(1) \quad \lambda_i \binom{k-i}{t-i} = \binom{v-i}{t-i} \lambda, \quad i = 0, 1, \dots, t,$$

restrict the possible parameter sets. A *symmetric design* is a 2-(v, k, λ) design, such that $\lambda_0 = v$, $\lambda_1 = k$, and any two blocks meet in λ points. A t -design with two block intersection sizes is said to be *quasi-symmetric*. We shall consider quasi-symmetric 3-(v, k, λ) designs, with block intersection sizes p and q , where $p < q$.

Calderbank [5] and Sane and Shrikhande [15] proved that the parameters of a quasi-symmetric 2-design $\mathfrak{B}$ satisfy

$$(2) \quad pq(v-1)(v-2) - k(k-1)(v-2)(p+q-1) + k(k-1)^2(k-2) \geq 0,$$

and that equality holds if and only if $\mathfrak{B}$ is a 3-design. Neumaier [13], [14, Prop. 12] had earlier derived an inequality equivalent to (2), but that is not as simple to state.

Sane and Shrikhande [15, Conj. 4.6] have conjectured that there exist only two nontrivial, quasi-symmetric 3-designs with $p = 1$ (we consider the designs formed by $(v-2)$ -subsets of a v -set to be trivial). The first example is the 4-(23, 7, 1) Witt design, with $q = 3$, and the second example is its residual, a 3-(22, 7, 4) design (also with $q = 3$). Both these designs are unique. The purpose of this paper is to prove this conjecture.

One interesting property of quasi-symmetric 2-designs is that the *block graph* is strongly regular (see Goethals and Seidel [10], or Cameron and van Lint [7]). The vertices of this graph are the blocks of design, and two blocks are joined if they meet in q points. Recall that a *strongly regular graph* is regular, and that the number of vertices joined to two given vertices z_1 and z_2 ($z_1 \neq z_2$), depends only on whether or not z_1 and

* Received by the editors February 14, 1989; accepted for publication (in revised form) July 14, 1989.

† Mathematical Sciences Research Center, AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

‡ Mathematics Department, Wellesley College, Wellesley, Massachusetts 02181.

z_2 are joined. The eigenvalues of the adjacency matrix of the block graph are

$$(3) \quad \begin{aligned} \rho_0 &= \frac{k(\lambda_1 - 1) - p(\lambda_0 - 1)}{q - p} && (\text{multiplicity} = 1), \\ \rho_1 &= \frac{\lambda_1 - \lambda_2 - (k - p)}{q - p} && (\text{multiplicity} = v - 1), \\ \rho_2 &= -\frac{(k - p)}{(q - p)} && (\text{multiplicity} = \lambda_0 - v), \end{aligned}$$

(see, for example, Shrikhande [16]). Since the eigenvalues are integers, we have the useful divisibility condition that $q - p$ divides $k - p$. For a quasi-symmetric 3-design, the number of blocks is given by

$$(4) \quad \lambda_0 = \left[\frac{(k-1)(v+k-3) - (p+q-1)(v-2)}{2(k-1)^2 - (p+q-1)(v-2)} \right] v$$

(this follows from (2) and [5, Thm. 1]). Setting $p = 1$ in (4) we obtain

$$(5) \quad \lambda_0 = \left[1 + \frac{(k-1)(v-k-1)}{2(k-1)^2 - q(v-2)} \right] v.$$

In § 2 we reduce the conjecture of Sane and Shrikhande to a number theoretic question by proving the following theorem.

THEOREM 1. *If $\mathfrak{B}$ is a quasi-symmetric 3- (v, k, λ) design with $p = 1$, then either*

- (i) $\mathfrak{B}$ is the 4-(23, 7, 1) Witt design with $q = 3$;
- (ii) $\mathfrak{B}$ is the 3-(22, 7, 4) residual of the Witt design with $q = 3$;
- (iii) $q \geq 5$, $k \geq 9$, and there is an integer solution (a, k) of

$$(6) \quad a^2 = k(4k^2 - 11k + 8),$$

with $q - 1 \mid k - 1$;

- (iv) $q = 4$, $k \geq 7$, and there is an integer solution (b, k) of

$$(7) \quad b^2 = 2k(k-1)(k-2) + 1,$$

with $3 \mid k - 1$.

In §§ 3–5 we prove the nonexistence of solutions to (6) and (7) by finding all integer points on the respective elliptic curves. It turns out that the curve in (6) has only a finite number of rational solutions (and rank 0), while (7) has an infinite number of rational solutions (rank 1). We note that (7) does have the large integer solution $(156, \pm 2729)$, but $3 \nmid 155$ so the divisibility condition is violated.

Cameron [6], [7] has classified quasi-symmetric 3-designs with $p = 0$ (and $v \geq 2k$).

THEOREM 2 (Cameron). *The possible quasi-symmetric 3-designs with $p = 0$ and $v \geq 2k$ are*

- (1) 3- $(4m, 2m, m - 1)$ designs with $q = m$ ($m \geq 2$). These are Hadamard 3-designs;
- (2) 3- $(m(m^2 + 3m + 1), m(m + 1), m - 1)$ designs with $q = m$ ($m \geq 2$);
- (3) 3-(112, 12, 1) design with $q = 2$;
- (4) 3-(496, 40, 3) design with $q = 4$.

Lam et al. [11] have shown the nonexistence of a 3-(112, 12, 1) design with $p = 0$ and $q = 2$. Sane and Shrikhande have also conjectured that the only quasi-symmetric 3-designs with $v \geq 2k$ are the designs listed in Theorem 2, the Witt design, or its residual.

2. The reduction. Throughout this section $\mathfrak{B}$ denotes a quasi-symmetric $3-(v, k, \lambda)$ design with $p = 1$ and $v > k + 2$. We shall apply the following theorem of Frankl and Füredi [9] (see Calderbank [5] for a short proof involving Hahn polynomials).

THEOREM 3 (Frankl and Füredi). *Let $\mathcal{F}$ be a family of k -subsets of a v -set such that $\mathcal{F}$ is a 1-design, and $|F_1 \cap F_2| \geq d$, for all $F_1, F_2 \in \mathcal{F}$. Then $v \leq (k^2 - k + d)/d$ with equality if and only if $\mathcal{F}$ is a symmetric $2-(v, k, d)$ design.*

The next lemma appears as part of Theorem 4.5 in [15]. For completeness we include a short proof.

LEMMA 4.

$$(i) \quad v \leq \frac{(k-2)(k-3)}{q-2} + 3,$$

$$(ii) \quad \lambda \geq q.$$

Proof. We contract $\mathfrak{B}$ to obtain a $2-(v-1, k-1, \lambda)$ design that is not symmetric. Fisher's inequality (see [7, Chap. 1]) then gives $\lambda_2 \geq k$. We contract $\mathfrak{B}$ once more to obtain a $1-(v-2, k-2, \lambda)$ design in which any pair of distinct blocks meet in $q-2$ points. Part (i) now follows from Theorem 3. If we pick a block B in the twice contracted design, and count pairs (p, B') , where $B' \neq B$, $p \in B \cap B'$, then we obtain $(k-2)(\lambda-1) = (\lambda_2-1)(q-2)$. Since $\lambda_2 \geq k$, it follows that $\lambda \geq q$.

LEMMA 5. *If $q = 5$ then $\lambda = 5$ or 6 , and if $q \geq 6$ then $\lambda = q$.*

Proof. From (1) and (5) we have

$$(8) \quad \lambda = \frac{k(k-1)(k-2)}{(v-1)(v-2)} \left(1 + \frac{(k-1)(v-k-1)}{2(k-1)^2 - q(v-2)} \right) = q + e,$$

for some $e \geq 0$. Now (2) gives

$$(9) \quad q = \frac{k(k-1)^2(k-2)}{(k(k-1) - (v-1))(v-2)},$$

so we may eliminate q from (8) to obtain

$$(10) \quad 0 = \frac{-f(v, k, e)}{(v-2)(v-k^2+k-1)(2v-k^2-2)},$$

where

$$(11) \quad \begin{aligned} f(v, k, e) = & 2ev^3 - [(3k^2 - 2k + 8)e + k(k-2)]v^2 \\ & + [(k^4 - k^3 + 9k^2 - 6k + 10)e + k^2(k-4)]v \\ & - (2k^4 - 2k^3 + 6k^2 - 4k + 4)e + k^5 - 5k^4 + 7k^3 - 3k^2 + 2k. \end{aligned}$$

It is possible, of course, to verify (10) by hand, but it is much easier to use a symbolic manipulation package such as MACSYMA. By Lemma 3 we may suppose

$$(12) \quad k+2 < v \leq \frac{(k-2)(k-3)}{q-2} + 3.$$

Let $v_1 = k + 2$, $v_2 = (k - 2)(k - 3)/4 + 3$, and $v_3 = (k - 2)(k - 3)/3 + 3$. If $q \geq 6$ then $v \in [v_1, v_2]$, and if $q = 5$ then $v \in [v_1, v_3]$. Direct calculation gives

$$\begin{aligned} f(v_1, k, e) &= (k^5 - 4k^4 + k^3 + 6k^2 + 2k)e + k^5 - 5k^4 + 7k^3 - 3k^2 + 2k, \\ 32f(v_2, k, e) &= (3k^6 + k^5 - 101k^4 + 331k^3 + 106k^2 - 1820k + 1960)e \\ &\quad - 2k^6 + 64k^5 - 362k^4 + 940k^3 - 1304k^2 + 784k, \\ 27f(v_3, k, e) &= (2k^6 + 12k^5 - 132k^4 + 230k^3 + 768k^2 - 2952k + 2592)e \\ &\quad - 3k^6 + 72k^5 - 405k^4 + 1068k^3 - 1476k^2 + 864k. \end{aligned}$$

Since $q \geq 5$, and since $q - 1$ is a proper divisor of $k - 1$, we may suppose $k \geq 9$. It follows that

$$(13) \quad \begin{aligned} f(v_1, k, e) &> 0, & \text{for all } e \geq 0, \\ f(v_2, k, e) &> 0, & \text{for all } e \geq 1, \text{ and} \\ f(v_3, k, e) &> 0, & \text{for all } e \geq 2. \end{aligned}$$

Differentiating $f(v, k, e)$ twice with respect to v gives

$$(14) \quad \frac{\partial^2 f}{\partial v^2} = (12v - 6k^2 + 4k - 16)e - 2k^2 + 4k.$$

For $e \geq 0$, we have $\partial^2 f / \partial v^2 < 0$ on the interval $[v_1, v_3]$. If $q \geq 6$, $e \geq 1$, then $f(v, k, e)$ is never zero for $v \in [v_1, v_2]$, since the curve $f(v, k, e)$ lies above the line connecting $f(v_1, k, e)$ and $f(v_2, k, e)$. We conclude that if $q \geq 6$, then $e = 0$ and $\lambda = q$. Similarly, if $q = 5$, $e \geq 2$, then $f(v, k, e)$ is never zero for $v \in [v_1, v_3]$. Thus if $q = 5$, then $\lambda = 5$ or 6.

LEMMA 6. *If $q = 5$ then $\lambda = 5$.*

Proof. By Lemma 5, we have $\lambda = 5$ or 6. Suppose $q = 5$ and $\lambda = 6$. Then (2) gives

$$v - 2 = (k^2 - k - 1 \pm \sqrt{(k^2 - k - 1)^2 - 4k(k - 1)^2(k - 2)})/2,$$

and by Theorem 3 we take the minus sign. Substituting this value for v in (8) gives the following 12th degree equation in k .

$$\begin{aligned} &k^{12} + 163k^{11} + 194k^{10} + 1045k^9 - 45164k^8 + 275835k^7 \\ &\quad - 841918k^6 + 757751k^5 + 1358875k^4 - 1958866k^3 \\ &\quad - 689716k^2 + 1530600k - 1468800 = 0. \end{aligned}$$

There are no integral roots.

LEMMA 7. *If $q \geq 5$ then there exists an integer solution (a, k) to the equation $a^2 = k(4k^2 - 11k + 8)$, with $k \geq 9$.*

Proof. Lemmas 5 and 6 imply $\lambda = q$ so that $f(v, k, 0) = 0$, where $f(v, k, e)$ is the function defined in (11). Now

$$-f(v, k, 0) = k(k - 2)[v^2 - (k + 2)v - k^3 + 3k^2 - k + 1],$$

so that

$$(15) \quad v^2 - (k + 2)v - k^3 + 3k^2 - k + 1 = 0.$$

The discriminant of (15) is $k(4k^2 - 11k + 8)$ which must be an integral square. Since $q - 1$ is a proper divisor of $k - 1$ we may suppose $k \geq 9$.

LEMMA 8. *If $q = 4$ then there exists an integer solution (b, k) to the equation $b^2 = 2k(k-1)(k-2) + 1$, with $k \geq 7$.*

Proof. Equation (2) becomes

$$(16) \quad z^2 - (k^2 - k - 1)z + \frac{k(k-1)^2(k-2)}{4} = 0,$$

where $z = v - 2$. The discriminant of (16) is $2k(k-1)(k-2) + 1$ which must be an integral square. Since $q-1$ properly divides $k-1$ we have $k \geq 7$.

LEMMA 9. *If $q \leq 3$ then $q = 3$, $k = 7$, and $v = 22$ or 23 .*

Proof. Equation (2) becomes

$$(17) \quad z^2 - (k^2 - k - 1)z + \frac{k(k-1)^2(k-2)}{q} = 0,$$

where $z = v - 2$.

For $q = 2$, the discriminant of (17) is $d^2 = -k^4 + 6k^3 - 11k^2 + 6k + 1$, and the only solution is $k = 3$. Then $v = 4$ or 5 , and $v = 4$ is impossible. The solution $v = 5$, $k = 3$ corresponds to $\mathfrak{B}$ being the 3-subsets of a 5-set, and we consider this to be a trivial solution ($v \leq k + 2$).

For $q = 3$, the discriminant of (17) is $d^2 = -(k^4 - 10k^3 + 23k^2 - 14k - 3)/3$ and the only solutions are $k = 4$ or 7 . Now $q-1$ does not divide 3, so $k = 7$, and $v = 22$ or 23 . These are the parameters of the Witt design and its residual.

3. Finding all solutions to $a^2 = k(4k^2 - 11k + 8)$. We now determine all integral solutions to (6) and (7). Considering (6) first, any integer solution (a, k) of (6) is obviously an integer solution of

$$(4a)^2 = (4k)^3 - 11(4k)^2 + 32(4k),$$

and so it suffices to work on the elliptic curve

$$(18) \quad E: y^2 = x^3 - 11x^2 + 32x.$$

As is well known, the rational solutions of (18), together with the point at infinity (the solution $O = (0, 1, 0)$ on the projective curve $y^2z = x^3 - 11x^2z + 32xz^2$ corresponding to E) form a finitely generated abelian group $E(\mathbb{Q})$, with O as identity.

To compute $E(\mathbb{Q})$ we follow the method of descent as presented in Silverman [17, Chap. X]. In this descent, we first determine $E(\mathbb{Q})/2E(\mathbb{Q})$ using the isogeny (rational map)

$$(19) \quad \phi(x, y) = \left(\frac{y^2}{x^2}, \frac{y}{x^2}(32 - x^2) \right),$$

which maps E to the curve

$$(20) \quad E': Y^2 = X^3 + 22X^2 - 7X,$$

as well as the isogeny

$$\hat{\phi}(X, Y) = \left(\frac{y^2}{4x^2}, \frac{y}{8x^2}(-7 - x^2) \right)$$

from E' to E . At the points O and $(0, 0)$, at which (19) is undefined, we have

$$\phi(O) = \phi(0, 0) = O',$$

where O' is the point at infinity on E' , and similarly

$$\hat{\phi}(O') = \hat{\phi}(0, 0) = O.$$

The maps $\phi, \hat{\phi}$ are homomorphisms between $E(\mathbb{Q})$ and $E'(\mathbb{Q})$ with the kernels

$$\ker \phi = (O, (0, 0)), \quad \ker \hat{\phi} = (O', (0, 0)).$$

Further, $\hat{\phi} \cdot \phi$ is just multiplication by 2 on $E(\mathbb{Q})$. Hence $2(0, 0) = \hat{\phi} \cdot \phi(0, 0) = \hat{\phi}(O') = O$, and $(0, 0)$ is a point of order 2 on E . Since

$$-(x, y) = (x, -y),$$

on E and E' , it is easy to see that $(0, 0)$ is the only rational point of order 2 on E . (Any such point must have $y = 0$ in (18).)

The technique for determining $E(\mathbb{Q})/2E(\mathbb{Q})$ depends on two exact sequences. The first is

$$(21) \quad \frac{E'(\mathbb{Q})}{\phi(E(\mathbb{Q}))} \xrightarrow{\hat{\phi}} \frac{E(\mathbb{Q})}{2E(\mathbb{Q})} \rightarrow \frac{E(\mathbb{Q})}{\hat{\phi}(E'(\mathbb{Q}))} \rightarrow 0,$$

(see [17, p. 301]), where the second map is the natural inclusion, and the second is the sequence

$$(22) \quad 0 \rightarrow \frac{E'(\mathbb{Q})}{\phi(E(\mathbb{Q}))} \xrightarrow{\delta} S^\phi(E/\mathbb{Q}) \rightarrow \text{III}(E/\mathbb{Q})[\phi] \rightarrow 0$$

and the analogous

$$(22') \quad 0 \rightarrow \frac{E(\mathbb{Q})}{\hat{\phi}(E'(\mathbb{Q}))} \xrightarrow{\hat{\delta}} S^{\hat{\phi}}(E'/\mathbb{Q}) \rightarrow \text{III}(E'/\mathbb{Q})[\hat{\phi}] \rightarrow 0$$

obtained by reversing the roles of E and E' in (22).

In order to describe the maps $\delta, \hat{\delta}$ and the groups S^ϕ (the Selmer group) and III (the Tate–Shafarevic group) we need a little notation. In general, let E, E' be given by the curves

$$E: y^2 = x^3 + ax^2 + bx$$

$$E': y^2 = x^3 - 2ax^2 + (a^2 - 4b)x,$$

and let S consist of the infinite prime (valuation) on $\mathbb{Q}$ together with the primes dividing $2b(a^2 - 4b)$; then S contains the primes at which E and E' have so-called “bad reduction.” In our case

$$S = \{\infty, 2, 7\}.$$

Further let $\mathbb{Q}(S)$ be the group of rational numbers (modulo squares) which are squares except for possible factors from S . For us

$$\mathbb{Q}(S) = \{\pm 1, \pm 2, \pm 7, \pm 14\} \bmod Q^2.$$

Next, define the curve C_d (“principal homogeneous space for E ”) by

$$(23) \quad C_d: dw^2 = d^2 - 2adz^2 + (a^2 - 4b)z^4$$

for $d \in \mathbb{Q}(S)$. Then the Selmer group S^ϕ is

$$S^\phi(E/\mathbb{Q}) = \{d \in \mathbb{Q}(S) : C_d \text{ has a point defined over the completion } \mathbb{Q}_p \text{ for all } p \in S\},$$

and $\text{III}(E/\mathbb{Q})[\phi]$ is $S^\phi(E/\mathbb{Q})$ modulo the d 's for which C_d has a rational point.

In our case we will see that

$$(24) \quad \text{III}(E/\mathbb{Q})[\phi] = \text{III}(E'/\mathbb{Q})[\hat{\phi}] = 0$$

(a fortuitous circumstance; one is not always so lucky) and so (22), (22') yield the isomorphisms

$$(25) \quad E'(\mathbb{Q})/\phi(E(\mathbb{Q})) \cong S^\phi(E/\mathbb{Q})$$

$$(25') \quad E(\mathbb{Q})/\hat{\phi}(E'(\mathbb{Q})) \cong S^{\hat{\phi}}(E'/\mathbb{Q}).$$

With (21), (25), and (25'), plus some "torsion work," we can then determine $E(\mathbb{Q})$.

Finally the map δ in (22) is given by

$$\delta(O') = 1$$

$$\delta(0, 0) = a^2 - 4b$$

$$\delta(X, Y) = X \text{ if } X \neq 0, \infty$$

(all mod Q^2), where a, b define E and $(X, Y) \in E'(\mathbb{Q})$. The analogous map $\hat{\delta}$ in (22') is

$$\hat{\delta}(O) = 1,$$

$$(26) \quad \hat{\delta}(0, 0) = 16b \equiv b \pmod{Q^2}$$

$$\hat{\delta}(X, Y) = X \text{ if } X \neq 0, \infty, (X, Y) \in E(\mathbb{Q}),$$

since $(2a)^2 - 4(a^2 - 4b) = 16b$.

We first compute $S^\phi(E/Q)$.

LEMMA 10. *We have*

$$S^\phi(E/Q) = \{1, -7\} \quad \text{and} \quad \text{III}(E/\mathbb{Q})[\phi] = 0.$$

Proof. We must examine the curves in (23), namely

$$(27) \quad C_d: dw^2 = d^2 + 22dz^2 - 7z^4, \quad d \in \mathbb{Q}(S),$$

for solvability in $\mathbb{Q}_p$ and $\mathbb{Q}$.

$d = 1$: $w^2 = 1 + 22z^2 - 7z^4$ has the rational solution $z = 0, w = 1$;

$d = -7$: $-7w^2 = 49 - 154z^2 - 7z^4$ is equivalent to $w^2 = -7 + 22z^2 + z^4$, which has the rational solution $z = 1, w = 4$.

To prove the lemma, it suffices to show $-1, \pm 2 \notin S^\phi(E/\mathbb{Q})$, since $S^\phi(E/\mathbb{Q})$ is a group, and since $d = 1, -7$ are in the trivial class in $\text{III}(E/\mathbb{Q})[\phi]$.

$d = -1$: We show

$$-w^2 = 1 - 22z^2 - 7z^4$$

has no solutions in $\mathbb{Q}_2$. Write $z = B/A, w = C/A$, where $A, B, C \in \mathbb{Z}_2$, the 2-adic integers, $A \neq 0$ and $(A, B, C) = 1$ in $\mathbb{Z}_2$ (recall that 2 is the only prime in $\mathbb{Z}_2$). Then

$$(28) \quad \begin{aligned} C^2 A^2 &= -A^4 + 22A^2 B^2 + 7B^4, \\ C^2 A^2 &= -(A^2 - 11B^2)^2 + 128B^4, \end{aligned}$$

or

$$C_1^2 = -(A^2 - 11B^2)^2 + 128B^4,$$

with $C_1 = CA$. This equation implies

$$C_1 \equiv A^2 - 11B^2 \equiv 0 \quad \text{and} \quad A \equiv B \pmod{2}.$$

If A, B are even, then we can replace A, B, C_1 by $A/2, B/2, C_1/4$. Repeating this argument shows we may assume $A \equiv B \equiv 1 \pmod{2}$ in (28). But then $A^2 \equiv B^2 \equiv 1 \pmod{8}$ and (28) gives

$$C_1^2 \equiv -1 + 22 + 7 \equiv 28 \equiv -4 \pmod{16},$$

an impossible congruence. Thus $-1 \notin S^\phi(E/\mathbb{Q})$ and consequently $7 \notin S^\phi(E/\mathbb{Q})$.

$d = 2$: In this case (27) is

$$2w^2 = 4 + 44z^2 - 7z^4.$$

We claim this is impossible in $\mathbb{Q}_2$. As above, write $z = B/A, w = C/A$, giving

$$(29) \quad 2C^2A^2 = 4A^4 + 44A^2B^2 - 7B^4 \quad \text{in } \mathbb{Z}_2, \quad (A, B, C) = 1, \quad A \neq 0.$$

Clearly $2 \mid B$. If A were odd we would have

$$C^2A^2 \equiv 2A^4 \pmod{8},$$

or

$$\left(\frac{C}{A}\right)^2 \equiv 2 \pmod{8},$$

which is impossible. Hence $2 \mid A$ and $(A, B) = 2^u$, where $u \geq 1$. Since 2^{4u} divides the right side of (29) we have

$$2^{4u} \mid 2C^2A^2 = 2^{2u+1}C^2\left(\frac{A}{2^u}\right)^2,$$

and so $2^{2u-1} \mid (A/2^u)^2$, whence $2^{2u} \mid (A/2^u)^2$ and $2^{2u} \mid A$. The left side of (29) is then divisible by 2^{4u+1} , and so (29) implies $2^{4u+1} \mid 7B^4$, whence $2^{u+1} \mid B$. Since $2u \geq u+1$ this shows $2^{u+1} \mid (A, B) = 2^u$, a contradiction.

Thus $2 \notin S^\phi(E/\mathbb{Q})$, and by the same argument $-2 \notin S^\phi(E/\mathbb{Q})$. It follows that $\pm 14 \notin S^\phi(E/\mathbb{Q})$ and the lemma is proved.

LEMMA 11. *We have*

$$S^{\hat{\phi}}(E'/\mathbb{Q}) = \{1, 2\} \quad \text{and} \quad \text{III}(E'/\mathbb{Q})[\hat{\phi}] = 0.$$

Proof. We note $1, 2 \in S^{\hat{\phi}}(E'/\mathbb{Q})$ by (22'), since

$$(30) \quad \hat{\delta}(O) = 1, \quad \hat{\delta}(0, 0) = 32 \equiv 2 \pmod{\mathbb{Q}^2}$$

from (26) (recall $a = -11, b = 32$). If we show that

$$S^{\hat{\phi}}(E'/\mathbb{Q}) = \{1, 2\},$$

then the map $\hat{\delta}$ in (22') is onto, whence $\text{III}(E'/\mathbb{Q})[\hat{\phi}] = 0$. Since we are computing $S^{\hat{\phi}}$ for the curve E' we replace a, b in (23) by the corresponding values in (20) for E' . Thus we are led to consider the curves

$$C_d: dw^2 = d^2 - 44dz^2 + 512z^4, \quad d \in \mathbb{Q}(S).$$

The curve C_d clearly has no real solutions if $d < 0$, and so we have only to show that $7 \notin S^\phi$. The argument is much like the argument for $d = 2$ in Lemma 9. We show that

$$7w^2 = 49 - 7.44z^2 + 512z^4$$

has no solutions in $\mathbb{Q}_7$. Writing $z = B/A$, $w = C/A$ with $A, B, C \in \mathbb{Z}_7$ and $(A, B, C) = 1$ gives

$$(31) \quad 7C^2A^2 = 49A^4 - 7.44A^2B^2 + 512B^4.$$

Clearly $7 \mid B$ and therefore $7 \mid A$ (otherwise $C^2A^2 \equiv 7A^4 \pmod{7^2}$, which is clearly impossible). Setting $(A, B) = 7^u$ we see that $7^{4u} \mid 7C^2A^2$, whence $7^{2u-1} \mid (A/7^u)^2$ and $7^{2u} \mid A$. Then (31) implies $7^{4u+1} \mid B^4$ and $7^{u+1} \mid B$, a contradiction.

From Lemmas 9 and 10 and (25), (25') we deduce

$$E'(\mathbb{Q})/\phi(E(\mathbb{Q})) \cong E(\mathbb{Q})/\hat{\phi}(E'(\mathbb{Q})) \cong \mathbb{Z}/2\mathbb{Z}.$$

In fact $(0, 0)$ generates $E'(\mathbb{Q})/\phi(E(\mathbb{Q}))$ by (30). Now (21) and $\hat{\phi}(0, 0) = O$ show that

$$(32) \quad E(\mathbb{Q})/2E(\mathbb{Q}) \cong E(\mathbb{Q})/\hat{\phi}(E'(\mathbb{Q})) \cong \mathbb{Z}/2\mathbb{Z}.$$

Since $E(\mathbb{Q})$ has exactly one point of order 2, it follows from (32) that $E(\mathbb{Q})$ has no points of infinite order, i.e., its rank is 0. Thus $E(\mathbb{Q})$ is a torsion group, which may be calculated using the following theorem of Lutz and Nagell [17, p. 221].

THEOREM. *Let $E/\mathbb{Q}$ be an elliptic curve with equation*

$$y^2 = x^3 + Ax + B, \quad A, B \in \mathbb{Z}.$$

If $P = (x, y) \in E(\mathbb{Q})$ is a nonzero torsion point, then $x, y \in \mathbb{Z}$ and either $y = 0$ or $y^2 \mid 4A^3 + 27B^2$.

To apply the Lutz–Nagell theorem, we write (18) in the form

$$(3^3y)^2 = (9x - 33)^3 - 25 \cdot 27(9x - 33) + 27 \cdot 11 \cdot 46.$$

The above theorem implies that if $P \in E(\mathbb{Q})$ then we have $P = (0, 0)$ or $P = (x, y)$ with

$$(3^3y)^2 \mid 4(-25 \cdot 27)^3 + 27(27 \cdot 11 \cdot 46)^2,$$

i.e.,

$$(3^3y)^2 \mid 2^{10} \cdot 3^{12} \cdot 7.$$

Hence $3^3y \mid 2^5 \cdot 3^6$, and any integer solution of (18) satisfies $y \mid 2^5 \cdot 3^3$. Hence any integer solution to (6) satisfies $4a \mid 2^5 \cdot 3^3$, or $a \mid 2^3 \cdot 3^3$. Further $k \mid a^2 = 2^6 \cdot 3^6$. From (6) it is also clear that $k = t^2$ or $2t^2$. This gives 28 values to check for k and yields only the solutions

$$(33) \quad (k, a) = (1, \pm 1), (2, \pm 2).$$

Checking for all $y \mid 2^5 \cdot 3^6$ on the curve

$$y^2 = x^3 - 25 \cdot 27x + 27 \cdot 11 \cdot 46$$

also shows that the points

$$(4, \pm 4) \quad \text{and} \quad (8, \pm 8)$$

are the only rational points on E other than O and $(0, 0)$. Hence $E(\mathbb{Q}) \cong \mathbb{Z}/6\mathbb{Z}$. We have proved the following theorem.

THEOREM 12. *The only integer solutions (k, a) of*

$$a^2 = k(4k^2 - 11k + 8)$$

are $(k, a) = (0, 0), (1, \pm 1), (2, \pm 2)$.

At this point we have eliminated quasi-symmetric 3-designs with $p = 1$ and $q \geq 5$. In the next two sections, we show that $p = 1$, $q = 4$ is impossible by finding all integer points (b, k) on the curve (7).

4. Finding all solutions to $b^2 = 2k(k-1)(k-2) + 1$. Writing (7) in the form

$$(2b)^2 = (2k-2)^3 - 4(2k-2) + 4$$

leads us to consider the integer points (x, y) on

$$(34) \quad E_1: y^2 = x^3 - 4x + 4.$$

From tables in [2, p. 92, conductor 88] we know that E_1 has rank 1, and so the determination of the integer points on (7) is more difficult.

We follow Bremner and Tzanakis [3] and write (34) as

$$(35) \quad \begin{aligned} y^2 &= \text{Norm}(x - \theta) \\ &= (x - \theta)(x^2 + \theta x + \theta^2 - 4) \end{aligned}$$

where

$$\theta^3 - 4\theta + 4 = 0.$$

Since all points (k, b) on (7) come from integer points on (34) with

$$(x, y) = (2k - 2, 2b)$$

it suffices to consider even x and y in (34). To take (35) further, we need the standard facts about the field $K = \mathbb{Q}(\theta)$: K has discriminant -44 , class number 1 and the integral basis $\{1, \theta, (\theta^2/2)\}$. Noting that

$$\theta^3 = 4(\theta - 1)$$

where $\theta - 1$ is a unit, shows that $(2) = \mathfrak{p}_2^3$ in K . The only other ramified prime is 11, with $(11) = \mathfrak{p}_{11}\mathfrak{q}_{11}^2$. Finally, the fundamental unit in K is $\varepsilon = 1 - \theta^2/2$. Actually, all we need about ε is that it is not an even power of a unit: this follows easily from equating coefficients of θ^2 in

$$\begin{aligned} \pm \left(1 - \frac{\theta^2}{2}\right) &= \left(a + b\theta + c\frac{\theta^2}{2}\right)^2, \quad (a, b, c \in \mathbb{Z}), \\ &= a^2 - 4bc + (2ab + 4bc - c^2)\theta + (b^2 + c^2 + ac)\theta^2. \end{aligned}$$

Now consider the terms $A = x - \theta$ and $B = x^2 + \theta x + \theta^2 - 4$ on the right side of (35). Their greatest common divisor divides

$$B - (x + 2\theta)A = 3\theta^2 - 4.$$

Since $\text{Norm}(3\theta^2 - 4) = \text{Disc}(x^3 - 4x + 4) = -176$, $(3\theta^2 - 4) = \mathfrak{p}_2^4\mathfrak{q}_{11}$. Using the assumption that x is even, and $(\theta) = \mathfrak{p}_2^2$, unique factorization applied to (35) yields

$$(x - \theta) = \mathfrak{p}_2^2 a^2 \quad \text{or} \quad \mathfrak{p}_2^2 \mathfrak{q}_{11} a^2$$

for some integral ideal $\mathfrak{a}$ of K . The field K has class number 1, so this factorization can be written as an equation in K :

$$(36A) \quad x - \theta = \begin{cases} \pm \varepsilon^r \theta \left(a + b\theta + c\frac{\theta^2}{2}\right)^2 \\ \pm \varepsilon^r (4 + \theta) \left(a + b\theta + c\frac{\theta^2}{2}\right)^2 \end{cases}$$

$$(36B)$$

for $r = 0$ or 1 and $a, b, c \in \mathbb{Z}$. (Note that $\text{Norm}(4 + \theta) = -44$, so $(4 + \theta) = \mathfrak{p}_2^2 \mathfrak{q}_{11}$; here $\mathfrak{q}_{11} = (11, 4 + \theta)$, since

$$x^3 - 4x + 4 \equiv (x+3)(x+4)^2 \pmod{11}.$$

Equation (36B) is easily dispensed with. Taking norms of both sides gives $y^2 = \text{Norm}(x - \theta) = \pm 44z^2$, which is clearly impossible. Hence it suffices to treat (36A). We have two cases to consider.

Case A_0 : $r = 0$ in (36A). In this case (36A) becomes

$$\pm(x - \theta) = -4(b^2 + c^2 + ac) + (a^2 + 4b^2 + 4c^2 + 4ac - 4bc)\theta + (2ab + 4bc - c^2)\theta^2,$$

so equating coefficients gives

$$\begin{aligned} \pm x &= -4(b^2 + c^2 + ac), \\ (37) \quad \mp 1 &= a^2 + 4b^2 + 4c^2 + 4ac - 4bc = (a + 2c)^2 + (2b - c)^2 - c^2, \\ 0 &= c^2 - 4bc - 2ab. \end{aligned}$$

The last equation shows $2 \mid c$, and the middle equation gives $\mp 1 \equiv a^2 \pmod{4}$, implying that $2 \nmid a$ and that the lower sign prevails. Hence $1 \equiv 1 + 4b^2 \pmod{8}$ which implies that $2 \mid b$. Considering the last equation as quadratic in c gives

$$(38) \quad 16b^2 + 8ab = m^2 \quad \text{and} \quad c = 2b \pm \frac{m}{2}.$$

The first equation in (38) is now quadratic in b so that

$$64a^2 + 64m^2 = n^2 \quad \text{and} \quad b = \frac{-8a \pm n}{32},$$

or

$$a^2 + m^2 = \left(\frac{n}{8}\right)^2.$$

It is easy to see that $(a, m, n/8) = 1$, and so

$$a = u^2 - v^2$$

$$m = 2uv$$

$$\frac{n}{8} = u^2 + v^2$$

for $u, v \in \mathbb{Z}$. By changing the sign of a we may assume that u is even. For an appropriate choice of sign for v we then find

$$a = u^2 - v^2, \quad b = -\frac{1}{2}u^2, \quad c = -u^2 - uv,$$

and plugging these values into (37) gives

$$\begin{aligned} 1 &= (u+v)^4 + u^2v^2 - u^2(u+v)^2 \\ &= (u+v)^2(2u+v)v + u^2v^2, \end{aligned}$$

whence $v = \pm 1$. If $v = 1$, then $u = 0, -1, -2$, and keeping even values of u we get, respectively,

$$u = 0, \quad v = 1, \quad a = -1, \quad b = 0, \quad c = 0, \quad x = 0$$

and

$$u = -2, \quad v = 1, \quad a = 3, \quad b = -2, \quad c = -2, \quad x = 8,$$

corresponding to the points $(0, \pm 2)$ and $(8, \pm 22)$ on (34). The value $v = -1$ gives $u = 0, 1, 2$ and the same values for x .

Case A_1 : $r = 1$ in (36A). Here we consider

$$\pm(x - \theta) = \varepsilon \theta \alpha^2, \quad \alpha \in \mathbb{Z} \left[1, \theta, \frac{\theta^2}{2} \right].$$

Noting $\theta^4/4 = -\theta\varepsilon^2$ allows us to rewrite this equation in the form

$$\pm(x - \theta) = -\varepsilon \alpha_1^2 = -\left(1 - \frac{\theta^2}{2}\right) \left(a + b\theta + c\frac{\theta^2}{2}\right)^2,$$

and so

$$\begin{aligned} \mp x &= a^2 - 2c^2 + 4ab + 4bc, \\ (39) \quad \pm 1 &= 2b^2 + 3c^2 - 2ab + 2ac - 4bc, \\ 0 &= a^2 + 2b^2 + 2c^2 + 2ac - 4bc = a^2 + 2ac + 2(b - c)^2. \end{aligned}$$

Thus $2 \mid a, c \equiv b \equiv 1 \pmod{2}$ and

$$\pm 1 \equiv 2b^2 + 3c^2 - 4bc \equiv 1 \pmod{8}$$

forces us to take the upper sign.

Considering the last equation in (39) as quadratic in a , we have

$$4c^2 - 8(b - c)^2 = m^2, \quad a = -c \pm \frac{m}{2},$$

or

$$c^2 - 2(b - c)^2 = \left(\frac{m}{2}\right)^2.$$

Factoring this equation in the unique factorization domain $\mathbb{Z}[\sqrt{2}]$, and using the fact that a unit of Norm 1 in $\mathbb{Z}[\sqrt{2}]$ is an even power of the fundamental unit $1 + \sqrt{2}$, we see that

$$c + (b - c)\sqrt{2} = \pm(u + v\sqrt{2})^2,$$

whence

$$\begin{aligned} c &= \pm(u^2 + 2v^2), \\ b &= \pm(u^2 + 2uv + 2v^2), \\ \frac{m}{2} &= u^2 - 2v^2, \end{aligned}$$

where $u, v \in \mathbb{Z}$ and u is odd. Without loss of generality $c > 0$, and so we have

$$\begin{aligned} c &= u^2 + 2v^2, \\ (40) \quad b &= u^2 + 2uv + 2v^2, \\ a &= -2u^2 \quad \text{or} \quad -4v^2. \end{aligned}$$

This gives us two cases to check.

If $a = -2u^2$, the middle equation in (39) becomes

$$\begin{aligned} 1 &= u^4 + 8u^3v + 12u^2v^2 + 4v^4 \\ &= (u+2v)^4 - 12(u+2v)^2v^2 + 16(u+2v)v^3 + 4v^4, \end{aligned}$$

or

$$(41) \quad 1 = \text{Norm}(u+2v-v\phi_1),$$

where ϕ_1 satisfies the equation

$$(41') \quad f(\phi_1) = \phi_1^4 - 12\phi_1^2 + 16\phi_1 + 4 = 0.$$

If $a = -4v^2$, we find from (39) and (40) that

$$1 = u^4 + 12u^2v^2 + 16uv^3 + 4v^4,$$

or

$$(42) \quad 1 = \text{Norm}(u-v\phi_2),$$

where ϕ_2 satisfies

$$(42') \quad g(\phi_2) = \phi_2^4 + 12\phi_2^2 + 16\phi_2 + 4 = 0.$$

Thus finding all solutions in Case A_1 amounts to finding all solutions u, v in $\mathbb{Z}$ of the unit equations (41) and (42) in the quartic fields $K_1 = \mathbb{Q}(\phi_1)$ and $K_2 = \mathbb{Q}(\phi_2)$. Since

$$4g(x) = x^4 f\left(\frac{2}{x} + 2\right),$$

it is clear that K_1 and K_2 are isomorphic fields. However, we will solve (41) and (42) by treating the fields as if they were distinct.

5. The unit equations. Equations (41) and (42) require us to find all the units of the form

$$u+2v-v\phi_1, \quad \text{respectively,} \quad u-v\phi_2$$

in the fields $K_1 = \mathbb{Q}(\phi_1)$ and $K_2 = \mathbb{Q}(\phi_2)$. First we set

$$(43) \quad \eta = \frac{1}{4}(\phi_1^2 - 6), \quad \lambda = \frac{1}{4}(\phi_2^2 + 6);$$

using (41') and (42') it is easy to see that η and λ satisfy

$$\eta^4 - 4\eta^2 - 4\eta - 2 = 0, \quad \lambda^4 - 4\lambda^2 - 4\lambda + 10 = 0.$$

These equations have discriminant $-2^8 \cdot 11$, and thus $K_1 = \mathbb{Q}(\eta)$ and $K_2 = \mathbb{Q}(\lambda)$ have two real and two complex conjugates. Since η and λ satisfy Eisenstein polynomials with respect to the prime 2, integral bases in K_1 and K_2 are the power bases generated by η and λ . From (43) we have

$$\phi_1 = 2 - \eta^2, \quad \phi_2 = 2 - \lambda^2,$$

and so the unit equations (41) and (42) become

$$(44') \quad 1 = \text{Norm}(u + v\eta^2),$$

$$(44'') \quad 1 = \text{Norm}(u - 2v + v\lambda^2).$$

In both cases we must find all units with missing linear and cubic terms. Since K_1 and K_2 have two fundamental units, this amounts to finding all solutions of

$$(45') \quad \pm(u + v\eta^2) = \varepsilon_1' \varepsilon_2^s,$$

$$(45'') \quad \pm(u - 2v + v\lambda^2) = \rho_1' \rho_2^s,$$

where $\{\varepsilon_1, \varepsilon_2\}$ and $\{\rho_1, \rho_2\}$ are pairs of fundamental units in K_1 and K_2 . (Note that K_1 and K_2 contain no roots of unity other than ± 1 , since they are nonnormal over $\mathbb{Q}$ and cannot contain the imaginary fields $\mathbb{Q}(\sqrt{-1})$ or $\mathbb{Q}(\sqrt{-3})$.)

We computed fundamental units in K_1 and K_2 using Berwick's method (see [1] or [3, Appendix]). Let the real conjugates of a number α in K_1 be α, α' , and its complex conjugates $\alpha'', \bar{\alpha}''$. If two units α_1, α_2 satisfy any two of the three conditions

$$(46) \quad \alpha > 1 \quad \text{and minimal,} \quad |\alpha'| < 1, \quad |\alpha''| < 1,$$

$$(46') \quad |\alpha| < 1, \quad \alpha' > 1 \quad \text{and minimal,} \quad |\alpha''| < 1,$$

$$(46'') \quad |\alpha| < 1, \quad |\alpha'| < 1, \quad |\alpha''| > 1 \quad \text{and minimal,}$$

then α_1 and α_2 are a pair of fundamental units for K_1 . Solving (46''), for example, amounts to finding a unit β with

$$|\beta''| > 1, \quad |\beta| < 1, \quad |\beta'| < 1,$$

and then showing that $\pm\beta$ are the only solutions α of (46'') with $|\alpha| \leq |\beta|$. This in turn amounts to solving a system of four linear inequalities in four unknowns. See [3].

Using Berwick's method, we showed that

$$\varepsilon_1 = \eta + 1 \quad \text{and} \quad \frac{\varepsilon_1}{\varepsilon_2} = \frac{\eta + 1}{2\eta + 3} = 5 + 7\eta + 6\eta^2 - 4\eta^3$$

satisfy (46) and (46'). Thus we may take the units

$$\varepsilon_1 = \eta + 1, \quad \varepsilon_2 = 2\eta + 3$$

as a fundamental pair for $K_1 = \mathbb{Q}(\eta)$. In the same way we computed a fundamental pair of units for K_2 to be

$$\rho_1 = 13 + 3\lambda - 3\lambda^2 - 2\lambda^3, \quad \rho_2 = 3 - \lambda^2,$$

where ρ_1 and ρ_2 satisfy (46) and (46''). It is not hard to check using (46) that $\rho_1 = \varepsilon_1$, and therefore that

$$(47) \quad \begin{aligned} \eta &= 12 + 3\lambda - 3\lambda^2 - 2\lambda^3, \\ \lambda &= -2 - \eta + \eta^2. \end{aligned}$$

These formulas give an explicit isomorphism between K_1 and K_2 .

With the fundamental units in hand, we have the solution

$$(48) \quad 1 = \varepsilon_1^0 \varepsilon_2^0 \quad \text{with} \quad u = 1, \quad v = 0$$

of (45'), and the solutions

$$(49) \quad \begin{aligned} 1 &= \rho_1^0 \rho_2^0 && \text{with} \quad u = 1, \quad v = 0, \\ 3 - \lambda^2 &= \rho_1^0 \rho_2^1 && \text{with} \quad u = 1, \quad v = -1, \\ -7 + 3\lambda^2 &= \rho_1^{-4} \rho_2^1 && \text{with} \quad u = -1, \quad v = 3, \end{aligned}$$

of (45''). Using (39) and (40), these correspond, respectively, to the points $(2, \pm 2)$, $(-2, \pm 2)$, $(6, \pm 14)$, and $(310, \pm 5458)$ on (34).

We shall show, using Skolem's method, that these are the only solutions of (45') and (45'').

The method involves finding a prime p which splits completely in K_1 , for which the only solutions of (45') or (45'') mod p correspond to global solutions, i.e., come from the solutions in (48) and (49). For $p = 103,587$ there are extraneous solutions mod p , and so we work with $p = 773$.

First consider (45'). Since $\eta + 1$ has Norm -1 , (44') shows that r must be even in (45'). Using MACSYMA we computed that

$$\begin{aligned} (\eta + 1)^{2 \cdot 386} &\equiv (1 + 2\eta + \eta^2)^{386} \equiv 1 + 773(84 - 235\eta + 209\eta^2 + 348\eta^3) \pmod{773^2} \\ (2\eta + 3)^{193} &\equiv 317 + 773(352 - 84\eta + 314\eta^2 - 91\eta^3) \pmod{773^2}, \end{aligned}$$

and that the only pair (r, s) with $0 \leq r < 386$, $0 \leq s < 193$ for which

$$(\eta + 1)^{2r}(2\eta + 3)^s$$

has both coefficients of η and η^3 divisible by 773 is $(r, s) = (0, 0)$. Since 386 is the order of $(\eta + 1)^2 \pmod{773}$ and 193 is the order of $2\eta + 3$ modulo the group

$$G = \{ \alpha \in \mathbb{Z}[\eta], (\alpha, 773) = 1 : \alpha \equiv n \pmod{773} \text{ with } n \in \mathbb{Z} \},$$

we may set $r = 386M$, $s = 193N$, and (45') becomes

$$\begin{aligned} u + v\eta^2 &= (1 + 2\eta + \eta^2)^{386M}(2\eta + 3)^{193N} \\ &= (1 + 773\xi_1)^M(317 + 773\xi_2)^N, \end{aligned}$$

or

$$(50) \quad (317)^{-N}(u + v\eta^2) = (1 + 773\xi_1)^M(1 + 773(317)^{-1}\xi_2)^N,$$

where

$$(51) \quad \begin{aligned} \xi_1 &\equiv 84 - 235\eta + 209\eta^2 + 348\eta^3, \\ \xi_2 &\equiv 352 - 84\eta + 314\eta^2 - 91\eta^3, \end{aligned} \pmod{773}.$$

We now work with (50) in the 773-adic field $\mathbb{Q}_{773}$. Since 773 splits into first degree prime ideals in K_1 , we may consider η as an element of $\mathbb{Q}_{773}$. In fact there are four distinct embeddings of $\mathbb{Q}(\eta)$ in $\mathbb{Q}_{773}$. We claim that the right-hand side of (50) can be expanded as a power series in M, N with 773-adic integral coefficients. One way to see this is as follows. Let exp and log denote the 773-adic exponential function and logarithm (see [8, p. 283]). Then

$$\begin{aligned} (1 + 773\xi_1)^M &= \exp(M \log(1 + 773\xi_1)) \\ &= \sum_{k=0}^{\infty} \frac{M^k}{k!} (\log(1 + 773\xi_1))^k. \end{aligned}$$

Now

$$\begin{aligned} \log(1 + 773\xi_1) &= \sum_{j=1}^{\infty} (773\xi_1)^j \frac{(-1)^{j-1}}{j} \\ &= A_0 + A_1\eta + A_2\eta^2 + A_3\eta^3 \end{aligned}$$

where the A_i are convergent series in $\mathbb{Q}_{773}$. Since the power of 773 dividing $773^j/j$ is ≥ 2 for $j \geq 2$ we also have

$$(52) \quad \log(1 + 773\xi_1) \equiv 773\xi_1 \pmod{773^2}.$$

Hence

$$\begin{aligned} (1 + 773\xi_1)^M &= \sum_{k=0}^{\infty} \frac{M^k}{k!} (A_0 + A_1\eta + A_2\eta^2 + A_3\eta^3)^k \\ &= B_0 + B_1\eta + B_2\eta^2 + B_3\eta^3, \end{aligned}$$

where each B_i is a convergent power series for $M \in \mathbb{Z}_{773}$ (the 773-adic integers). From (52) and the fact that $773^2 \mid (773^k/k!)$ for $k \geq 2$ (see [8, p. 65]) it follows that

$$(1 + 773\xi_1)^M \equiv 1 + 773M\xi_1 \pmod{773^2}.$$

Thus (50) gives

$$\begin{aligned} (53) \quad (317)^{-N}(u + v\eta^2) &= K_0 + K_1\eta + K_2\eta^2 + K_3\eta^3 \\ &\equiv 1 + 773M\xi_1 + 773N(317)^{-1}\xi_2 \pmod{773^2}, \end{aligned}$$

where each K_i is a power series in M and N with 773-adic integral coefficients.

It is easy to see that the series K_i in (53) are independent of the embedding $\eta \rightarrow \eta_j$ ($1 \leq j \leq 4$) into $\mathbb{Q}_{773}$. Since the η_j are distinct, (53) may be interpreted as the matrix equation

$$(54) \quad A \begin{pmatrix} * \\ 0 \\ * \\ 0 \end{pmatrix} = A \begin{pmatrix} K_0 \\ K_1 \\ K_2 \\ K_3 \end{pmatrix},$$

where

$$A = \begin{pmatrix} 1 & \eta_1 & \eta_1^2 & \eta_1^3 \\ \vdots & & & \vdots \\ 1 & \eta_4 & \eta_4^2 & \eta_4^3 \end{pmatrix}$$

has nonzero determinant. We conclude from (54) that $K_1 = K_3 = 0$, i.e., that

$$\begin{aligned} (55) \quad 0 &= -235M + (317)^{-1}(-84)N + 773(\quad) + \cdots \\ 0 &= 348M + (317)^{-1}(-91)N + 773(\quad) + \cdots \end{aligned}$$

But Skolem's theorem ([8, p. 228]), together with

$$\begin{vmatrix} -235 \cdot 317 & -84 \\ 348 \cdot 317 & -91 \end{vmatrix} \equiv 428 \pmod{773},$$

implies that (55) has a unique 773-adic solution (M, N) , which must be $(0, 0)$. Hence the only solution of (45') is $r = s = 0$, the solution in (48).

For (45'') we take $p = 773$ as well. Equation (44'') shows that r must be even in (45''), and

$$\rho_1^2 = 39 + 10\lambda - 9\lambda^2 - 6\lambda^3.$$

Now MACSYMA tells us that

$$\begin{aligned} (\rho_1^2 \rho_2)^{193} &= (27 + 6\lambda - 6\lambda^2 - 4\lambda^3)^{193} \\ &\equiv 317 + 773(154 + 103\lambda + 111\lambda^2 - 214\lambda^3)(\text{mod } 773^2) \end{aligned}$$

and

$$\begin{aligned} (\rho_2)^{386} &= (3 - \lambda^2)^{386} \\ &\equiv -1 + 773(-129 - 38\lambda - 125\lambda^2 + 384\lambda^3)(\text{mod } 773^2). \end{aligned}$$

Furthermore, the only pairs (r, s) with $-96 < r \leq 97$, $-193 < s \leq 193$, for which $(\rho_1^2 \rho_2)^r \rho_2^s$ has both coefficients of λ and λ^3 divisible by 773 are

$$(56) \quad \begin{aligned} (r, s) &= (0, 0), & (\rho_1^2 \rho_2)^r \rho_2^s &= 1, \\ (r, s) &= (0, 1), & (\rho_1^2 \rho_2)^r \rho_2^s &= 3 - \lambda^2, \\ (r, s) &= (-2, 3), & (\rho_1^2 \rho_2)^r \rho_2^s &= -7 + 3\lambda^2. \end{aligned}$$

Replacing (r, s) by $(r + 193M, s + 386N)$ and expanding in power series in $\mathbb{Q}_{773}$ as before we rewrite (45'') as

$$(57) \quad \begin{aligned} &(-1)^N 317^{-M} (u - 2v + v\lambda^2) \\ &= (\rho_1^2 \rho_2)^r \rho_2^s (1 + 773(317^{-1})\xi_1)^M (1 - 773\xi_2)^N \\ &= (\rho_1^2 \rho_2)^r \rho_2^s (1 + 773(317)^{-1}M\xi_1 - 773N\xi_2 + 773^2(\dots) + \dots), \end{aligned}$$

where

$$\begin{aligned} \xi_1 &\equiv 154 + 103\lambda + 111\lambda^2 - 214\lambda^3, \\ \xi_2 &\equiv -129 - 38\lambda - 125\lambda^2 + 384\lambda^3 \pmod{773}. \end{aligned}$$

Using (56) we have, respectively,

$$(r, s) = (0, 0):$$

$$\begin{aligned} (-1)^N (317)^{-M} (u - 2v + v\lambda^2) &= 1 + 773(317)^{-1}M(154 + 103\lambda + 111\lambda^2 - 214\lambda^3) \\ &\quad + 773N(129 + 38\lambda + 125\lambda^2 - 384\lambda^3) + \dots \end{aligned}$$

$$(r, s) = (0, 1):$$

$$\begin{aligned} (-1)^N (317)^{-M} (u - 2v + v\lambda^2) &= 3 - \lambda^2 + 773(317)^{-1}M(1572 - 2275\lambda + 591\lambda^2 + 111\lambda^3) \\ &\quad + 773N(1637 - 4226\lambda + 1282\lambda^2 + 346\lambda^3) + \dots \end{aligned}$$

$$(r, s) = (-2, 3):$$

$$\begin{aligned} (-1)^N (317)^{-M} (u - 2v + v\lambda^2) &= -7 + 3\lambda^2 + 773(317)^{-1}M(-4408 + 7031\lambda - 1551\lambda^2 - 761\lambda^3) \\ &\quad + 773N(-4653 + 12754\lambda - 3596\lambda^2 - 1806\lambda^3) + \dots \end{aligned}$$

Each of these equations leads to two equations analogous to (55), obtained by setting coefficients of λ and λ^3 equal to zero. The corresponding determinants are

$$(317)^{-1} \begin{vmatrix} 103 & 38 \\ -214 & -384 \end{vmatrix} \equiv 35, \quad (317)^{-1} \begin{vmatrix} -2275 & -4226 \\ 111 & 346 \end{vmatrix} \equiv 33,$$

and

$$(317)^{-1} \begin{vmatrix} 7031 & 12754 \\ -761 & -1806 \end{vmatrix} \equiv -246 \pmod{773};$$

by Skolem's theorem there is exactly one solution in each case, which must be $(M, N) = (0, 0)$. This proves that all solutions of (45'') are given by (49).

Therefore the only solutions of (34) with x even are

$$(0, \pm 2), \quad (8, \pm 22), \quad (2, \pm 2), \quad (-2, \pm 2), \quad (6, \pm 14), \quad \text{and} \quad (310, \pm 5458).$$

Using the transformation

$$(k, b) = \left(\frac{x+2}{2}, \frac{y}{2} \right)$$

we finally have the following conclusion.

THEOREM 13. *The only integer points on the curve*

$$b^2 = 2k(k-1)(k-2) + 1$$

are $(k, b) = (1, \pm 1), (5, \pm 11), (2, \pm 1), (0, \pm 1), (4, \pm 7),$ and $(156, \pm 2729)$.

The divisibility condition $3 \mid k-1$ implies that the integer point $(156, \pm 2729)$ does not arise from a quasi-symmetric 3-design. By Lemma 8 we have now eliminated quasi-symmetric 3-designs with $p = 1$ and $q = 4$. This completes the proof of the Sane-Shrikhande conjecture.

Acknowledgment. We would like to thank Andrew Bremner for pointing out that the rank of the curve $y^2 = x^3 - 11x^2 + 32x$ is 0, and for providing the details of case $d = -1$ in Lemma 10.

REFERENCES

- [1] W. E. H. BERWICK, *Algebraic number fields with two independent units*, Proc. London Math. Soc., 34 (1932), pp. 360–378.
- [2] B. J. BIRCH AND W. KUYK, EDS., *Modular functions of one variable IV*, Proc. of International Summer School at Antwerp, 1972, Springer Lecture Notes 476, New York, 1975.
- [3] A. BREMNER AND N. TZANAKIS, *Integer points on $y^2 = x^3 - 7x + 10$* , Math. Comp., 41 (1983), pp. 731–741.
- [4] A. R. CALDERBANK, *Symmetric designs as the solution of an extremal problem in combinatorial set theory*, Europ. J. Combinatorics, 8 (1987), pp. 171–173.
- [5] ———, *Inequalities for quasi-symmetric designs*, J. Combin. Theory Ser. A, 48 (1988), pp. 53–64.
- [6] P. J. CAMERON, *Extending symmetric designs*, J. Combin. Theory Ser. A, 14 (1973), pp. 215–220.
- [7] P. J. CAMERON AND J. H. VAN LINT, *Graph theory, coding theory, and block designs*, London Math. Soc. Lecture Notes Series 43, Cambridge University Press, 1980.
- [8] J. W. S. CASSELS, *Local fields*, London Math. Soc. Student Texts 3, Cambridge University Press, Cambridge, 1986.
- [9] P. FRANKL AND Z. FÜREDI, *Finite projective spaces and intersecting hypergraphs*, Combinatorica, 6 (1986), pp. 335–354.

- [10] J.-M. GOETHALS AND J. J. SEIDEL, *Strongly regular graphs derived from combinatorial designs*, *Canad. J. Math.*, 22 (1970), pp. 597–614.
- [11] C. W. H. LAM, L. THIEL, S. SWIERCZ, AND J. MCKAY, *The nonexistence of ovals in a projective plane of order 10 (communication)*, *Discrete Math.*, 45 (1983), pp. 319–321.
- [12] W. NARKIEWICZ, *Elementary and Analytic Theory of Numbers*, PWN-Polish Scientific Publishers, Warszawa, 1973.
- [13] A. NEUMAIER, *New inequalities for the parameters of an association scheme*, in *Combinatorics and Graph Theory*, *Lecture Notes in Mathematics* 885, Springer-Verlag, Berlin, 1981, pp. 365–367.
- [14] ———, *Regular sets and quasi-symmetric 2-designs*, in *Combinatorial Theory*, D. Jungnickel and K. Vedder, eds., *Lecture Notes in Mathematics* 969, Springer-Verlag, Berlin, 1982, pp. 258–275.
- [15] S. S. SANE AND M. S. SHRIKHANDE, *Quasi-symmetric 2, 3, 4-designs*, *Combinatorica*, 7 (1987), pp. 291–301.
- [16] M. S. SHRIKHANDE, *A survey of some problems in combinatorial designs—a matrix approach*, *Linear Algebra Appl.*, 79 (1986), pp. 215–247.
- [17] J. H. SILVERMAN, *The Arithmetic of Elliptic Curves*, *Springer Graduate Texts in Mathematics* 106, Springer-Verlag, New York, 1986.

MAXIMAL CHAINS AND ANTICHAINS IN BOOLEAN LATTICES*

D. DUFFUS†, B. SANDS‡, AND P. WINKLER†

Abstract. The following equivalent results in the Boolean lattice 2^n are proven.

- (a) Every fibre of 2^n contains a maximal chain.
- (b) Every cutset of 2^n contains a maximal antichain.
- (c) Every red-blue colouring of the vertices of 2^n produces either a red maximal chain or a blue maximal antichain.

- (d) Given any n antichains in 2^n there is a disjoint maximal antichain.

Statement (a) is then improved to:

- (a') Every fibre of 2^n contains at least $n!/2^{n-1}$ maximal chains.

One conjecture of Long and Rival is supported, and another conjecture disproved, by showing:

- (i) Every fibre of 2^n has order $\Omega(1.25^n)$ elements.
- (ii) There is a minimal fibre of 2^n ($n \geq 4$) of size $2^{n-1} + 2$.

Key words. Boolean lattice, maximal chain, maximal antichain

AMS(MOS) subject classifications. 06E05, 06A10

1. Introduction. A *cutset* of a finite partially ordered set P is a subset of P that intersects all maximal chains, and a *fibre* of P is a subset intersecting all maximal antichains. The reader may consult [2] for more information on these concepts.

A simple example of a cutset is the set of all minimal elements (or maximal elements) of P . More generally, if all maximal chains of P have the same length, then the *levels* of P are cutsets. (Here the k th level of P is the set of all elements $x \in P$ such that all maximal chains through x contain exactly k elements less than x . Thus the 0th level is just the set of minimal elements of P .)

For fibres, there is also a natural example. The *cone* of an element $x \in P$ is the set of all elements comparable to x (i.e., either $\leq x$ or $\geq x$). It is a simple exercise to see that every cone is a fibre (e.g., see [2]).

Both of these constructions reinforce the intuitive idea of a cutset as something stretching “horizontally” through P and a fibre as something stretching “vertically” through P . In particular, every level contains (in fact, is) a *maximal* antichain, and every cone contains a *maximal* chain. This will not be the case for every cutset or fibre of every poset P ; for example in the poset of Fig. 1, $\{a, d\}$ is a cutset with no maximal antichain and $\{b, c\}$ a fibre with no maximal chain. The main result of this paper shows that intuition holds for one familiar family of finite posets: finite Boolean lattices.

We denote by 2^n the Boolean lattice with n atoms, that is, the lattice of all subsets of an n -element set.

THEOREM 1. (a) *Every fibre of 2^n contains a maximal chain.*

(b) *Every cutset of 2^n contains a maximal antichain.*

In fact these two statements are equivalent for any poset P ; it is easy to see that if F were a fibre of P containing no maximal chain, then $P - F$ would be a cutset of P containing no maximal antichain, and conversely. We may even note a third equivalent statement of Theorem 1:

* Received by the editors February 3, 1989; accepted for publication June 22, 1989.

† Department of Mathematics and Computer Science, Emory University, Atlanta, Georgia 30322. This research was supported by Office of Naval Research contract N00014-85-K-0769.

‡ Department of Mathematics and Statistics, The University of Calgary, 2500 University Drive North West, Calgary, Alberta, Canada T2N 1N4. This research was supported by Natural Sciences and Engineering Research Council of Canada grant 69-3378 and was conducted while the author was on sabbatical at Emory University.

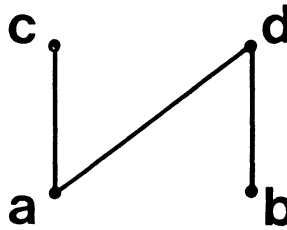


FIG. 1

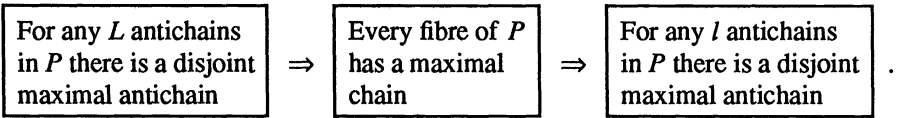
(c) *If the elements of 2^n are coloured red and blue, there is either a red maximal chain or a blue maximal antichain.*

The equivalence again follows for arbitrary posets P since, for example, if the red elements do not contain a maximal chain, then by (a) of Theorem 1 they cannot be a fibre, and therefore must be disjoint from some maximal antichain which is necessarily all blue. Conversely, assuming (c) and given a fibre F of P , colour the elements of F red and everything else blue; then there cannot be a blue maximal antichain, so by (c) F contains a maximal chain.

We give one more equivalent formulation of Theorem 1.

(d) *Given any n antichains in 2^n , there is a maximal antichain disjoint from all of them.*

Note that this is best possible, as 2^n is the union of its $n + 1$ levels. This time we do not get equivalence for arbitrary posets P . Let the shortest maximal chain in P have $l + 1$ elements and the longest have $L + 1$ elements; then we have the following implications:



For the first implication, if a fibre F of P does not have a maximal chain, then every chain of F has at most L elements. Thus F is the union of at most L antichains, so there is a maximal antichain disjoint from F , a contradiction. For the second implication, the union of l antichains in P cannot contain a maximal chain, so cannot be a fibre, so there is a disjoint maximal antichain.

Of course, when $P = 2^n$ we have $L = l = n$, whence (a) and (d) are equivalent.

In the next section we prove Theorem 1 in form (a), or actually a stronger version which will enable us to deduce the following corollary.

COROLLARY. *Every fibre of 2^n contains at least $n!/2^{n-1}$ maximal chains.*

In the final section we give some results bearing on two conjectures of Lonc and Rival [2] on the sizes of *minimal* fibres (i.e., fibres none of whose proper subsets is a fibre) of 2^n . In particular we prove in the following theorem that all fibres of 2^n are of size exponential in n .

THEOREM 2. *If $\mathcal{F}$ is a fibre of 2^n then $|\mathcal{F}| = \Omega(1.25^n)$.*

2. Proof of Theorem 1. We take the elements of 2^n to be all subsets of $[n] = \{1, 2, \dots, n\}$, and shall denote them by capitals. Subsets of 2^n will be denoted by script letters, so a fibre shall be denoted $\mathcal{F}$, for example. If $\mathcal{S} \subseteq 2^n$ we write

$$\mathcal{S}^- = \{X \in 2^n \mid X \subseteq S \text{ for some } S \in \mathcal{S}\}$$

and

$$\mathcal{S}^+ = \{X \in 2^n \mid X \supseteq S \text{ for some } S \in \mathcal{S}\}.$$

Let α be an arbitrary total ordering of the elements of $[n]$. For $X \in 2^n$ we let $X = \{x_1, x_2, \dots, x_k\}$ where $x_1 <_\alpha x_2 <_\alpha \dots <_\alpha x_k$ ($<_\alpha$ being determined by α), and we also put $[n] - X = \{y_1, y_2, \dots, y_{n-k}\}$ where $y_1 <_\alpha y_2 <_\alpha \dots <_\alpha y_{n-k}$. Define the α -lexical chain through X , denoted $\mathcal{L}_\alpha(X)$, to be the chain $\mathcal{L}_\alpha^-(X) \cup \mathcal{L}_\alpha^+(X)$ where

$$\mathcal{L}_\alpha^-(X) = \{X, X - \{x_1\}, X - \{x_1, x_2\}, \dots, X - \{x_1, x_2, \dots, x_k\} = \emptyset\},$$

$$\mathcal{L}_\alpha^+(X) = \{X, X \cup \{y_1\}, X \cup \{y_1, y_2\}, \dots, X \cup \{y_1, y_2, \dots, y_{n-k}\} = [n]\}.$$

That is, $\mathcal{L}_\alpha^-(X)$ is constructed by removing elements from X one at a time, smallest to largest according to α ; $\mathcal{L}_\alpha^+(X)$ is constructed by adding elements to X one at a time, smallest to largest according to α ; and $\mathcal{L}_\alpha(X)$ is just these two chains put together. Clearly $\mathcal{L}_\alpha(X)$ is a maximal chain of 2^n for each α and X . If $\mathcal{X}$ is a subset of 2^n , we write

$$\mathcal{L}_\alpha^-(\mathcal{X}) = \bigcup \{ \mathcal{L}_\alpha^-(X) : X \in \mathcal{X} \},$$

with similar definitions for $\mathcal{L}_\alpha^+(\mathcal{X})$ and $\mathcal{L}_\alpha(\mathcal{X})$. If α is the usual ordering $1 < 2 < \dots < n$, we drop the α 's, writing $\mathcal{L}(X)$ instead of $\mathcal{L}_\alpha(X)$, for instance.

We will prove that

(*) every fibre of 2^n contains the lexical chain $\mathcal{L}(X)$ for some $X \in 2^n$,

and in particular form (a) of Theorem 1 holds. To do this we need the following—somewhat awkward—definition. Call a subset $\mathcal{S}$ of a fibre $\mathcal{F}$ of 2^n *critical* if there do not exist subsets $\mathcal{A}, \mathcal{B}$ of 2^n satisfying

- (1) $\mathcal{A} \cup \mathcal{B}$ is an antichain disjoint from $\mathcal{F}$,
- (2) $\mathcal{S} \subseteq \mathcal{A}^- \cup \mathcal{B}^+$, and
- (3) $\mathcal{A} \subseteq \mathcal{L}^+(\mathcal{S})$, $\mathcal{B} \subseteq \mathcal{L}^-(\mathcal{S})$.

The idea behind this definition is as follows. Since $\mathcal{F}$ is a fibre, every antichain avoiding $\mathcal{F}$, when extended to a maximal antichain, must meet $\mathcal{F}$. A subset $\mathcal{S}$ of $\mathcal{F}$ for which there is no $\mathcal{A}$ and $\mathcal{B}$ satisfying (1) and (2) will by itself prevent any antichain outside $\mathcal{F}$ from being extended to a maximal antichain outside $\mathcal{F}$, in that any such maximal antichain must contain an element of $\mathcal{S}$. It can be assumed, without violating (1) or (2), that $\mathcal{A} \subseteq \mathcal{S}^+$ and $\mathcal{B} \subseteq \mathcal{S}^-$; condition (3), found necessary for our proof to work, requires further that each element of $\mathcal{A}$ be on the upper half of some lexical chain through an element of $\mathcal{S}$, and dually for $\mathcal{B}$.

Note that $\mathcal{F}$ itself is critical; even if $\mathcal{A}, \mathcal{B} \subseteq 2^n$ were to satisfy just (1) and (2) for $\mathcal{S} = \mathcal{F}$, $\mathcal{A} \cup \mathcal{B}$ could be extended to a maximal antichain which would necessarily avoid $\mathcal{F}$, a contradiction. On the other hand the empty set is trivially not critical, as is witnessed by the choice $\mathcal{A} = \mathcal{B} = \emptyset$.

The proof of (*) is by contradiction. Suppose $\mathcal{F}$ is a fibre of 2^n containing no lexical chain. By the above we may choose a nonempty critical subset $\mathcal{M}$ of $\mathcal{F}$ which is *minimal*, that is, no proper subset of $\mathcal{M}$ is critical. For each $X \in \mathcal{M}$ define the *rank* $r(X)$ of X to be the least positive integer r such that either $X \cup \{1, 2, \dots, r\}$ or $X - \{1, 2, \dots, r\}$ is not in $\mathcal{F}$. Note that $r(X)$ exists for each X since the entire lexical chain $\mathcal{L}(X)$ is never in $\mathcal{F}$. Choose $M \in \mathcal{M}$ such that $r(M) \leq r(X)$ for all $X \in \mathcal{M}$.

Since $\mathcal{M}$ is minimally critical, $\mathcal{M} - \{M\}$ cannot be critical, so there exist subsets $\mathcal{A}, \mathcal{B}$ of 2^n satisfying (1), (2), and (3) for $\mathcal{S} = \mathcal{M} - \{M\}$. Since $\mathcal{M}$ is critical, $\mathcal{A}$ and

$\mathcal{B}$ must fail one of (1)–(3) for $\mathcal{S} = \mathcal{M}$. As is easy to see, the only possibility is that $M \notin \mathcal{A}^- \cup \mathcal{B}^+$.

By the definition of $r(M)$ either

(i) $M' = M \cup \{1, 2, \dots, r(M)\} \notin \mathcal{F}$

or

(ii) $M'' = M - \{1, 2, \dots, r(M)\} \notin \mathcal{F}$.

We may suppose (i), by a duality argument we now give. For a set $S \in 2^n$, let $\bar{S} = [n] - S$ be the complement. If $\mathcal{S}$ is a subset of 2^n we write

$$\bar{\mathcal{S}} = \{\bar{S} \mid S \in \mathcal{S}\}.$$

(Note carefully that $\bar{\mathcal{S}} \neq 2^n - \mathcal{S}$!) $\bar{\mathcal{S}}$ is the image of $\mathcal{S}$ under the dual automorphism $S \rightarrow \bar{S}$ of 2^n . Thus it is clear that if $\mathcal{S}$ is a maximal antichain (or maximal chain, or fibre), so is $\bar{\mathcal{S}}$. In particular, if $\mathcal{S}$ is a fibre with no maximal chain, so is $\bar{\mathcal{S}}$. Furthermore, if $X \in 2^n$ then

$$\overline{\mathcal{L}^-(X)} = \mathcal{L}^+(\bar{X}) \text{ and } \overline{\mathcal{L}^+(X)} = \mathcal{L}^-(\bar{X}),$$

and so

$$\overline{\mathcal{L}(\bar{X})} = \mathcal{L}(X),$$

all as *sets*; as chains they are dual to each other in each case. It now follows that a subset $\mathcal{S}$ of a fibre $\mathcal{F}$ is (minimally) critical if and only if $\bar{\mathcal{S}}$ is a (minimally) critical subset of $\bar{\mathcal{F}}$. Moreover if $\mathcal{F}$ is a fibre containing no lexical chain, then $\bar{\mathcal{F}}$ also contains no lexical chain, and for each $X \in \mathcal{F}$, $r(X) = r(\bar{X})$, where $r(\bar{X})$ denotes the rank of $\bar{X}$ in the fibre $\bar{\mathcal{F}}$. Thus, if (ii) holds for the element M of the fibre $\mathcal{F}$, then (i) will hold for the (equally minimal) element $\bar{M}$ of the fibre $\bar{\mathcal{F}}$. Hence we may assume (i) holds for the given element and fibre.

Now let

$$\mathcal{A}' = (\mathcal{A} - \{M'\}^-) \cup \{M'\}$$

(see Fig. 2 for a schematic).

We claim that the sets $\mathcal{A}'$, $\mathcal{B}$ satisfy (1), (2), and (3) for $\mathcal{S} = \mathcal{M}$. This is impossible since $\mathcal{M}$ is critical, and (*) follows.

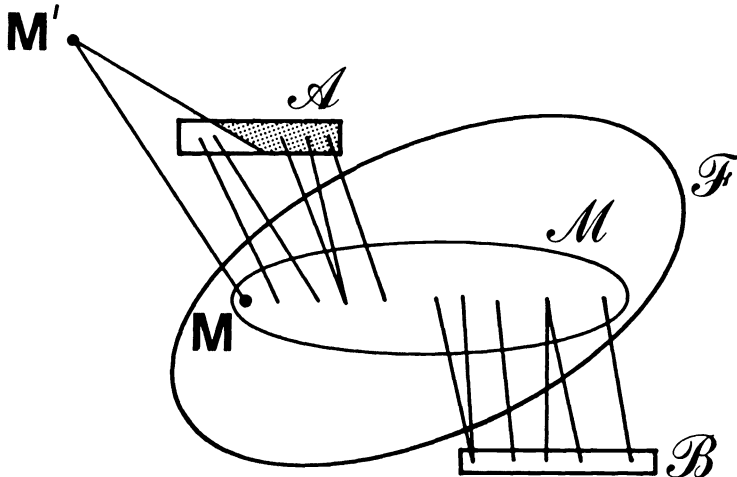


FIG. 2

First, (3) is trivially true since $\mathcal{B}$ has not changed, $\mathcal{S}$ has been enlarged (from $\mathcal{M} - \{M\}$ to $\mathcal{M}$) so $\mathcal{L}^-(\mathcal{S})$ and $\mathcal{L}^+(\mathcal{S})$ could only get bigger, and the only element of $\mathcal{A}'$ not in $\mathcal{A}$ is M' which is an element of $\mathcal{L}^+(M)$ and thus of $\mathcal{L}^+(\mathcal{M})$. (2) is immediate also; $\mathcal{B}^+$ is unchanged, and it is easy to see that $\mathcal{A}'^-$ contains $\mathcal{A}^-$, therefore we need only observe that $M \in \mathcal{A}'^-$ follows from $M \leq M'$ and $M' \in \mathcal{A}'$.

For (1), $\mathcal{A}' \cup \mathcal{B}$ is of course disjoint from $\mathcal{F}$ since $M' \notin \mathcal{F}$. Thus it remains to check that M' is incomparable to all elements of $\mathcal{A}' - \{M'\} = \mathcal{A} - \{M'\}^-$ and of $\mathcal{B}$. If $A \in \mathcal{A} - \{M'\}^-$, then by definition $A \not\leq M'$, and $M' \not\leq A$ since $M \notin \mathcal{A}'^-$. Finally let $B \in \mathcal{B}$. Then by (3) $B \in \mathcal{L}^-(X)$ for some $X \in \mathcal{M} - \{M\}$, and $r(X) \geq r(M)$ by the choice of M . Thus B cannot contain any of the elements $\{1, 2, \dots, r(M)\}$; if in X at all, they were the first to be deleted from X in forming the chain $\mathcal{L}^-(X)$. But since $M \notin \mathcal{B}^+$, $B \not\leq M$, and hence $B \not\leq M \cup \{1, 2, \dots, r(M)\} = M'$. Since $1 \in M' - B$, we also have $M' \not\leq B$, and we are done. $\square$

The reader should note the following examples before attempting to extend this theorem.

(1) If extreme points 0, 1 are added to the poset of Fig. 1, we obtain a poset isomorphic to the product of a two- and a three-element chain, with a fibre $\{0, b, c, 1\}$ which does not contain a maximal chain. Thus Theorem 1 cannot be extended to arbitrary finite products of finite chains.

(2) We now describe a minimal fibre $\mathcal{F}$ of 2^6 containing an element X but not containing any maximal chain through X . To simplify notation we will denote elements of 2^6 by strings of integers, writing 123 instead of $\{1, 2, 3\}$, for example. Let

$$\mathcal{A} = \{12, 13, 14, 25, 26, 156, 234, 345, 346, 356, 456\}.$$

It can easily be checked that $\mathcal{A}$ is a maximal antichain of 2^6 (first check that no two- or three-element subset of $\{1, 2, \dots, 6\}$ can be added; it then follows that no other subset can be added either). However there is *no* maximal antichain of 2^6 contained in

$$\mathcal{B} = (\mathcal{A} \cup \{123, 124, 125, 126\}) - \{12\}.$$

(To avoid our adding 12 to such an antichain, it would have to contain, say, 123; this means that 13 cannot be in the antichain, and now nothing can prevent 135 from being included in the antichain.) This means that $2^6 - \mathcal{B}$ is a fibre containing 12 but no upper cover of 12, and hence no maximal chain through 12. Let $\mathcal{F}$ be any minimal fibre contained in $2^6 - \mathcal{B}$; then $X = 12 \in \mathcal{F}$, since otherwise the maximal antichain $\mathcal{A}$ misses $\mathcal{F}$.

(3) We modify a construction of Nowakowski [3] to find a minimal cutset $\mathcal{C}$ of 2^n containing an element X but no maximal antichain containing X . Using a notation analogous to that of the previous example, for $n \geq 4$ let

$$\mathcal{C} = \{12k: 3 \leq k \leq n\} \cup \{1k: 3 \leq k \leq n\} \cup \{2k: 3 \leq k \leq n\} \cup \{k: 3 \leq k \leq n\}.$$

It is easy to see that $\mathcal{C}$ is a minimal cutset of 2^n (consider the smallest element of a given maximal chain of 2^n which is not contained in 12). We claim that, for instance, $\mathcal{C}$ contains no maximal antichain containing 13. For if $\mathcal{A}$ were such an antichain, then for each $k \geq 4$, to avoid adding $3k$ to $\mathcal{A}$ we must have $k \in \mathcal{A}$, and now nothing can stop 12 from joining $\mathcal{A}$, a contradiction.

In proving above that every fibre of 2^n contains the lexical chain $\mathcal{L}(X)$ for some $X \in 2^n$, by relabeling we have actually shown that *for every ordering α of $\{1, 2, \dots, n\}$, every fibre of 2^n contains $\mathcal{L}_\alpha(X)$ for some $X \in 2^n$* . With this observation and a little counting, form (a) of Theorem 1 can be strengthened considerably.

COROLLARY. *Every fibre of 2^n contains at least $n!/2^{n-1}$ maximal chains.*

Proof. We first count the number of lexical chains $\mathcal{L}(X)$ in 2^n . It is clear that the generating set X of a lexical chain $\mathcal{L}(X)$ must either be the smallest set in $\mathcal{L}(X)$ containing 1 or the largest set in $\mathcal{L}(X)$ missing 1, since 1 is either the first element added to X or the first element deleted, depending on whether $1 \in X$ or not. Thus each lexical chain $\mathcal{L}(X)$ is generated by two of the 2^n subsets of $[n]$, so there are exactly 2^{n-1} lexical chains.

By symmetry, there are exactly 2^{n-1} α -lexical chains for each ordering α . Since there are $n!$ orderings of $[n]$ and $n!$ maximal chains in 2^n , by symmetry each maximal chain must be of the form $\mathcal{L}_\alpha(X)$ (for some X) for exactly 2^{n-1} different orderings α . Therefore, since every fibre must contain an α -lexical chain for each α , every fibre must contain at least $n!/2^{n-1}$ different maximal chains. $\square$

To end this section, we note the following stronger version of form (b) of Theorem 1. For an ordering α of $[n]$, call a subset $\mathcal{C}$ of 2^n an α -generalized cutset if $\mathcal{C}$ intersects every α -lexical chain $\mathcal{L}_\alpha(X)$, $X \in 2^n$. Obviously every cutset is an α -generalized cutset for each α . Then for each α , every α -generalized cutset of 2^n contains a maximal antichain. This follows simply because if an α -generalized cutset $\mathcal{C}$ did not contain a maximal antichain, then its complement $2^n - \mathcal{C}$ would be a fibre and thus by the above observation must contain some α -lexical chain $\mathcal{L}_\alpha(X)$. But then $\mathcal{L}_\alpha(X)$ would be disjoint from $\mathcal{C}$, a contradiction.

Here is an application. Consider 2^n as the disjoint union of posets $\mathcal{P}_1$ and $\mathcal{P}_2$, where

$$\mathcal{P}_1 = \{X \in 2^n : X \subseteq [n-1]\}, \mathcal{P}_2 = \{X \in 2^n : n \in X\}.$$

Note that both $\mathcal{P}_1$ and $\mathcal{P}_2$ are isomorphic to 2^{n-1} . Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be cutsets of $\mathcal{P}_1$ and $\mathcal{P}_2$, respectively. Then we claim that $\mathcal{C}_1 \cup \mathcal{C}_2$ contains a maximal antichain of 2^n . For let α be any ordering of $[n]$ ending in the element n (for example, α could be the usual ordering). Then for any $X \in 2^n$, if $n \in X$ the upper cover of $\emptyset$ in $\mathcal{L}_\alpha(X)$ will be $\{n\}$, while if $n \notin X$ the lower cover of $[n]$ in $\mathcal{L}_\alpha(X)$ will be $[n-1]$. Hence for each $X \in 2^n$ either $\mathcal{L}_\alpha(X) - \{\emptyset\} \subseteq \mathcal{P}_2$ or $\mathcal{L}_\alpha(X) - \{[n]\} \subseteq \mathcal{P}_1$, which means that $\mathcal{C}_1 \cup \mathcal{C}_2$ is an α -generalized cutset, and the claim follows.

3. Sizes of minimal fibres of 2^n . In [2], Lonc and Rival made the following conjectures:

(i) The minimum size of a fibre of 2^n is

$$\begin{aligned} 2^{(n+1)/2} + 2^{(n-1)/2} - 1, & \quad n \text{ odd,} \\ 2^{n/2+1} - 1, & \quad n \text{ even.} \end{aligned}$$

(ii) The maximum size of a minimal fibre of 2^n is $2^{n-1} + 1$.

Both sizes are attained by cones, (i) by the cone of an element in the middle level(s) of 2^n , (ii) by the cone of an atom or co-atom of 2^n .

Regarding (i), we can show at least that every fibre of 2^n is of size exponential in n .

THEOREM 2. *If $\mathcal{F}$ is a fibre of 2^n then $|\mathcal{F}| = \Omega(1.25^n)$.*

Proof. Let k be an integer, $1 \leq k < n/2$, and let S be an element of 2^n of size $2k-1$. Let $\mathcal{T}$ be the family of all k -element subsets of S . We claim that $\mathcal{T} \cup \bar{\mathcal{T}}$ is a maximal antichain in 2^n . $\mathcal{T} \cup \bar{\mathcal{T}}$ is certainly an antichain; given $T, U \in \mathcal{T}$, T and U have at least one element of S in common, so $T \not\subseteq \bar{U}$, and $T \cup U \neq [n]$ so $T \not\supseteq \bar{U}$. To show $\mathcal{T} \cup \bar{\mathcal{T}}$ is maximal, let $X \in 2^n$; then X either contains at least k elements of S and thus contains some member of $\mathcal{T}$, or misses at least k elements of S and thus is

contained in some member of $\bar{\mathcal{T}}$. Hence X cannot be added to $\mathcal{T} \cup \bar{\mathcal{T}}$ to get a larger antichain.

Consider all such maximal antichains $\mathcal{T} \cup \bar{\mathcal{T}}$ corresponding to elements $S \in 2^n$, $|S| = 2k - 1$. Any fibre $\mathcal{F}$ of 2^n must intersect each of these maximal antichains, of which there are $\binom{n}{2k-1}$, one for each S . Consider a k -element set T in $\mathcal{F}$; T is a k -element subset of exactly $\binom{n-k}{k-1}$ sets S of size $2k - 1$ and so can be the representative in $\mathcal{F}$ of only this many maximal antichains $\mathcal{T} \cup \bar{\mathcal{T}}$. Similarly, an $(n - k)$ -element set $\bar{T}$ in $\mathcal{F}$ can only belong to (the $\bar{\mathcal{T}}$ -part of) $\binom{n-k}{k-1}$ maximal antichains $\mathcal{T} \cup \bar{\mathcal{T}}$. Thus $\mathcal{F}$ must contain at least

$$\binom{n}{2k-1} / \binom{n-k}{k-1}$$

elements of size k or $n - k$. This turns out to be optimized for $k = n/5$. The above quotient can then be estimated by Stirling's approximation to yield that $|\mathcal{F}|$ is of order at least

$$\frac{n^n}{(2n/5)^{2n/5} (3n/5)^{3n/5}} \cdot \frac{(n/5)^{n/5} (3n/5)^{3n/5}}{(4n/5)^{4n/5}} = (5/4)^n. \quad \square$$

Any family $\mathcal{S}$ of $(2k - 1)$ -element subsets of $[n]$ with the property that no two members of $\mathcal{S}$ intersect in k or more elements will yield $|\mathcal{S}|$ pairwise disjoint maximal antichains $\mathcal{T} \cup \bar{\mathcal{T}}$ of 2^n . Consider the graph whose vertices are all $(2k - 1)$ -element subsets of $[n]$ (where k is a fixed proportion of n to be chosen later), two such subsets being adjacent if they have at least k elements in common. Then the degree of any vertex is

$$\sum_{l=k}^{2k-2} \binom{2k-1}{l} \binom{n-2k+1}{2k-l-1}.$$

If $k < n/4$, it can easily be checked that the first term of this sum is the largest and, exponentially, will dominate the sum. Thus there will exist an independent set of order at least

$$\frac{\binom{n}{2k-1}}{\binom{2k-1}{k} \binom{n-2k+1}{k-1}}.$$

Putting $k = \lambda n$ and applying Stirling's approximation, this quotient is exponentially equal to

$$\left(\frac{(1-3\lambda)^{1-3\lambda}}{(16\lambda)^\lambda (1-2\lambda)^{2-4\lambda}} \right)^n$$

which is maximized for λ the real root of

$$112x^3 - 88x^2 + 20x - 1,$$

i.e., $\lambda \approx 0.0692304$. Plugging this value of λ into the above expression, we obtain: *there exists a family of $\Omega(1.0674422^n)$ pairwise disjoint maximal antichains in 2^n .*

Of course 2^n cannot have more than $O(\sqrt{2})^n$ pairwise disjoint maximal antichains since there is a fibre of this size, namely the cone of a middle-level element.

Problem. Find better bounds for the maximum number of pairwise disjoint maximal antichains in 2^n .

The base 1.25 in Theorem 2 compares quite well with the base $\sqrt{2}$ in conjecture (i) above. Further evidence in support of conjecture (i) might be presumed from the corollary to Theorem 1, as the cone of a middle-level element of 2^n contains $(n/2)!^2 \approx \pi n(n/2e)^n$ maximal chains, scarcely more than the $n!/2^{n-1} \approx 2\sqrt{2}\pi n(n/2e)^n$ maximal chains known to exist in every fibre from the corollary.

On the other hand, we can disprove conjecture (ii) for all $n \geq 4$, although not resoundingly: the following is an example of a minimal fibre $\mathcal{F}$ of 2^n of size $2^{n-1} + 2$, one more than the conjectured maximum size!

We use the notation of example (2) following the proof of Theorem 1, so that 12 means $\{1, 2\}$, etc. Let

$$\mathcal{F} = \{12\}^+ \cup \{\overline{12}\}^- \cup \{2, \overline{1}\}$$

(Fig. 3). Then $|\mathcal{F}| = 2^{n-2} + 2^{n-2} + 2 = 2^{n-1} + 2$. Why is $\mathcal{F}$ a fibre? If $\mathcal{A}$ were a maximal antichain missing $\mathcal{F}$ then we must have $1 \in \mathcal{A}$, since 1 is the only element of the cone of 12 missing from $\mathcal{F}$. Similarly $\overline{2} \in \mathcal{A}$ by considering the cone of $\overline{12}$. But this is nonsense since $1 \subset \overline{2}$. Finally, why is $\mathcal{F}$ minimal? $\mathcal{F} - \{[n]\}$ is obviously not a fibre of 2^n since $\{[n]\}$ is itself a maximal antichain, and $\mathcal{F} - \{2\}$ is not a fibre of 2^n because the maximal antichain $\{2, \overline{2}\}$ misses $\mathcal{F} - \{2\}$. So to finish the proof, by symmetry we need only show that $\mathcal{F} - \{X\}$ is not a fibre whenever $12 \subseteq X \subset [n]$. To do this we claim that

$$\mathcal{A} = \{\overline{2}, X\} \cup \{2y \mid y \notin X\}$$

is a maximal antichain. Since (for $n \geq 4$) $\mathcal{A}$ intersects $\mathcal{F}$ only in the element X , we would be done. It is easy to check that $\mathcal{A}$ is an antichain. Moreover, if $Y \in 2^n$ with $Y \not\subseteq \overline{2}$ and $Y \not\subseteq X$ then $2 \in Y$ and there is $y \in Y$, $y \notin X$. Thus $2y \subseteq Y$, so Y cannot be added to $\mathcal{A}$.

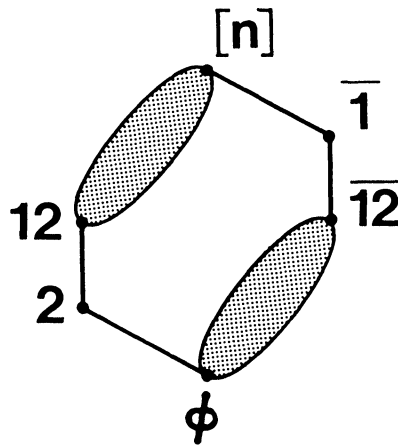


FIG. 3

Recently Füredi, Griggs, and Kleitman [1] have found minimal cutsets of 2^n which contain almost all elements of 2^n . We withhold judgment on whether or not the above minimal fibre is largest.

Acknowledgment. We would like to thank Vojtěch Rödl for conversations helpful in establishing the exponential lower bound, given in § 3, for the number of pairwise disjoint maximal antichains in 2^n .

REFERENCES

- [1] Z. FÜREDI, J. R. GRIGGS, AND D. J. KLEITMAN, *A minimal cutset of the Boolean lattice with almost all members*, IMA Preprint Series 421, 1988.
- [2] Z. LONC AND I. RIVAL, *Chains, antichains, and fibres*, J. Combin. Theory Ser. A, 44 (1987), pp. 207–228.
- [3] R. NOWAKOWSKI, *Cutsets of Boolean lattices*, Discrete Math., 63 (1987), pp. 231–240.

TRANSVERSALS OF VERTEX PARTITIONS IN GRAPHS*

MICHAEL R. FELLOWS†

Abstract. This paper studies graph properties of the following forms: For every partition of the vertex set that satisfies an upper (or lower) bound on the number of elements in each partition class, there is a transversal of the partition that is an independent (or dominating) set. A possible application to fault-tolerant data storage is discussed, and bounds for the parameters that are functions of minimum and maximum degree are established. The complexity of associated decision problems is also addressed.

Key words. Ramsey theory, fault-tolerant computing, graph theory, analysis of algorithms

AMS(MOS) subject classifications. 05C55, 05C15, 68Q25, 68R10

1. Introduction. In this paper we study a number of graph-theoretic parameters that are defined by statements of the form: “For every partition of the vertex set satisfying P there is a transversal of the partition satisfying Q .” We discuss a possible application to a problem of fault-tolerant data storage for the parameter that we term the *partition-independence number* of a graph. A number of challenging conjectures concerning this parameter have circulated widely in recent years. We also study an analogously defined *partition-domination number* and variants of these two parameters.

In § 2 we introduce some basic terminology and notation and discuss two conjectures. In § 3 we study the partition-independence number and related parameters. Our main results give bounds on these that are functions of the maximum degree of a graph. In § 4 we consider the analogously defined partition-domination number and obtain bounds that are a function of the minimum degree of a graph. In § 5 we consider the complexity of some related computational problems. In § 6 we conclude with a discussion of open problems.

2. Notation and terminology. All graphs $G = (V, E)$ in this paper are simple, without loops or multiple edges. An *independent set* of vertices in a graph is a set $V' \subseteq V$ such that for all $x, y \in V'$, $xy \notin E$. A set of vertices $V' \subseteq V$ is a *dominating set* if for every vertex $u \in V$ there is a vertex $w \in V'$ such that $u = w$ or $uw \in E$. Other standard graph-theoretic terminology and notation may be found in [Bo].

DEFINITION. A partition π of a set X is *k-thick* if for each class $[x]$ of π , $x \in X$, we have $|[x]| \geq k$. Similarly a partition is *k-thin* (*k-exact*) if each class of the partition has at most (exactly) k elements.

We may sometimes refer to a partition of the vertex set of a graph as a coloring, and will also consider partial colorings. A partial coloring of V is termed *k-thick* if each color class contains at least k vertices. Analogously to partitions, partial colorings will be termed *k-thin* or *k-exact*. A *transversal* of a partition (or coloring or partial coloring) is a set of distinct representatives of the classes of the partition (or coloring or partial coloring).

DEFINITION. A partition π of a set X is *orthogonal* to a partition $\pi^\perp$ of X if each class of π is a transversal of $\pi^\perp$ and vice versa.

DEFINITION. The *partition independence number* $\text{pi}(G)$ of a graph $G = (V, E)$ is the least cardinal k such that for every k -thick partition π of V there is a transversal T of π that is an independent set of vertices in G .

* Received by the editors December 19, 1988; accepted for publication (in revised form) July 26, 1989.

† Department of Computer Science, University of Idaho, Moscow, Idaho 83843. This research was supported in part by National Science Foundation grant MIP-8603879, by Office of Naval Research contract N00014-88-K-0456, and by National Aeronautics and Space Administration grant NAGW-1406.

The following conjecture has recently attracted widespread attention (but as of yet, no solution). By C_ω we denote the two-way infinite path with countable vertex set.

CONJECTURE 1. $\text{pi}(C_\omega) = 3$.

It is easy to see that for every graph G we must have $\text{pi}(G) \leq |G|$. The lower bound $\text{pi}(C_\omega) \geq 3$ is easily discerned from the 2-thick coloring of C_ω indicated by the following sequence.

...444121323444...

Truthfully, that isn't *exactly* the way we originally heard the conjecture, and it is not the only form in which it has circulated. It has also been termed the "Circle-Triangle Conjecture"; that no matter how a collection of triangles is inscribed in a circle (with all points of intersection with the circle distinct), one can always choose one vertex of each triangle so that no two of the chosen vertices are consecutive with respect to the circle. In this form we learned the conjecture first from Frank Hsu. It has circulated widely for a few years, partly through the offices of Paul Erdős. Theorem 2 shows that these conjectures are equivalent. It is pleasant to reformulate the conjecture as stated above as a problem concerning colorings of the integers; if an infinite number of colors are used, with each color used at least three times, then one can choose one integer of each color so that no consecutive pair i and $i + 1$ are chosen.

Somewhat more than Conjecture 1 may well be true. It seems that one can find not just one independent transversal in a 3-exact coloring of C_ω but three of them!

DEFINITION. The *strong partition independence number* $\text{pi}^*(G)$ of a graph $G = (V, E)$ is the least cardinal k such that for every k -exact partition π of V there is an orthogonal partition $\pi^\perp$ of V with each class of $\pi^\perp$ an independent set of vertices in G .

CONJECTURE 2. $\text{pi}^*(C_\omega) = 3$.

To the best of our knowledge, this conjecture was made most primordially by Schur (in an equivalent form) and is also due independently to Schönheim [Sc]. In the next section we will show that $\text{pi}^*(C_\omega) \leq 4$.

We make the following analogous definitions concerning transversals that are dominating sets.

DEFINITION. The *partition domination number* $\text{pd}(G)$ of a graph $G = (V, E)$ is the greatest cardinal k such that for every k -thin partition π of V there is a transversal T of π that is a dominating set of vertices of G . The *strong partition domination number* $\text{pd}^*(G)$ of a graph G is the greatest cardinal k such that for every k -exact partition π of V there is an orthogonal partition $\pi^\perp$ of V with each class of $\pi^\perp$ a dominating set of vertices in G .

3. Partition independence. The following discussion illustrates the notion of partition independence with a possible application.

Living with destructive reading. We consider a measure of the cost of insuring the integrity of a database against a failure mode of the read mechanism in which the data in a neighborhood of a memory location x is destroyed in the process of reading from x . Such a failure we term *destructive reading*.

The memory medium is modeled as a simple undirected graph. For example, a tape can be modeled as a path. Each node represents a unit of memory capacity. A unit of memory can store exactly one unit of data. Copies of a data unit are stored at several memory locations to achieve the required fault tolerance.

In the following example, a path with 12 vertices represents the memory medium. There are four units of data, each stored at three locations. The indicated sequence of read operations allows all four units of data to be retrieved, even in the event that each

of the read operations destroys all of the data in the immediate neighborhood of the memory location that is read.

Example.

	read time	data configuration											
	0	<i>a</i>	<i>c</i>	<i>b</i>	<i>d</i>	<i>a</i>	<i>d</i>	<i>b</i>	<i>b</i>	<i>a</i>	<i>c</i>	<i>d</i>	<i>c</i>
<i>a</i> is read	1	<i>a</i>	<i>c</i>	<i>b</i>	*	*	*	<i>b</i>	<i>b</i>	<i>a</i>	<i>c</i>	<i>d</i>	<i>c</i>
<i>b</i> is read	2	<i>a</i>	<i>c</i>	<i>b</i>	*	*	*	*	*	<i>a</i>	<i>c</i>	<i>d</i>	<i>c</i>
<i>c</i> is read	3	*	*	*	*	*	*	*	*	<i>a</i>	<i>c</i>	<i>d</i>	<i>c</i>
<i>d</i> is read	4	*	*	*	*	*	*	*	*	<i>a</i>	*	*	*

We model the situation where the database is dynamic. Thus the copies of a new item of data are not necessarily stored in adjacent memory locations, but rather only in some number of “first available” locations. We are interested in knowing, for a given graph G representing the memory medium, the *least* positive integer k such that no matter how data is assigned to the vertices of G in such a way that at least k copies of each unit of data is stored, there is some sequence of (possibly faulty) read operations to recover all of the stored data. What is required, then, is a set of *read locations* that

- (1) represents all of the data, and
- (2) is an independent set in the graph of the storage medium.

The parameter $\text{pi}(G)$ models these requirements and measures the minimum amount of duplication of records necessary to achieve fault-tolerance against destructive reads for a memory medium represented by G .

The following lemmas are easily established. A subgraph H of a graph G is termed *spanning* if $V(H) = V(G)$. By ωG we denote the disjoint union of countably many copies of G .

LEMMA 1. *If H is a spanning subgraph of G then $\text{pi}(H) \leq \text{pi}(G)$ and $\text{pi}^*(H) \leq \text{pi}^*(G)$.*

LEMMA 2. *If H is a subgraph of G then $\text{pi}(\omega H) \leq \text{pi}(\omega G)$ and $\text{pi}^*(\omega H) \leq \text{pi}^*(\omega G)$.*

LEMMA 3. *For all G , $\text{pi}(G) \leq \text{pi}^*(G)$.*

The next lemma, giving exact values of pi and pi^* for the graphs mK_n , will be quite useful.

LEMMA 4. *For $n \geq 1$ and $m \geq n - 1$, $\text{pi}^*(mK_n) = \text{pi}(mK_n) = \text{pi}^*(mK_{1,n-1}) = \text{pi}(mK_{1,n-1}) = n$.*

Proof. By Lemmas 1 and 3 a lower bound of n is established by showing that $\text{pi}((n-1)K_{1,n-1}) \geq n$. Let $v(i, 0)$ be the central vertex of the i th copy of the star $K_{1,n-1}$ and let the ray vertices be denoted $v(i, 1), \dots, v(i, n-1)$. Let π be the $(n-1)$ -exact partition with classes $\{v(i, 0) : 1 \leq i \leq n-1\}$, $\{v(1, i) : 1 \leq i \leq n-1\}$, $\dots$, $\{v(n-1, i) : 1 \leq i \leq n-1\}$. It is easy to check that there is no independent transversal of π .

Also by Lemmas 1 and 3, an upper bound of n is shown by arguing that $\text{pi}^*(mK_n) \leq n$ for all $m \geq n-1$. This follows easily by induction on n using Hall's Theorem, since at least j colors must occur on any j copies of K_n for an n -exact coloring. Removing the vertices of one independent transversal yields an $(n-1)$ -exact coloring of mK_{n-1} to which the induction hypothesis can be applied. $\square$

The following theorem supplies a bound on the partition independence number of finite graphs that is a linear function of the maximum degree.

THEOREM 1. *If $G = (V, E)$ is a finite graph of maximum degree Δ , $\Delta \geq 1$, then $\text{pi}(G) \leq 16\Delta$.*

Proof. We use the Lovasz Local Lemma [Sp]. Suppose that π is a k -thick partition of V . Without loss of generality, we may assume that each class of π contains at most

$2k - 1$ elements since otherwise the class can be split. For each class of π , independently and with equal probability choose a representative. The set of chosen representatives is a transversal T of π .

The vertex set of the event graph is E . For each edge $uv \in E$ define the event E_{uv} to be $\{u, v\} \subseteq T$. The probability of E_{uv} is bounded above by

$$(1) \quad P(E_{uv}) \leq \left(\frac{1}{k}\right)^2.$$

The event E_{uv} is independent of any Boolean combination of events E_{xy} for which $\{[u], [v]\} \cap \{[x], [y]\} = \emptyset$. Correspondingly, in the event graph we place an edge between uv and xy if and only if $\{[u], [v]\} \cap \{[x], [y]\} \neq \emptyset$. The degree of a vertex uv of the event graph constructed in this manner is at most $2\Delta(2k - 2) + 2(\Delta - 1) = 4\Delta k - 2\Delta - 2$. The Local Lemma applies if the following inequality is satisfied.

$$(2) \quad \frac{1}{k^2} \leq \frac{1}{4(4\Delta k - 2\Delta - 2)}$$

or equivalently,

$$(3) \quad k^2 \geq 8(2\Delta k - \Delta - 1).$$

This holds if $k \geq 16\Delta$. When the Local Lemma applies we are guaranteed that there exists a transversal T such that for all $uv \in E(G)$ either $u \notin T$ or $v \notin T$ and therefore T is an independent set of vertices in G . $\square$

As a Corollary to the above theorem we can obtain the bound $\text{pi}(C_\omega) \leq 32$ by a compactness argument. A much improved upper bound is obtained in Theorem 4 below. The following definitions are introduced as a technical convenience.

DEFINITION. The *partial partition independence number* $\text{ppi}(G)$ of a graph G is the least cardinal k such that every k -thick partial coloring of $V(G)$ possesses a transversal that is an independent set in G .

DEFINITION. The *partial strong partition independence number* $\text{ppi}^*(G)$ of a graph G is the least cardinality k such that for every k -exact partial coloring of $V(G)$, there is an orthogonal partition of the set of colored vertices with each class of the orthogonal partition a transversal of the partial coloring and an independent set in G .

The following are immediate consequences of the definitions.

LEMMA 5. For every graph G , $\text{pi}(G) \leq \text{ppi}(G)$.

LEMMA 6. For every graph G , $\text{pi}^*(G) \leq \text{ppi}^*(G)$.

The next lemma supplies a useful compactness argument.

LEMMA 7. Suppose $G_0, G_1, G_2, \dots$ is a sequence of finite subgraphs of G satisfying

(0) $G_0 = \emptyset$.

(1) For $i = 0, 1, 2, \dots$ $G_i \subseteq G_{i+1}$.

(2) For every finite subgraph H of G , $\exists i$ such that H is a subgraph of G_i .

(3) For $i = 0, 1, 2, \dots$ $\text{ppi}(G_i) \leq k$. (For $i = 0, 1, 2, \dots$ $\text{ppi}^*(G_i) \leq k$.)

Then $\text{ppi}(G) \leq k$ ($\text{ppi}^*(G) \leq k$).

Proof. We argue the case of $\text{ppi}^*(G)$. The proof for $\text{ppi}(G)$ is entirely similar. Let π be a k -exact partial coloring of $V(G)$. For $i = 0, 1, 2, \dots$ let

$$\pi_i = \{\alpha : \alpha \in \pi \text{ and } |\alpha \cap V(G_i)| = k\}$$

be the induced k -exact partial coloring of $V(G_i)$.

We describe an infinite tree T as follows. The vertex set of T is the set $V(T) = \{(i, \pi'_i) : i \in \{0, 1, 2, \dots\}\}$ and π'_i is a partition of the colored vertices of G_i under the partial coloring π_i , with π'_i orthogonal to π_i and with each class of π'_i independent in G_i .

By (3) there is at least one vertex on the i th level of T for $i = 0, 1, 2, \dots$. The parent of (i, π'_i) in T is the vertex $(i-1, \pi'_{i-1})$ where π'_{i-1} is the restriction of π'_i to the vertices of the partial coloring π_{i-1} . Note that π'_{i-1} is orthogonal to π_{i-1} since π'_i is orthogonal to π_i and each class of π'_{i-1} is independent in G_{i-1} .

By (0), we are assured that T has root $(0, \emptyset)$. By König's Lemma there is an infinite path $(0, \pi'_0), (1, \pi'_1), \dots$ in T . Define the partition $\pi^\perp$ orthogonal to π to be the partition of the vertices of G colored by π corresponding to the equivalence relation: $u \sim v$ if $\exists i$ such that $\{u, v\} \subseteq \alpha \in \pi'_i$. This is well-defined by (1). It is easy to check that this is an equivalence relation.

It remains to see that each equivalence class $[u]$ is an independent transversal of π . That $[u]$ is an independent set of vertices of G follows from the fact that each class of π'_i is an independent set. Let α be a class of π and suppose $u \in \beta \in \pi$. By (2), $\exists i_0$ such that $\{\alpha, \beta\} \subseteq \pi_i$ for $i \geq i_0$. Let γ be the class of π'_i containing u . Since γ is a transversal of π_i there is a vertex $v \in \gamma$ such that $v \in \alpha$. So then $u \sim v$ and $[u]$ meets every class α of π . Now suppose also that $u \sim v'$ with $v' \in \alpha$. Then $\exists j$ and a class γ' of π'_j with $\{u, v, v'\} \subseteq \gamma'$. Since γ' is a transversal of π_j and $\alpha \in \pi_j$ we must have $v = v'$. $\square$

THEOREM 2. *The following statements are equivalent:*

- (1) For all n , $\text{pi}(C_n) \leq k$.
- (2) For all n , $\text{pi}(P_n) \leq k$.
- (3) $\text{pi}(C_\omega) \leq k$.

Proof. Statement (1) implies (2) by Lemma 1. Statement (2) easily implies that for all n , $\text{ppi}(P_n) \leq k$, and (3) follows by Lemma 7 and Lemma 5.

To see that (3) implies (1), consider the projection map p from C_ω (as the universal cover of C_n) to C_n . Thus, if we take the vertex set of C_ω to be the integers, and the vertex set of C_n to be the set $\{0, \dots, n-1\}$, then $p(i) = i \pmod n$. Suppose γ is a k -thick coloring of $V(C_n)$ that contradicts (1). For each integer j divisible by k , let $V_j = \{j, \dots, j+k-1\}$. Using a different set of colors for each set of vertices V_j , we induce a k -thick coloring γ' of C_ω according to the projection map p .

Suppose T' is an independent transversal of γ' . The set $T_0 = p(T' \cap V_0)$ is a transversal of γ . Since it is not an independent set, there are adjacent vertices u and v of C_n with $u, v \in T_0$. This implies that $p^{-1}(u)$ and $p^{-1}(v)$ are adjacent in C_ω , a contradiction, or that $\{p^{-1}(u), p^{-1}(v)\} = \{0, k-1\}$. In the latter case, $k \notin T'$. The set $T_1 = p(T' \cap V_1)$ is also a transversal of γ , and therefore not an independent set in C_n . The only possibility now for adjacent vertices $x, y \in T_1$ is that $p^{-1}(x)$ and $p^{-1}(y)$ are adjacent in C_ω a contradiction. $\square$

THEOREM 3. *The following statements are equivalent:*

- (1) For all n divisible by k , $\text{pi}^*(C_n) \leq k$.
- (2) For all n divisible by k , $\text{pi}^*(P_n) \leq k$.
- (3) $\text{pi}^*(C_\omega) \leq k$.

Proof. The argument is similar to that for Theorem 2 and is left to the reader. $\square$

THEOREM 4. $3 \leq \text{pi}^*(C_\omega) \leq 4$.

Proof. A 2-thick coloring of C_ω that demonstrates the lower bound was noted in § 1. By Theorem 3 it is enough to show that for all n divisible by 4, $\text{pi}(C_n) \leq 4$. Accordingly, let π be a 4-exact partition of $V(C_n) = \{0, \dots, n-1\}$. For each class δ of π , partition δ into two 2-element sets δ_0 and δ_1 . This gives a 2-exact partition π' of $V(C_n)$, $\pi' = \{\delta_0 \mid \delta \in \pi\} \cup \{\delta_1 \mid \delta \in \pi\}$. The graph $(n/2)K_2$ is a subgraph of C_n , where each copy of K_2 has vertices i and $i+1$ for $i = 0, 2, \dots, n-2$. By the proof of Lemma 4, there is a transversal T' of π' such that T' intersects each copy of K_2 in precisely one vertex. For each color class δ of π there are precisely two vertices in $\delta_2 = \delta - T'$. Let π'' be the 2-

exact partition of $V' = \{0, \dots, n-1\} - T'$, $\pi'' = \{\delta_2 | \delta \in \pi\}$, and let H denote the subgraph of C_n induced by V' . The subgraph H is isomorphic to a subgraph of $(n/4)K_2$, so by Lemmas 1 and 4 there are transversals T''_0 and T''_1 of π'' that are independent in H , with $T''_0 \cap T''_1 = \emptyset$. Since H is an induced subgraph of C_n , T''_0 and T''_1 are independent transversals of π . Repeating this argument for the subgraph H' induced by $\{0, \dots, n-1\} - T''_0 - T''_1$ yields the theorem. $\square$

Next we give a bound on $\text{pi}^*(G)$ in terms of the maximum degree of G . Although we were able to obtain such a bound on $\text{pi}(G)$ by a probabilistic argument, those tools do not seem to be as readily applicable to the strong partition independence number.

THEOREM 5. *If G is a graph of maximum degree Δ then $\text{pi}^*(G) \leq 2(6^{\Delta-1})$.*

Proof. Induction on Δ . Lemma 4 establishes the base step. For the induction step we first argue that we can restrict our attention to graphs G of the form $G \cong 6H$ where H has no isolated vertices. If G has isolates, then let G' denote the graph obtained from two isomorphic copies of G by adding a set of edges forming a matching between the sets of isolated vertices in the two copies. To see that $\text{pi}^*(G) \leq \text{pi}^*(G')$, note that any k -exact partition π of $V(G)$ induces a k -exact partition π' of G' just by taking π' to be the union of the copies of π . If $\pi^\perp$ is a partition of $V(G')$ orthogonal to π' , then $\pi^\perp$ restricts to a partition of $V(G)$ orthogonal to π , and for class α of $\pi^\perp$ that is independent in G' the restriction of α is independent in G . By essentially the same argument, $\text{pi}^*(H) \leq \text{pi}^*(6H)$ for any graph H . This same argument also shows that for $G \cong 6H$ we may further restrict our attention to k -exact partitions of $V(G)$ that have isomorphic k -exact restrictions to the copies of H .

For $G \cong 6H$, let the copies of H be denoted $H_1, \dots, H_6$. Since the minimum degree of H is at least one, we can partition $V(H)$ into 2- and 3-element sets S so that the subgraph H_S of H induced by each set S has minimum degree at least one. For $i = 1, \dots, 6$ let $\{H_{ij}\}_{j \in J}$ denote the corresponding family of subgraphs of H_i . (To find such a partition of $V(H)$, begin with a maximal matching.) A central role in our argument is played by a decomposition of G into subgraphs of order 6 that are formed by collections of the subgraphs H_{ij} as follows. For $j \in J$, if $|H_{ij}| = 2$ then let $G_{j,1} = H_{1,j} \cup H_{2,j} \cup H_{3,j}$ and let $G_{j,2} = H_{4,j} \cup H_{5,j} \cup H_{6,j}$. For $j \in J$, if $|H_{ij}| = 3$ then let $G_{j,1} = H_{1,j} \cup H_{2,j}$, let $G_{j,2} = H_{3,j} \cup H_{4,j}$ and let $G_{j,3} = H_{5,j} \cup H_{6,j}$. Thus $\{G_{jk}\}$ is a collection vertex disjoint subgraphs of G , each of order 6, corresponding to a partition of $V(G)$, and each subgraph $G_{j,k}$ has minimum degree at least one.

Now suppose that $G \cong 6H$ has maximum degree $\Delta + 1$ and let π_H be a $2(6^\Delta)$ -exact partition of $V(H)$. Let π denote the induced $2(6^\Delta)$ -exact partition of $V(G)$ formed as the union of 6 copies of π_H . For each class α of π , partition α into $2(6^{\Delta-1})$ subclasses, each having 6 elements. Let π_0 denote the 6-exact partition of $V(G)$ into these subclasses. By Lemma 4 there is an orthogonal partition $\pi_0^\perp$ for which each class is an independent set of vertices in the disjoint union of the subgraphs $G_{j,k}$ of G . Let γ be a class of $\pi_0^\perp$, and let G_γ denote the subgraph of G induced by γ . Since each subgraph $G_{j,k}$ has minimum degree at least one, G_γ has maximum degree Δ . Because γ is a transversal of π_0 , there are exactly $2(6^{\Delta-1})$ vertices of G_γ belonging to each class α of π . By the induction hypothesis, there is a partition $\pi_\gamma^\perp$ of $V(G_\gamma)$ that is orthogonal to the restriction π_γ of π to G_γ , with each class of $\pi_\gamma^\perp$ independent in G_γ , and therefore in G , since G_γ is an induced subgraph of G . Let $\pi^\perp = \bigcup_{\gamma \in \pi_0^\perp} \pi_\gamma^\perp$. Since $\pi^\perp$ is orthogonal to π we are done. $\square$

The bound on $\text{pi}^*(G)$ given by Theorem 5 is likely to be quite poor. For some time it seemed possible that always $\text{pi}^*(\omega G) \leq \Delta(G) + 1$. The following example, due to Truszczyński, demonstrates that $\text{pi}^*(\omega G)$ can be as large as $\Delta(G) + 2$, and this is as much as is presently known.

Example. $\text{pi}^*(\omega C_4) = 4$. An upper bound of 4 follows from Lemma 1 and Lemma 4. A 3-exact coloring without an independent transversal is described:

$$(1212), (3434), (1423), \dots$$

It is not quite obvious that pi^* behaves monotonically. The following reassures us that the parameter is at least this well-behaved.

DEFINITION. A graph G is k^* -good if for every k -exact partition π of $V(G)$ there is an orthogonal partition $\pi^\perp$ with each class of $\pi^\perp$ independent in G .

THEOREM 6. *If G is k^* -good and $k < j$ then G is j^* -good, for $k, j \in \mathbb{N}$ and G any infinite graph.*

Proof. Let a $(k + 1)$ -exact partition π of $V(G)$ be given. Let T be any transversal. For each class C of π let $C' = C - T$. Each set C' contains exactly k vertices of G . Let π_T be any k -exact partition of T . Then $\pi' = \pi_T \cup \{C' \mid C \in \pi\}$ is a k -exact partition of $V(G)$, so there is an independent transversal T' of π' . Let $R = T' \cap (\cup_{C \in \pi} C')$ be the representatives in T' of the “shortened” classes of the original partition π . R is an independent transversal of π . For each class C of π let $C'' = C - R$. Each set C'' contains exactly k vertices of G . Let π_R be any k -exact partition of R . Then $\pi'' = \pi_R \cup \{C'' \mid C \in \pi\}$ is a k -exact partition of $V(G)$. By the hypothesis of the theorem, there is an orthogonal partition π_0 of $V(G)$ with each class of π_0 an independent transversal of π'' . For each class D of π_0 let $D' = D \cap (\cup_{C \in \pi} C')$. Since $D' \subseteq D$, D' is an independent set of vertices that is a transversal of π . Thus $\pi^\perp = \{R\} \cup \{D' \mid D \in \pi_0\}$ is a suitable orthogonal partition of $V(G)$. The theorem follows by elementary induction. $\square$

4. Partition domination. The *partition domination* number $\text{pd}(G)$ of a graph G is the greatest positive integer k such that for every k -thin partition of $V(G)$ there is a transversal that is a dominating set.

Note that the $\text{pd}(G)$ is well-defined, since there is a unique 1-thin partition of $V(G)$ for any graph G , and the unique transversal of this partition is $V(G)$, a dominating set.

By analogy with $\text{pi}^*(G)$ we define the parameter $\text{pd}^*(G)$ to be the greatest positive integer k such that for every k -exact partition π of $V(G)$ there is an orthogonal partition $\pi^\perp$ with each class of $\pi^\perp$ a dominating set of vertices in G . Clearly $\text{pd}^*(G) \geq 1$ for every graph G . The following theorem shows that this parameter also behaves monotonically for graphs of the form ωG .

THEOREM 7. *If ωG is $(k + 1)^*$ -good (with respect to the definition of pd^*) then ωG is k^* -good.*

Proof. Let γ be a k -exact partition of vertex set of ωG . Form $k + 1$ disjoint copies of ωG , each copy partitioned according to γ . The union of these partitions yields a k -exact partition of the vertices of $(k + 1)(\omega G)$, which is, of course, isomorphic to ωG . Let $H_0, \dots, H_k$ denote the copies of ωG , let H denote the union of these copies and let $\gamma_0, \dots, \gamma_k$ denote the corresponding partitions, each isomorphic to γ . For a class C of γ , let $C_0, \dots, C_k$ denote the corresponding classes of $\gamma_0, \dots, \gamma_k$. Let β be the partition of the vertex set of H defined $\beta = \cup_{i=1}^k \{C'_i \mid C_i \in \gamma_i\}$ where for $C \in \gamma$ and for $i = 1, \dots, k$, $C'_i = C_i \cup \{x_i\}$, where $C_0 = \{x_1, \dots, x_k\}$. In other words, each class C_0 of γ_0 is “dismembered” and the k vertices of C_0 are matched to the classes $C_1, \dots, C_k$ which are correspondingly augmented. This yields the $(k + 1)$ -exact partition β of H .

By the hypothesis of the theorem, there is a partition $\beta^\perp$ orthogonal to β such that each class of $\beta^\perp$ is a dominating set of vertices in H . Let $D_0, \dots, D_k$ denote these classes. Define the indicator function $f : V(H_1) \rightarrow \{0, \dots, k\}$ by $f(v) = i$ if $v \in D_i$. Let $f' :$

$V(H_1) \rightarrow \{1, \dots, k\}$ be the map defined

$$f'(v) = \begin{cases} i & \text{if } v \in D_i \text{ and } i \neq 0 \\ j & \text{if } v \in D_0 \text{ and } j \notin S_v \end{cases}$$

where $S_v = \{k \mid v \in C_1 \in \gamma_1 \text{ and } \exists u \in C_1 \text{ with } f(u) = k\}$.

To see that f' is well-defined, note that since C'_1 is the augmentation of C_1 with exactly one vertex of C_0 , there is exactly one index j in the range $1 \leq j \leq k$ omitted from S_v in the case that $v \in D_0$.

We claim that f' describes a partition $\gamma_1^\perp$ of $V(H_1)$ orthogonal to γ_1 . This follows from the fact that f' is one-to-one for any restriction to a class C_1 of γ_1 (by the definition of f'). Furthermore, each class of $\gamma_1^\perp$ is a dominating set in H_1 since it is a superset of $D_i \cap V(H_1)$ for some i , $1 \leq i \leq k$. Since H_1 and γ_1 are isomorphic to the original partition γ of ωG , we are done. $\square$

THEOREM 8. *If G is a δ -regular graph with $\delta \geq 18k \log_2 k$, $k \geq 2$, then $\text{pd}(G) \geq k$.*

Proof. We again use the Lovasz Local Lemma [Sp]. Suppose that γ is a k -thin partition of $V(G)$. Independently and with equal probability choose one vertex from each class of the partition γ to form a transversal T . The vertex set of the event graph is $V(G)$. For $v \in V(G)$ define the event E_v to be $N[v] \cap T = \emptyset$. In other words, E_v is the event that v is not dominated by T . By the *class graph* of γ we refer to the graph with vertex set the classes of the partition, with classes $[u]$ and $[v]$ adjacent if and only if there are adjacent vertices x, y of G with $x \in [u]$ and $y \in [v]$. The event E_u is independent of any Boolean combination of events E_v for which the distance from $[u]$ to $[v]$ is at least 3 in the class graph. There is an edge uv in the event graph if and only if $d([u], [v]) \leq 2$ in the class graph. Since G is δ -regular, a partition class $[u]$ has degree at most $k\delta$ in the class graph, and there are at most $(k\delta)^2 + 1$ classes $[v]$ such that $d([u], [v]) \leq 2$ in the class graph. Since γ is k -thin, the degree of a vertex u in the event graph is at most $((k\delta)^2 + 1)k - 1$. The probability of E_v is bounded by

$$(1) \quad P(E_v) \leq \left(\frac{k-1}{k} \right)^{\delta+1}.$$

The Local Lemma will apply if

$$(2) \quad \left(\frac{k-1}{k} \right)^{\delta+1} \leq \frac{1}{4k(k^2\delta^2 + 1) - 4}.$$

The inequality (2) will hold if

$$(3) \quad \left(\frac{k-1}{k} \right)^{\delta+1} \leq \frac{1}{5k^3\delta^2}$$

or if

$$(4) \quad \left(1 + \frac{1}{k} \right)^{\delta+1} \geq 5k^3\delta^2.$$

We may rewrite the left-hand side as $(1 + 1/k)^{k(\delta+1/k)}$ and note that $(1 + 1/k)^k \geq \frac{9}{4}$, since $k \geq 2$. Thus (4) will hold if

$$(5) \quad 2^{(\delta+1/k)} \geq 5k^3\delta^2$$

or if

$$(6) \quad \frac{\delta + 1}{k} \geq \log_2 5 + 3 \log_2 k + 2 \log_2 \delta.$$

The inequality (6) will hold if $\delta \geq \delta_1 + \delta_2$ where $\delta_1 \geq 6k \log_2 k \geq k(\log_2 5 + 3 \log_2 k)$, $k \geq 2$, and where δ_2 is such that $\delta_2/k \geq 2 \log_2 \delta_2$.

Setting $\delta_2 = Ck \log_2 k$, we must have

$$(7) \quad C \log_2 k \geq 2(\log_2 C + \log_2 k + \log_2 \log_2 k)$$

and this is the case if $C - 4 \geq 2 \log_2 C$, so we may set $C = 12$. Thus (2) holds if $\delta \geq 18k \log_2 k$. $\square$

If we denote by $l(k)$ the least positive integer δ such that (2) holds, then we have the following particular values: $l(2) = 11$, $l(3) = 27$, $l(4) = 46$.

THEOREM 9. $\text{pd}(C_\omega) = \text{pd}^*(C_\omega) = 2$.

Proof. An upper bound of 2 can be discerned from the sequence

$$\cdots 76551142243341235668 \cdots$$

A lower bound can be obtained by Hall's Theorem. $\square$

5. Computational complexity. In this section we study the computational complexity of a problem associated with the partition independence number of a graph: the problem of computing a safe sequence of read operations.

Storing $\text{pi}(G)$ copies of each unit of data is a sufficient condition to insure that no matter how these copies are arranged, there is a sequence of read operations that allows all of the data to be retrieved in spite of the possible occurrence of destructive read operations. Since it is not also a necessary condition, we may wish to consider the problem of determining, for a given storage pattern, whether a safe sequence of read operations exists. This problem is defined as follows.

SAFE READING

Input: A vertex colored graph $G = (V, E)$.

Question: Is there a set of representatives $V' \subseteq V$ of the color classes that is an independent set in G ?

THEOREM 10. SAFE READING is NP-complete, even for input for which the underlying graph is a forest.

Proof. The problem is in NP since it is easy to check, for a given set $V' \subseteq V$, whether V' is both a set of representatives and independent. To show that it is NP-hard we reduce from 3SAT. Given a Boolean expression E with variables $v_1, \dots, v_r$ and clauses $c_1, \dots, c_n$ define the graph G_E to have the vertex set

$$\{v_1^+, v_1^-, \dots, v_r^+, v_r^-, c_1^1, c_1^2, c_1^3, \dots, c_n^1, c_n^2, c_n^3\}$$

with an edge from v_i^+ to c_j^s if and only if the s th literal in the clause c_j is v_i , and, similarly, with an edge from v_i^- to c_j^s if and only if the literal $\bar{v}_i$ is the s th literal in c_j . The color classes of G_E are $\{v_i^+, v_i^-\}, \dots, \{v_r^+, v_r^-\}, \{c_1^1, c_1^2, c_1^3\}, \dots, \{c_n^1, c_n^2, c_n^3\}$. Any independent set of representatives V' for the color classes of G_E must include exactly one of v_i^+, v_i^- for $i = 1, \dots, r$. This determines a truth assignment $\tau_{V'}$ by rule $\tau_{V'}(v_i) = \text{true}$ if and only if $v_i^+ \in V'$.

If $\tau_{V'}$ did not satisfy E then there would be some clause c_j with associated "clause" vertices c_j^1, c_j^2, c_j^3 , each adjacent in G_E to a vertex in V' , contradicting that V' contains a representative from the color class $\{c_j^1, c_j^2, c_j^3\}$.

Conversely, if τ is a satisfying truth assignment for E , then an independent set of representatives V'_τ is described by choosing v_i^+ to be an element of V'_τ if $\tau(v_i) = \text{false}$ and choosing v_i^- otherwise. Since τ is satisfying, at least one vertex in each color class of “clause” vertices can be chosen that is not adjacent to any of the “vertex” vertices chosen. $\square$

With a little more trouble one can show that SAFE READING remains NP-complete even for input for which the underlying graph is a path. A similar result holds for the problem of determining whether a dominating transversal exists in a vertex-colored path.

6. Open problems. The parameters that we have explored in this paper have a “Ramsey-like” character, and they are difficult to compute or to bound efficiently, even for very simple graphs. One wonders, for example, what $\text{pi}^*(\omega C_5)$ might be or whether determining if $\text{pi}^*(\omega G) \leq k$ is decidable. Is it possible that $\text{pi}^*(\omega G)$ is within some constant of the maximum degree? It seems that many simple questions concerning these parameters are quite thorny.

Acknowledgments. The author thanks Hal Sudborough for suggesting a possible application, and Brian Alspach, Frank Hsu, Janos Komlos, and Mirek Truszczynski for helpful conversations.

Note added in proof. It has come to our attention that Noga Alon earlier obtained Theorems 1, 4, and 5 with an improved bound of $2^{\Delta+1}$ for Theorem 5. A proof of Theorem 1 appears in [A1]. A further result is that Theorem 8 fails if “ δ -regular” is replaced with “minimum degree δ .”

REFERENCES

- [A1] N. ALON, *The linear arboricity of graphs*, Israel J. Math., 62 (1988), pp. 311–325.
- [Bo] B. BOLLOBAS, *Extremal Graph Theory*, Academic Press, New York, 1978.
- [Sc] J. SCHÖNHEIM, *Private communication*, 1988.
- [Sp] J. SPENCER, *Ten Lectures on the Probabilistic Method*, CBMS-NSF Regional Conference Series 52, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1987.

SINGULARITY ANALYSIS OF GENERATING FUNCTIONS*

PHILIPPE FLAJOLET† AND ANDREW ODLYZKO‡

Abstract. This work presents a class of methods by which one can translate, on a term-by-term basis, an asymptotic expansion of a function around a dominant singularity into a corresponding asymptotic expansion for the Taylor coefficients of the function. This approach is based on contour integration using Cauchy's formula and Hankel-like contours. It constitutes an alternative to either Darboux's method or Tauberian theorems that appears to be well suited to combinatorial enumerations, and a few applications in this area are outlined.

Key words. asymptotic analysis, generating functions, combinatorial enumeration

AMS(MOS) subject classifications. 05, 40, 68

1. Introduction. Several applications in analysis, especially combinatorics, necessitate determining the asymptotic order of growth of coefficients of a function that is analytic at the origin. It has been recognized for a long time that the function's dominant singularities (the ones of smallest modulus) contain a great deal of information on the coefficients. This paper describes a very general method based on earlier works of ours (Odlyzko [1982], Flajolet and Odlyzko [1982]) that applies to functions of "moderate" variation. We restrict our attention to functions with a unique dominant singularity. (Functions with a *finite* number of singularities on their circle of convergence can also be treated by a direct extension of our methods, using composite integration contours.) By normalization, we may always assume that the dominant singularity occurs at $z = 1$, and we consider functions that satisfy, for some arbitrary real number α ,

$$f(z) \approx (1 - z)^\alpha \quad (z \rightarrow 1).$$

Let f_n be a sequence of numbers, with a generating function $f(z) = \sum_{n \geq 0} f_n z^n$ that is analytic at the origin. In nonelementary cases that we encounter in combinatorial analysis, the f_n are only accessible via $f(z)$, and $f(z)$ itself is either explicitly defined by a closed-form expression or implicitly specified as the solution to a functional equation. (See, for instance, Comtet [1974], Goulden and Jackson [1983], Stanley [1986].) The problem is thus to obtain estimates for f_n from whatever analytic information is available on $f(z)$.

For example, the generating function of 2-regular graphs (Comtet [1974]) is

$$f(z) = \frac{e^{-z/2 - z^2/4}}{\sqrt{1 - z}},$$

and the expansion of $f(z)$ at $z = 1$,

$$(1.1a) \quad f(z) = e^{-3/4} \left[\frac{1}{\sqrt{1 - z}} + \sqrt{1 - z} + \cdots \right],$$

* Received by the editors April 4, 1988; accepted for publication (in revised form) April 19, 1989.

† Institut National de Recherche en Informatique et en Automatique, Rocquencourt, 78150 Le Chesnay, France. The work of this author was done in part while visiting Stanford University under support from National Science Foundation grant CCR-8610181 and Office of Naval Research grant N-00014-87-K-052.

‡ AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

has a matching expansion for coefficients f_n as $n \rightarrow \infty$,

$$(1.1b) \quad \begin{aligned} f_n &\sim e^{-3/4} \left[\binom{n-\frac{1}{2}}{n} + \binom{n-\frac{3}{2}}{n} + \cdots \right] \\ &\sim e^{-3/4} \left[\frac{1}{\sqrt{\pi n}} + \frac{c}{\sqrt{n^3}} + \cdots \right]. \end{aligned}$$

Darboux's method is one way of achieving the term-by-term transition from (1.1a) to (1.1b). (It is succinctly described, along with Tauberian methods, in § 5.) The *smoothness condition* ensuring its validity is that the expansion (1.1a) can be pushed until an error term, sufficiently differentiable on the circle $|z| = 1$, is obtained (Henrici [1977], Bender [1974], Comtet [1974], Olver [1974]). We present here a general method that ensures the validity of an expansion like (1.1b), using only *order-of-growth* information on the remainder terms in the asymptotic expansion of $f(z)$ in a suitable domain of the complex plane. For instance, under suitable analytic conditions, an expansion

$$(1.2a) \quad \begin{aligned} f(z) &\sim \frac{1}{\sqrt{1-z}} \left(\frac{c_0}{\sqrt{\log(1-z)^{-1}}} + \frac{c_1}{\log(1-z)^{-1}} \right. \\ &\quad \left. + \frac{c_2}{\sqrt{\log^3(1-z)^{-1}}} + \cdots \right) \quad (z \rightarrow 1) \end{aligned}$$

“transfers” to coefficients as

$$(1.2b) \quad f_n \sim \frac{1}{\sqrt{\pi n}} \left(\frac{c_0}{\sqrt{\log n}} + \frac{c_1}{\log n} + \frac{c'_2}{\sqrt{\log^3 n}} + \cdots \right) \quad (n \rightarrow \infty)$$

(where $c'_2, c'_3, \dots$ depend only on $c_0, c_1, \dots$), but there is no way of achieving this by Darboux's method, since a remainder term introduced in (1.2a) cannot be differentiable at $z = 1$ unless the expansion is trivial.

More generally, we provide sufficient conditions for the validity of the implications

$$(T1) \quad f(z) = O(g(z)) \Rightarrow f_n = O(g_n)$$

$$(T2) \quad f(z) = o(g(z)) \Rightarrow f_n = o(g_n)$$

$$(T3) \quad f(z) \sim g(z) \Rightarrow f_n \sim g_n.$$

The conditions are that $g(z)$ should belong to a well-defined asymptotic scale $\mathcal{S}$, and that the asymptotic form for f should be valid in a suitable domain of the complex plane, which usually requires *analytic continuation* of $f(z)$ outside its circle of convergence. The asymptotic scale $\mathcal{S}$ we consider here contains functions of z of the form

$$g(z) = A \left(\frac{1}{1-z} \right) \quad \text{with } A(u) \sim u^\beta (\log u)^\gamma (\log \log u)^\delta \quad \text{as } (u \rightarrow \infty)$$

whose coefficients will be later proved to satisfy

$$g_n \sim \frac{1}{\Gamma(\beta)} \frac{A(n)}{n}.$$

We observe from this last equation that larger singular functions at $z = 1$ have larger Taylor coefficients. Thus, again under suitable conditions on $f(z)$, we are justified in applying the transfer principle from the first equation to the second equation in the pair

$$\begin{aligned} f(z) &= h_0(z) + h_1(z) + \cdots + h_k(z) + O(g(z)) \\ &\quad \text{with } h_0(z) \gg \cdots \gg h_k(z) \gg g(z) \quad (z \rightarrow 1) \\ \text{(T0)} \quad f_n &= h_{0,n} + h_{1,n} + \cdots + h_{k,n} + O(g_n) \\ &\quad \text{with } h_{0,n} \gg \cdots \gg h_{k,n} \gg g_n \quad (n \rightarrow \infty) \end{aligned}$$

when the $h_i(z)$ and $g(z)$ are in $\mathcal{S}$. Asymptotic expansions in a large class translate term-by-term in a direct way.

From now on, we shall call transfer theorems of types **(T1)**, **(T2)**, **(T3)**, and **(T0)** by the more suggestive names of O -transfers, o -transfers, $\sim$ -transfers, and Σ -transfers (read as “big O ,” “little o ,” “sim” and “sigma” transfers!). The most basic transfers are O -transfers. A refinement of the proof of a O -transfer will usually lead to a o -transfer. As a direct consequence of o -transfers, we obtain $\sim$ -transfers since

$$f(z) \sim g(z) \quad \text{is equivalent to} \quad f(z) = g(z) + o(g(z)).$$

As indicated in the previous paragraph, Σ -transfers follow directly from O -transfers for expansions with an $O(\cdot)$ remainder term as in **(T0)** and there is an obvious analogue for o -transfers where we have an $o(\cdot)$ remainder term. A side issue to be considered is the determination of coefficients of standard singular functions (here, the scale $\mathcal{S}$). Fortunately, that task can itself be carried out by methods rather similar to the proofs of transfer theorems.

Several of our statements are inspired by Tauberian theorems, most notably the Hardy–Littlewood theorem (Hardy and Littlewood [1914]), though our analysis is quite different since it is based on contour integration. It is clearly less deep, but it has the advantage of great technical simplicity. Tauberian theorems impose no condition on the function, but they require a priori “side conditions” on the coefficients (positivity, monotonicity, etc.). (See Titchmarsh [1939], Hardy [1949], Postnikov [1980].) Our method imposes conditions on the function in the complex domain, but no a priori condition on the coefficients. That approach is therefore quite adequate for obtaining expansions of the Σ -type, for which Tauberian methods tend to be of little help (side conditions are hard to establish on error terms). Furthermore, in the context of combinatorial enumerations, large classes of generating functions are expected to be analytically continuable since they are obtained as combinations of analytic functionals applied to entire functions. In that context, our conditions are seldom a limitation.

The paper is organized as follows. In § 2, we start with a restricted scale $\mathcal{S}_0$ that contains only functions of the form $(1 - z)^\alpha$, where α is *any* (positive or negative) real number. The O -transfer theorem (Theorem 1) that covers all values of α requires analytic continuation of $f(z)$ in an angular domain outside its circle of convergence and validity of the asymptotic expansions there. It is proved—as are all other results in this paper—by choosing a suitable contour (reminiscent of Hankel contours) in Cauchy’s integral formula for coefficients of analytic functions, and “integral splitting.” A modification of the proof gives us o -transfers (Corollary 1), from which we deduce $\sim$ -transfers (Corollary 2) and a variety of Σ -transfers of which Corollary 3 is only a typical example. These results, though later generalized, are treated in some detail, as they serve to introduce basic techniques without unnecessary complications.

Our next objective, in § 3, is to extend theorems to the larger class $\mathcal{S}$ that has logarithms and iterated logarithms. We first establish (Theorem 2 and Corollary 4) general O - and o -transfer results for that class. More precise asymptotic estimates require us to characterize the growth of Taylor coefficients of functions that belong to the scale $\mathcal{S}$ (Theorem 3). This is done by using Hankel contours and modifications of the integral splitting techniques employed in our earlier proofs. Of the variety of conceivable $\sim$ -transfers and Σ -transfers, we state only Corollaries 5 and 6, which correspond to functions with a descending expansion involving powers of logarithms or iterated logarithms.

Section 4 discusses various possible extensions. When $\alpha < -1$, so that $f(z) \approx (1-z)^\alpha$ is “large” at its singularity, the conditions on the function to be analyzed can be weakened somewhat (Theorem 4). Also, a superset of $\mathcal{S}$ that includes functions of “slow variation” towards infinity is shown to be amenable to transfer methods.

Section 5 is a brief discussion of the relation to Darboux’s method and Tauberian theorems. We finally conclude, in § 6, by sketching the transfer part of a few applications to combinatorial enumerations and the analysis of algorithms.

Relation to other works. The Hankel contours are classical tools in the study of the gamma and zeta functions (Whittaker and Watson [1927]). They appear to be well suited to extract asymptotic information on coefficients of analytic functions. They have been used in combinatorial applications in Odlyzko [1982] and Flajolet and Odlyzko [1982], and in another context (enumeration of polynomials over finite fields) by Car [1982], [1984]. Some results similar to ours, but requiring different analytic conditions, have been derived by Wong and Wyman [1974].

In analytic number theory, Hankel contours are useful in the study of Dirichlet series with algebraic singularities by means of Perron’s formula, a typical example being the study of the coefficient of $[n^{-s}]$ in $(\zeta(s))^{1/2}$ (Selberg [1954], Hardy [1940, p. 62]). They are also useful in the inversion of Laplace or Mellin transforms with algebraic or logarithmic singularities (Doetsch [1955, p. 158]). Techniques of §§ 2 and 3 also bear some resemblance to the Watson–Doetsch lemma for Laplace transforms (Henrici [1977, p. 389]).

2. A basic transfer theorem. We let $\mathcal{S}_0$ be the class of functions $g_\alpha(z) = K(1-z)^\alpha$ for α a real number and K a constant. The Taylor coefficients of any member of $\mathcal{S}_0$ are known both exactly¹

$$(2.1) \quad [z^n](1-z)^\alpha = \binom{n-\alpha-1}{n} = \frac{\Gamma(n-\alpha)}{\Gamma(-\alpha)\Gamma(n+1)}$$

and asymptotically from Stirling’s formula ($\alpha \notin \{0, 1, 2, \dots\}$):

$$(2.2) \quad [z^n](1-z)^\alpha \sim \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} \left[1 + \frac{\alpha(\alpha+1)}{2n} + \frac{\alpha(\alpha+1)(\alpha+2)(3\alpha+1)}{24n^2} + \frac{\alpha^2(\alpha+1)^2(\alpha+2)(\alpha+3)}{48n^3} + \frac{\alpha(\alpha+1)(\alpha+2)(\alpha+3)(\alpha+4)(15\alpha^3+30\alpha^2+5\alpha-2)}{5760n^4} + \dots \right].$$

¹ We let, as usual, $[z^n]f(z)$ denote the coefficient of z^n in the Taylor expansion of $f(z)$: If $f(z) = \sum_n f_n z^n$, then $[z^n]f(z) = f_n$.

Thus the binomial coefficients (2.1), as well as their main asymptotic equivalents in (2.2), form an asymptotic scale. There is in fact a general form of (2.2).

PROPOSITION 1. *The binomial coefficients expressing $[z^n](1-z)^\alpha$ have an asymptotic expansion as $n \rightarrow \infty$,*

$$(2.3) \quad [z^n](1-z)^\alpha \sim \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} \left(1 + \sum_{k \geq 1} \frac{e_k^{(\alpha)}}{n^k} \right), \quad \alpha \notin \{0, 1, 2, \dots\},$$

where

$$(2.4) \quad e_k^{(\alpha)} = \sum_{l=k}^{2k} (-1)^l \lambda_{k,l} (\alpha+1)(\alpha+2) \cdots (\alpha+l)$$

$$\text{with } \sum_{k,l \geq 0} \lambda_{k,l} v^k t^l = e^t (1+vt)^{-1-1/v}.$$

Proposition 1, although it would probably follow by close inspection of Stirling's formula, is most easily proved by techniques introduced in § 3, so that we delay the proof until then. We also observe, incidentally, that in (2.1)–(2.3) α may be complex: If $\alpha = \sigma + it$, we have

$$[z^n](1-z)^\alpha \sim \frac{n^{-\sigma-1}}{\Gamma(-\sigma-it)} [\cos(t \log n) - i \sin(t \log n)].$$

In that case, the main term in (2.2), (2.3) is of order $n^{-\sigma-1}$ and it is multiplied by a periodic function of $\log n$.

We now propose to prove a transfer condition of the O -type. We give the proof in some detail for two reasons: first, the implied constant in the O 's are “constructive” and tight, a fact of independent interest; second, it serves as a guiding pattern for later deriving a variety of transfer conditions. We let $\Delta \equiv \Delta(\phi, \eta)$ denote the closed domain

$$(2.5) \quad \Delta(\phi, \eta) = \{z / |z| \leq 1 + \eta, |\operatorname{Arg}(z-1)| \geq \phi\},$$

where we take $\eta > 0$ and $0 < \phi < (\pi/2)$. This domain has the form of an indented disk depicted on Fig. 1(a).

THEOREM 1. *Assume that, with the sole exception of the singularity $z = 1$, $f(z)$ is analytic in the domain $\Delta = \Delta(\phi, \eta)$, where $\eta > 0$ and $0 < \phi < (\pi/2)$. Assume further that as z tends to 1 in Δ ,*

$$(2.6a) \quad f(z) = O(|1-z|^\alpha),$$

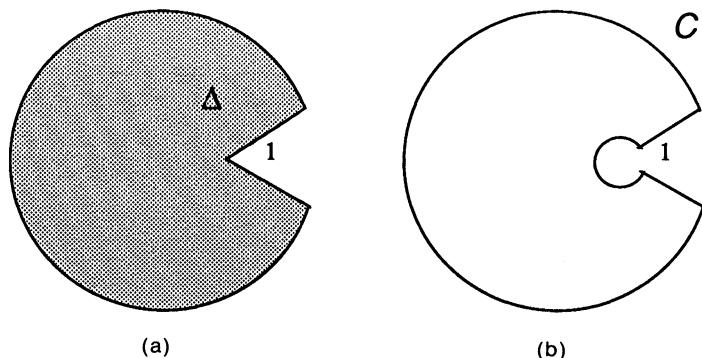


FIG. 1. (a) The domain $\Delta(\phi, \eta)$. (b) The contour $\mathcal{C}$ used in the proof of Theorem 1.

for some real number α . Then the n th Taylor coefficient of $f(z)$ satisfies

$$(2.6b) \quad f_n = [z^n]f(z) = O(n^{-\alpha-1}).$$

Proof. Since the modulus of $(1-z)^\alpha$ is bounded below by a constant > 0 in any compact set in Δ that does not contain 1, and $f(z)$ is analytic in such a set, the *local* condition (2.6a) of the theorem is equivalent to assuming that for some constant $K > 0$, we have in the *whole* of Δ with the possible exception of $z = 1$,

$$(2.7) \quad |f(z)| < K|1-z|^\alpha.$$

In the derivation, we assume that $n \geq 2|\alpha| + 4$. This technical constraint is sufficient to ensure the validity of estimate (2.9) as well as the existence of the integrals appearing in (2.12), (2.13). We start from Cauchy's formula

$$(2.8) \quad f_n = \frac{1}{2i\pi} \int_{O^+} f(z) \frac{dz}{z^{n+1}},$$

where O^+ is any positively oriented contour in Δ that encloses the origin, and we choose the (positively oriented) contour $\mathcal{C} = \gamma_1 \cup \gamma_2 \cup \gamma_3 \cup \gamma_4$ depicted in Fig. 1(b), with

$$\begin{aligned} \gamma_1 &= \left\{ z / |z-1| = \frac{1}{n}, |\text{Arg}(z-1)| \geq \phi \right\} \\ \gamma_2 &= \left\{ z / \frac{1}{n} \leq |z-1|, |z| \leq 1+\eta, \text{Arg}(z-1) = \phi \right\} \\ \gamma_3 &= \left\{ z / |z-1| = 1+\eta, |\text{Arg}(z-1)| \geq \phi \right\} \\ \gamma_4 &= \left\{ z / \frac{1}{n} \leq |z-1|, |z| \leq 1+\eta, \text{Arg}(z-1) = -\phi \right\}. \end{aligned}$$

We proceed to evaluate the contributions to f_n due to each of the γ_j separately. So, we define

$$f_n^{(j)} = \frac{1}{2\pi} \int_{\gamma_j} |f(z)| \frac{|dz|}{|z|^{n+1}},$$

and we have $|f_n| \leq f_n^{(1)} + f_n^{(2)} + f_n^{(3)} + f_n^{(4)}$.

1. *Smaller circle.* From (2.7), using trivial bounds on Cauchy's integral, we find as soon as $n \geq 4$,

$$(2.9) \quad \begin{aligned} f_n^{(1)} &\leq \left(\frac{1}{2\pi}\right) \cdot K \left(\frac{1}{n}\right)^\alpha \cdot \left(1 - \frac{1}{n}\right)^{-n-1} \cdot \left(\frac{2\pi}{n}\right) \\ &\leq 5 \cdot (Kn^{-\alpha-1}). \end{aligned}$$

2. *Rectilinear part.* We next turn to the evaluation of $f_n^{(2)}$. By obvious symmetry considerations, the same bound will hold for $f_n^{(4)}$. We set $\omega = e^{i\phi}$ and perform the change of variable: $z = 1 + (\omega t/n)$. The definition of $f_n^{(2)}$ gives

$$(2.10) \quad \begin{aligned} f_n^{(2)} &\leq \frac{1}{2\pi} \int_1^{En} K \left(\frac{t}{n}\right)^\alpha \left|1 + \frac{\omega t}{n}\right|^{-n-1} \frac{dt}{n} \\ &\leq (Kn^{-\alpha-1}) \cdot \frac{1}{2\pi} \int_1^\infty t^\alpha \left|1 + \frac{\omega t}{n}\right|^{-n-1} dt. \end{aligned}$$

Here E is defined so that γ_2 and γ_3 join: E is the positive root of $|1 + Ee^{i\phi}| = 1 + \eta$. We need to prove that the last integral in (2.10) is bounded above independently of n . First observe that

$$(2.11) \quad \left| 1 + \frac{\omega t}{n} \right| \geq 1 + \Re \left(\frac{\omega t}{n} \right) = 1 + \frac{t}{n} \cos \phi.$$

Thus, from (2.10) and (2.11), we find

$$(2.12) \quad f_n^{(2)} < \frac{J_n}{2\pi} (Kn^{-\alpha-1}) \quad \text{where } J_n = \int_1^\infty t^\alpha \left(1 + \frac{t \cos \phi}{n} \right)^{-n} dt.$$

It is now easy to see that as $n \rightarrow \infty$,

$$J_n \rightarrow \int_1^\infty t^\alpha e^{-t \cos \phi} dt$$

and hence all the J_n are bounded above by some constant which depends only on α and ϕ . In fact, for positive λ , function $(1 + \lambda/n)^{-n}$ is a monotone decreasing function of n . In summary, we have proved that

$$(2.13) \quad f_n^{(2)} < \frac{J(\alpha, \phi)}{2\pi} (Kn^{-\alpha-1}) \quad \text{where } J(\alpha, \phi) = \int_1^\infty t^\alpha \left(1 + \frac{t \cos \phi}{\nu} \right)^{-\nu} dt$$

$$\text{and } \nu = 2|\alpha| + 4.$$

The fact that ϕ is strictly less than $\pi/2$, whence $\cos \phi > 0$, is obviously crucial to this part of the analysis.

3. *Larger circle.* The majorization of $f_n^{(3)}$ gives an exponentially decreasing term:

$$(2.14) \quad \begin{aligned} f_n^{(3)} &< \frac{1}{2\pi} \cdot K\eta^\alpha \cdot (1 + \eta)^{-n-1} \cdot (2\pi(1 + \eta)) \\ &< K \frac{\eta^\alpha}{(1 + \eta)^n}. \end{aligned}$$

4. Collecting the results of (2.9), (2.13), and (2.14), we have proved, for all $n \geq 2|\alpha| + 4$:

$$(2.15) \quad f_n < (Kn^{-\alpha-1}) \left[5 + \frac{J(\alpha, \phi)}{\pi} + \frac{\eta^\alpha}{(1 + \eta)^n} n^{\alpha+1} \right].$$

There is an effectively computable constant n_1 (only depending on α and η), such that, for all $n \geq n_1$,

$$(2.16) \quad \frac{\eta^\alpha}{(1 + \eta)^n} n^{\alpha+1} < 1.$$

Thus, from (2.15), (2.16), we obtain our main bound,

$$(2.17) \quad f_n < (Kn^{-\alpha-1}) \cdot \left[6 + \frac{J(\alpha, \phi)}{\pi} \right] \quad \text{for all } n \geq n_0,$$

where $n_0 = \max(n_1, 2|\alpha| + 4)$. Equation (2.17) is a stronger form of the statement of the theorem. $\square$

The idea of a contour that “goes away” from the singularity at an angle has the essential feature of introducing, in Cauchy’s integral, a “kernel” (z^{-n} or $(1 + t/n)^{-n}$) that decreases very fast along the contour, so that it captures the dominant contribution from an immediate vicinity of the singularity.

The proof techniques of Theorem 1, slightly modified, will give us a transfer result of o -type, from which we immediately deduce $\sim$ - and Σ -transfers. The results that follow are not strictly speaking corollaries of Theorem 1, but rather of the line of proof taken there.

COROLLARY 1. Assume that $f(z)$ is analytic in $\Delta \setminus \{1\}$, and that as $z \rightarrow 1$ in Δ ,

$$f(z) = o((1 - z)^\alpha).$$

Then, as $n \rightarrow \infty$,

$$f_n = o(n^{-\alpha-1}).$$

Proof. The proof is an “ ε - δ exercise.” We use the same contour $\mathcal{C}$ as in Theorem 1. Observe again that there exists a $K > 0$ such that in the whole of Δ , $|f(z)| < K|1 - z|^\alpha$. By the o hypothesis on $f(z)$, for each $\varepsilon > 0$, there exists a $\delta = \delta(\varepsilon) > 0$ such that for z in Δ ,

$$|z - 1| < \delta \Rightarrow |f(z)| < \varepsilon |1 - z|^\alpha.$$

We need to prove, with some fixed constant $K' > 0$, that for any $\varepsilon > 0$, there exists an $n_0 = n_0(\varepsilon)$ such that

$$|f_n| < \varepsilon K' n^{-\alpha-1} \quad \text{whenever } n \geq n_0(\varepsilon).$$

In the following derivations, the constants also depend on α , ϕ , and η , but we shall only indicate the dependence on ε . We choose a fixed (but arbitrary) ε , with its associated $\delta = \delta(\varepsilon)$.

1. *Smaller circle.* For the part γ_1 of the contour, we first choose $n_1 = n_1(\varepsilon)$ such that $1/n_1 \leq \delta(\varepsilon)$. This choice ensures that part γ_1 of the contour is inside a domain where $|f(z)| < \varepsilon |1 - z|^\alpha$, and thus for $n > n_1$, as in (2.9),

$$(2.18) \quad f_n^{(1)} < 5(\varepsilon n^{-\alpha-1}).$$

2. *Rectilinear part.* Following the lines of (2.10), (2.11), with $z = 1 + \omega t/n$, we have

$$(2.19) \quad f_n^{(2)} < \frac{1}{2\pi} \int_1^{En} \left| f\left(1 + \frac{\omega t}{n}\right) \right| \left(1 + \frac{t \cos \phi}{n}\right)^{-n} \frac{dt}{n}.$$

We now decompose the integral in (2.19), and set $f_n^{(2)} = f_n^{(21)} + f_n^{(22)}$, where

$$f_n^{(21)} = \frac{1}{2\pi} \int_1^{\log^2 n} \quad \text{and} \quad f_n^{(22)} = \frac{1}{2\pi} \int_{\log^2 n}^{En}.$$

We can choose $n_{21} = n_{21}(\varepsilon)$ such that for all $n \geq n_{21}$, we have $\log^2 n/n < \delta(\varepsilon)$, and then $1 + \omega t/n$ is in the “epsilon region” of f . Thus

$$(2.20) \quad \begin{aligned} f_n^{(21)} &< \frac{1}{2\pi} \int_1^{\log^2 n} \varepsilon \left(\frac{t}{n}\right)^\alpha \left(1 + \frac{t \cos \phi}{n}\right)^{-n} \frac{dt}{n} \\ &< (\varepsilon n^{-\alpha-1}) \cdot \frac{1}{2\pi} \int_1^\infty t^\alpha \left(1 + \frac{t \cos \phi}{n}\right)^{-n} dt. \end{aligned}$$

From the argument used for (2.13), we find

$$(2.21) \quad f_n^{(21)} < (\varepsilon n^{-\alpha-1}) \frac{J(\alpha, \phi)}{2\pi}.$$

By similar devices, we get for $f_n^{(22)}$ the bound

$$(2.22) \quad f_n^{(22)} < (Kn^{-\alpha-1}) \frac{1}{2\pi} \int_{\log^2 n}^{En} t^\alpha \left(1 + \frac{t \cos \phi}{n}\right)^{-n} dt.$$

The integrand in (2.22) is already exponentially small at $t = \log^2 n$. Without loss of generality, we may assume (by taking η small enough) that $E < 1/(4 \cos \phi)$. This guarantees $u = t \cos \phi / n < \frac{1}{4}$, so that, in the given range of values of t , $\log(1 + u) > u/2$ and

$$\left(1 + \frac{t \cos \phi}{n}\right)^{-n} < \exp\left(-\frac{\cos \phi}{2} \log^2 n\right).$$

Thus the integral in (2.22) is, say, $< n^{-2}$ for $n \geq n_{22}$, and so

$$(2.23) \quad f_n^{(22)} < \frac{1}{n^2} (Kn^{-\alpha-1}).$$

3. *Larger circle.* Finally, for $f_n^{(3)}$, we just use the previously established bound (2.14), which we repeat here,

$$(2.24) \quad f_n^{(3)} < K \frac{\eta^\alpha}{(1 + \eta)^n}.$$

4. Collecting the results of (2.18), (2.21), (2.23), and (2.24), we find that, for all $n \geq n_4(\varepsilon)$, where $n_4 = \max(n_1, n_{21}, n_{22})$,

$$(2.25) \quad n^{\alpha+1} f_n < \varepsilon \left[5 + \frac{J(\alpha, \phi)}{\pi} \right] + K \left[\frac{2}{n^2} + \frac{\eta^\alpha}{(1 + \eta)^n} n^{\alpha+1} \right].$$

We can obviously choose an $n_5 = n_5(\varepsilon)$ such that for all $n \geq n_5$,

$$K \left(\frac{2}{n^2} + \frac{\eta^\alpha}{(1 + \eta)^n} n^{\alpha+1} \right) < \varepsilon.$$

Then for $n \geq n_0$ where $n_0 = n_0(\varepsilon) = \max(n_4, n_5)$, we obtain

$$(2.26) \quad n^{\alpha+1} f_n < \varepsilon \cdot \left[6 + \frac{J(\alpha, \phi)}{\pi} \right],$$

and (2.26) yields our corollary. $\square$

We can now conclude this section by stating a $\sim$ -transfer and a Σ -transfer.

COROLLARY 2. Assume that $f(z)$ is analytic in $\Delta \setminus \{1\}$, and that as $z \rightarrow 1$ in Δ ,

$$f(z) \sim K(1 - z)^\alpha.$$

Then, as $n \rightarrow \infty$: (i) If $\alpha \notin \{0, 1, 2, \dots\}$,

$$f_n \sim \frac{K}{\Gamma(-\alpha)} n^{-\alpha-1};$$

(ii) If α is a nonnegative integer, then

$$f_n = o(n^{-\alpha-1}).$$

Proof. It suffices to apply Corollary 1 to the expansion

$$f(z) = K(1-z)^\alpha + o((1-z)^\alpha). \quad \square$$

COROLLARY 3. Assume that $f(z)$ is analytic in $\Delta \setminus \{1\}$, and that as $z \rightarrow 1$ in Δ ,

$$(2.27a) \quad f(z) = \sum_{j=0}^m c_j (1-z)^{\alpha_j} + O((1-z)^A)$$

where $\Re(\alpha_0) \leq \Re(\alpha_1) \leq \dots \leq \Re(\alpha_m) < A$. Then as $n \rightarrow \infty$,

$$(2.27b) \quad f_n = \sum_{j=0}^m c_j \binom{n - \alpha_j - 1}{n} + O(n^{-A-1}).$$

Proof. The proof is a direct consequence of Theorem 1. $\square$

By Proposition 1, expansion (2.27b) can in turn be converted into another asymptotic expansion

$$(2.27c) \quad f_n = \sum_{j=0}^{m'} c'_j n^{-\alpha'_j - 1} + O(n^{-A-1}),$$

where the α'_j belong to $\{\alpha_0, \dots, \alpha_m\} + \{0, 1, 2, 3, \dots\}$ and satisfy $\Re(\alpha'_0) \leq \dots \leq \Re(\alpha'_{m'}) < A$. Obviously, the method applies to a large class of asymptotic expansions by “subtracting singularities.” For instance, from

$$(2.28a) \quad f(z) = \log \frac{1}{1-z} + c_0 + c_1(1-z)^{1/4} + c_2(1-z)^{1/2} + o((1-z)^{1/2}),$$

we observe that $f(z) - \log(1-z)^{-1} = c_0 + c_1(1-z)^{1/4} + \dots$, and derive

$$(2.28b) \quad f_n = \frac{1}{n} \left[1 + \frac{c_1}{\Gamma(-\frac{1}{4})} n^{-1/4} + \frac{c_2}{\Gamma(-\frac{1}{2})} n^{-1/2} + o(n^{-1/2}) \right].$$

3. A general asymptotic scale. The approach that gave us transfer results for functions of type $(1-z)^\alpha$ (the asymptotic scale $\mathcal{S}_0$) can be easily extended to cover a larger class $\mathcal{S}$ of singular functions, which we take to be of the form

$$(3.1) \quad g(z) = K(1-z)^\alpha \cdot L\left(\frac{1}{1-z}\right) \quad \text{where } L(u) = (\log u)^\gamma (\log \log u)^\delta.$$

Essentially, our previous results hold true provided we add an extra factor of $L(n)$ in asymptotic formulæ for coefficients. It should be clear from the derivations that our results depend on the fact that functions $L(u)$ “vary slowly” towards infinity, so that they behave almost like constants in our proofs: The key property is that $L(u)$ should satisfy

$$\frac{L(\lambda e^{i\theta} u)}{L(u)} \rightarrow 1, \quad (u \rightarrow +\infty),$$

in a suitably uniform way for any fixed $\lambda > 0$ and $|\theta| \leq \pi - \phi$ (see § 4 for a more complete discussion).

THEOREM 2. Assume that, with the sole exception of the singularity $z = 1$, $f(z)$ is analytic in the domain $\Delta = \Delta(\phi, \eta)$, where $\eta > 0$ and $0 < \phi < (\pi/2)$. Assume further that as z tends to 1 in Δ ,

$$(3.2a) \quad f(z) = O\left((1-z)^\alpha L\left(\frac{1}{1-z}\right)\right) \quad \text{where } L(u) = (\log u)^\gamma (\log \log u)^\delta,$$

for some real numbers α, γ, δ . Then the n th Taylor coefficient of $f(z)$ satisfies

$$(3.2b) \quad f_n = [z^n]f(z) = O(n^{-\alpha-1}L(n)).$$

Proof. Logarithms are taken with their principal determination. Without loss of generality, we may assume that η is small enough. Evaluation of the various contributions to Cauchy's integral proceeds very much like the evaluation in the proof of Corollary 1.

1. *Smaller circle.* Using trivial bounds, we find with $z = 1 - e^{i\theta}/n$, and θ varying in $[-(\pi - \phi), \pi - \phi]$,

$$f_n^{(1)} = O(n^{-\alpha-1}M_1(n)) \quad \text{where } M_1(n) = \sup_{\theta} |L(ne^{-i\theta})|.$$

It is easy to see that $L(u)$ does not vary much along any arc of a large circle centered at the origin, so that $M_1(n) \sim L(n)$ as $n \rightarrow \infty$, and

$$(3.3) \quad f_n^{(1)} = O(n^{-\alpha-1}L(n)).$$

2. *Rectilinear contour.* We set $z = 1 + \omega t/n$, and use the same splitting as in Corollary 2: $f_n^{(2)} = f_n^{(21)} + f_n^{(22)}$. First, we have

$$(3.4) \quad f_n^{(21)} = O\left(n^{-\alpha-1}M_2(n) \int_1^{\log^2 n} t^\alpha \left(1 + \frac{t \cos \phi}{n}\right)^{-n} dt\right)$$

where $M_2(n) = \sup_{t \in [1, \log^2 n]} L\left(-\frac{n}{\omega t}\right)$.

Again, the "slow variation" of $L(u)$ towards infinity entails that $M_2(n) \sim L(n)$. The integral in (3.4) is $O(1)$, hence

$$(3.5) \quad f_n^{(21)} = O(n^{-\alpha-1}L(n)).$$

We now turn to $f_n^{(22)}$, for which we have

$$(3.6) \quad f_n^{(22)} < \frac{Kn^{-\alpha-1}}{2\pi} \int_{\log^2 n}^{En} t^\alpha \left|L\left(-\frac{n}{\omega t}\right)\right| \left(1 + \frac{t \cos \phi}{n}\right)^{-n} dt.$$

When z is on γ_2 , quantity $u(t) = -n/\omega t$ goes from an area (for $t = En$) where it is $O(1)$ to a neighbourhood of infinity as $e^{-i(\pi-\phi)}\infty$ (for $t = \log^2 n$). Over $u(\gamma_2)$, function $L(u)$ is upperbounded by a linear function of $|u|$. (Actually, it cannot grow faster than $|u|^\varepsilon$). We thus have $|L(u)| < K_1|u| + K_2$, and

$$f_n^{(22)} < \frac{n^{-\alpha-1}}{2\pi} \int_{\log^2 n}^{En} t^\alpha \left(K_1 \left|\frac{n}{t}\right| + K_2\right) \left(1 + \frac{t \cos \phi}{n}\right)^{-n} dt.$$

If we use the crude bound $K_1|n/t| + K_2 = O(n)$, we find,

$$(3.7) \quad f_n^{(22)} = \frac{n^{-\alpha-1}}{2\pi} \cdot O(n) \cdot \int_{\log^2 n}^{En} t^\alpha \left(1 + \frac{t \cos \phi}{n}\right)^{-n} dt.$$

By an argument already encountered in (2.22), (2.23), the integral in (3.7) is $O(1/n^2)$, so that finally

$$(3.8) \quad f_n^{(22)} = O(n^{-\alpha-2}).$$

3. *Larger circle.* Again, we only need to use

$$(3.9) \quad f_n^{(3)} = O((1 + \eta)^{-n}).$$

4. Collecting the bounds from (3.3), (3.5), (3.8), and (3.9), we find

$$(3.10) \quad \begin{aligned} f_n &= O(L(n)n^{-\alpha-1}) + O(n^{-\alpha-2}) + O((1 + \eta)^{-n}) \\ &= O(L(n)n^{-\alpha-1}), \end{aligned}$$

since $1/n$ is always $o(L(n))$ as $n \rightarrow \infty$. $\square$

A slight modification of the proof of Theorem 2 (see also Corollary 2) yields the following corollary.

COROLLARY 4. *Assume that $f(z)$ is analytic in $\Delta(\phi, \eta) \setminus \{1\}$ and that as $z \rightarrow 1$ in Δ*

$$(3.11a) \quad f(z) = o\left((1-z)^\alpha L\left(\frac{1}{1-z}\right)\right) \quad \text{where } L(u) = (\log u)^\gamma (\log \log u)^\delta,$$

Then the n th Taylor coefficient of $f(z)$ satisfies

$$(3.11b) \quad f_n = [z^n]f(z) = o(n^{-\alpha-1}L(n)).$$

In order to proceed further, we need to find detailed asymptotic expansions for coefficients of a set of functions of the form (3.1), thereby generalizing the classical expansion of $(1-z)^\alpha$ that was stated in Proposition 1. There is only a minor technical difficulty, namely that the functions in (3.1) are not in general analytic at the origin, so that we operate with slightly modified functions. It will be recognized that this modification is of no consequence for asymptotic expansions of coefficients. (See the remarks following Corollary 5).

Our proof technique is based on the use of contours of Hankel type for the Gamma function.

THEOREM 3A. *Let α and γ be real or complex numbers, $\alpha, \gamma \notin \{0, 1, 2, \dots\}$. Define the function $f(z)$ by*

$$f(z) = (1-z)^\alpha \left(\frac{1}{z} \log \frac{1}{1-z} \right)^\gamma.$$

Then, the Taylor coefficients of $f(z)$ satisfy

$$f_n = [z^n]f(z) \sim \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} (\log n)^\gamma \left(1 + \sum_{k \geq 1} \frac{e_k^{(\alpha, \gamma)}}{\log^k n} \right),$$

with

$$e_k^{(\alpha, \gamma)} = (-1)^k \binom{\gamma}{k} \Gamma(-\alpha) \frac{d^k}{ds^k} \left(\frac{1}{\Gamma(-s)} \right) \Big|_{s=\alpha}.$$

Proof. Observe that $f(z)$ is analytic in the plane slit along $[1, +\infty]$. We evaluate the Cauchy integral giving coefficients f_n along a contour (see Fig. 2(a)) $\mathcal{C} = \gamma_1 \cup \gamma_2 \cup \gamma_3 \cup \gamma_4$, where γ_3 is an arc of the circle with radius 2, the rest of the contour being an

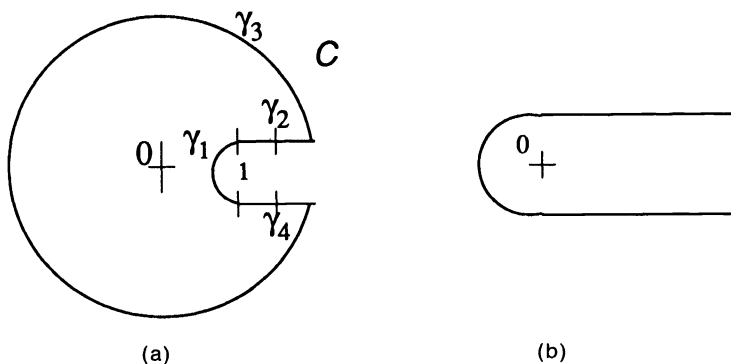


FIG. 2. Various Hankel contours: (a) Contour $\mathcal{C}$ for the proof of Theorem 3. (b) Contour H .

open loop around $[1, 2]$ at distance $1/n$. In symbols,

$$\begin{aligned}\gamma_1 &= \left\{ z = 1 - \frac{t}{n} / t = e^{i\theta}, \theta \in \left[-\frac{\pi}{2}, +\frac{\pi}{2} \right] \right\} \\ \gamma_2 &= \left\{ z = 1 + \frac{t+i}{n} / t \in [0, n] \right\} \\ \gamma_3 &= \left\{ z / |z| = \left(4 + \frac{1}{n^2} \right)^{1/2}, \Re(z) \leq 2 \right\} \\ \gamma_4 &= \left\{ z = 1 + \frac{t-i}{n} / t \in [0, n] \right\}.\end{aligned}$$

We immediately dispose of the contribution to Cauchy's integral due to γ_3 . It satisfies

$$(3.12) \quad f_n^{(3)} = O(2^{-n}),$$

and is thus exponentially small. Let $f_n^{(124)}$ denote the contribution from the rest of the contour, $h_1 = \gamma_1 \cup \gamma_2 \cup \gamma_4$. We perform the change of variable $z = 1 + t/n$, and let H_1 be the contour on which t varies: H_1 is an open loop at distance 1 from the segment $[0, n]$ of the positive real axis. We have

$$(3.13) \quad n^{\alpha+1} f_n^{(124)} = \frac{1}{2i\pi} \int_{H_1} (-t)^\alpha \left(\log \left(-\frac{n}{t} \right) \right)^\gamma \left(1 + \frac{t}{n} \right)^{-n-1-\gamma} dt.$$

Most of the contribution is expected to come from the area where $t \ll n$ because of the fast decreasing $(1 + t/n)^{-n}$ factor. Let H'_1 be the part of the contour H_1 such that $|t| < \log^2 n$. Along $H_1 \setminus H'_1$, the integrand contains an exponentially small factor of the form $e^{-c \log^2 n}$, and is thus negligible. Along H'_1 , by devices that should now be familiar, we may use in (3.13) the approximation $(1 + (t/n))^{-n} \approx e^{-t}$. In this way, we get

$$(3.14) \quad n^{\alpha+1} f_n^{(124)} = \frac{1}{2i\pi} \int_{H'_1} (-t)^\alpha \left(\log \left(-\frac{n}{t} \right) \right)^\gamma e^{-t} dt + O\left(\frac{\log^\gamma n}{n} \right).$$

Still along H'_1 , we have $\log t \ll \log n$, hence we can expand the logarithmic part of the integrand:

$$(3.15) \quad \begin{aligned} \left(\log \left(-\frac{n}{t} \right) \right)^\gamma &= \log^\gamma n \left(1 - \frac{\log(-t)}{\log n} \right)^\gamma \\ &= \log^\gamma n \left(\sum_{k=0}^{m-1} \binom{\gamma}{k} (-1)^k \left(\frac{\log(-t)}{\log n} \right)^k + O \left(\left(\frac{\log(-t)}{\log n} \right)^m \right) \right). \end{aligned}$$

We substitute expansion (3.15) inside the integral of (3.14), and get

$$(3.16) \quad \begin{aligned} \frac{n^{\alpha+1} f_n^{(124)}}{\log^\gamma n} &= \sum_{k=0}^{m-1} \binom{\gamma}{k} (-1)^k \frac{I_k}{\log^k n} + O \left(\frac{1}{\log^m n} \right) \quad \text{where} \\ I_k &= \frac{1}{2i\pi} \int_{H'_1} (-t)^\alpha (\log(-t))^k e^{-t} dt. \end{aligned}$$

We can now extend the rectilinear parts of contour H'_1 towards $+\infty$ (see Fig. 2(b)). This gives us a new contour H , and the process introduces only exponentially small terms in the integral. In this way, we find from (3.16), which is valid for any $m \geq 1$,

$$(3.17) \quad \frac{n^{\alpha+1} f_n^{(124)}}{\log^\gamma n} \sim \sum_{k=0}^{\infty} \binom{\gamma}{k} (-1)^k \frac{G_k}{\log^k n} \quad \text{where } G_k = \frac{1}{2i\pi} \int_H (-t)^\alpha (\log(-t))^k e^{-t} dt.$$

From the bound (3.12) for $f_n^{(3)}$, we see that the same expansion (3.17) holds for f_n . All that remains is to compute the G_k . But, by a familiar integral (Whittaker and Watson [1927, p. 245]), we have

$$G_0 = \frac{1}{2i\pi} \int_H (-t)^\alpha e^{-t} dt = \frac{1}{\Gamma(-\alpha)},$$

and obviously G_k is the k th derivative of G_0 with respect to α . Our proof is now complete. $\square$

THEOREM 3B. *Let α , γ , and δ be complex numbers not in $\{0, 1, 2, \dots\}$. Define the function $f(z)$*

$$f(z) = (1-z)^\alpha \left(\frac{1}{z} \log \frac{1}{1-z} \right)^\gamma \left(\frac{1}{z} \log \left(\frac{1}{z} \log \frac{1}{1-z} \right) \right)^\delta.$$

Then, the Taylor coefficients of $f(z)$ satisfy

$$f_n \sim \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} (\log n)^\gamma (\log \log n)^\delta \left(1 + \sum_{k \geq 1} \frac{e_k(\log \log n)}{(\log n \log \log n)^k} \right),$$

where $e_k(x) = e_k^{(\alpha, \gamma, \delta)}(x)$ is a polynomial of degree k ,

$$\begin{aligned} e_k(x) &= \Gamma(-\alpha) \frac{d^k}{ds^k} \frac{1}{\Gamma(-s)} \Big|_{s=\alpha} \cdot E_k(x) \quad \text{with} \\ \sum_{k=0}^{\infty} E_k(x) u^k &= (1-xu)^\gamma \left(1 - \frac{1}{x} \log(1-xu) \right)^\delta. \end{aligned}$$

Sketch of proof. The proof starts with the analogue of (3.14),

(3.18)

$$n^{\alpha+1} f_n = \frac{1}{2i\pi} \int_{H_1} (-t)^\alpha \left(\log \left(-\frac{n}{t} \right) \right)^\gamma \left(\log \log \left(-\frac{n}{t} \right) \right)^\delta e^{-t} dt + O\left(\frac{(\log n)^\gamma (\log \log n)^\delta}{n} \right)$$

and uses (3.15) together with

$$(3.19) \quad \left(\log \log \left(-\frac{n}{t} \right) \right)^\delta = (\log \log n)^\delta \left(1 + \frac{1}{\log \log n} \log \left(1 - \frac{\log(-t)}{\log n} \right) \right)^\delta.$$

The right-hand side of (3.19) can then be expanded in descending powers of $\log n$ and $\log \log n$, etc. $\square$

The same line of proof now enables us to prove the basic asymptotic estimate for the binomial coefficients already announced in Proposition 1.

Proof of Proposition 1. Using again a variant of (3.13), (3.14), we find that

$$(3.20) \quad [z^n](1-z)^\alpha \sim \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} \cdot \frac{1}{2i\pi} \int_{H_1} (-t)^\alpha \left(1 + \frac{t}{n} \right)^{-n-1} dt$$

and it is the expansion

$$(3.21) \quad \left(1 + \frac{t}{n} \right)^{-n-1} = e^{-t} \left(1 + \frac{t^2 - 2t}{2n} + \frac{6t^4 - 40t^3 + 48}{48n^2} + \dots \right)$$

in descending powers of $1/n$ that provides an explicit form of the $e_k^{(\alpha)}$. $\square$

There are obvious $\sim$ - and Σ -transfer results that follow from these equations. We only cite two simple analogues of Corollary 3.

COROLLARY 5. Assume that $f(z)$ is analytic in $\Delta \setminus \{1\}$, and that as $z \rightarrow 1$ in Δ ,

$$(3.22a) \quad f(z) = (1-z)^\alpha \left(\log \frac{1}{1-z} \right)^\gamma \left[\sum_{j=0}^{m-1} c_j \left(\log \frac{1}{1-z} \right)^{-j} + O\left(\left(\log \frac{1}{1-z} \right)^{-m} \right) \right]$$

for some $\alpha, \gamma \notin \{0, 1, 2, \dots\}$. Then as $n \rightarrow \infty$,

$$(3.22b) \quad f_n = \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} \log^\gamma n \left[\sum_{j=0}^{m-1} c'_j \log^{-j} n + O(\log^{-m} n) \right].$$

Proof. We only need to check that in expansion (3.22a), replacing

$$\left(\log \frac{1}{1-z} \right)^\gamma \quad \text{by} \quad \left(\frac{1}{z} \log \frac{1}{1-z} \right)^\gamma$$

introduces error terms that are of order $(1-z)^{\alpha-1} (\log(1-z))^\gamma$, using the expansion of $1/z$ at $z = 1$. We then conclude by an application of Theorem 3A and Theorem 2. The c'_j are computable from the c_j by the expansion of Theorem 3A. $\square$

COROLLARY 6. Assume that $f(z)$ is analytic in $\Delta \setminus \{1\}$, and that as $z \rightarrow 1$ in Δ ,

(3.23a)

$$f(z) = (1-z)^\alpha \left(\log \frac{1}{1-z} \right)^\gamma \left(\log \log \frac{1}{1-z} \right)^\delta \left[\sum_{j=0}^{m-1} c_j \left(\log \log \frac{1}{1-z} \right)^{-j} + O\left(\left(\log \log \frac{1}{1-z} \right)^{-m} \right) \right]$$

for some $\alpha, \gamma \notin \{0, 1, 2, \dots\}$. Then as $n \rightarrow \infty$,

$$(3.23b) \quad f_n = \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} (\log n)^\gamma (\log \log n)^\delta \left[\sum_{j=0}^{m-1} c_j (\log \log n)^{-j} + O((\log \log n)^{-m}) \right].$$

(The coefficients c_j are the same in both expansions.)

A few historical remarks on the ancestry of Theorem 3 and particular cases not yet explicitly covered are due here. We assumed in the statement of Theorem 3A that neither α nor γ are integers. This leaves three cases to be discussed that can also be treated by our methods.

1. The case where γ is an integer and α not an integer was studied by Jungen [1931]. That important paper² was partly motivated by Hadamard products and singular differential systems with regular singular points. There Jungen makes use of a method introduced by Fröbenius (and classical in the study of differential systems) that consists in starting from the binomial expansion (2.1) of $(1-z)^\alpha$ and differentiating with respect to the parameter α to deduce, with $\gamma = k$ an integer,

$$[z^n](1-z)^\alpha \log^k \frac{1}{1-z} \sim \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} \left[E_0(\log n) + \frac{E_1(\log n)}{n} + \frac{E_2(\log n)}{n^2} + \dots \right],$$

where the E_j are polynomials of degree at most j . In other words, the expansion of our Theorem 3A terminates, and more terms in descending powers of n can be obtained. Another derivation, on which we partly based our proof of Theorem 3, is given by Flajolet and Puech [1986].

2. When α is a positive integer, $1/\Gamma(-\alpha) = 0$, so that the first term in the coefficient expansion of Theorem 3A vanishes and the expansion “jumps” to the next term in descending powers of $\log n$. For instance, we have

$$[z^n] \frac{1}{\log(1-z)^{-1}} \sim \frac{C}{n \log^2 n}.$$

Pólya cites without proof an example of this case in Pólya [1954, p. 9]. Pólya was also Jungen’s advisor so that several of our theorems were probably known (or obvious) to him!

3. When both α and γ are positive integers, coefficients f_n are α th order differences of integral powers of a logarithm and explicit forms are directly available by the calculus of finite differences. For instance, with $\alpha = k$ (a positive integer) and $\gamma = 1$, we have

$$[z^n](1-z)^k \log \frac{1}{1-z} = (-1)^k \frac{k!}{n(n-1)\cdots(n-k)}.$$

In this context, we may refer to a short note by Zave [1976] that discusses the case where γ is an integer and α a *negative* integer. This is directly covered by our Theorem 3A, but Zave gives an interesting direct derivation using Bell polynomials and generalized harmonic numbers.

In other words, *the results of Theorem 3A remain valid when any of α, γ may be a positive integer provided we interpret $1/\Gamma(-\alpha)$ as 0, for α a nonnegative integer, and*

² Jungen proves the classical theorem: The Hadamard product of a rational function and an algebraic function is an algebraic function. It may be of interest to combinatorialists to note that this theorem has a multivariate noncommutative “lifting” due to Schützenberger, a special form of which is: *The intersection of a regular language and a context-free language is a context-free language.*

terminate descending expansions appropriately when terms become identically zero. We leave to the reader the pleasure of working out the degeneracy cases for Theorem 3B.

The first few terms of expansions in Theorems 3A and 3B are given below:

$$\begin{aligned}
 [z^n](1-z)^\alpha \left(\frac{1}{z} \log \frac{1}{1-z} \right)^\gamma &= \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} (\log n)^\gamma \left[1 - \frac{C_1}{1!} \frac{\gamma}{\log n} + \frac{C_2}{2!} \frac{\gamma(\gamma-1)}{(\log n)^2} + \dots \right]. \\
 [z^n](1-z)^\alpha \left(\frac{1}{z} \log \frac{1}{1-z} \right)^\gamma \left(\frac{1}{z} \log \left(\frac{1}{z} \log \frac{1}{1-z} \right) \right)^\delta \\
 &= \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} (\log n)^\gamma (\log \log n)^\delta \cdot \left[1 - C_1 \frac{(\delta + \gamma \log \log n)}{\log n \log \log n} \right. \\
 &\quad \left. + \frac{C_2}{2} \frac{\delta(\delta-1) + \delta(2\gamma-1) \log \log n + \gamma(\gamma-1)(\log \log n)^2}{(\log n \log \log n)^2} + \dots \right].
 \end{aligned}$$

There $C_j = C_j(\alpha)$ represents

$$\Gamma(-\alpha) \frac{d^k}{ds^k} \frac{1}{\Gamma(-s)} \Big|_{s=\alpha}.$$

4. Extensions: Large functions and slowly varying functions. We briefly give here indications on possible extensions of our methods, first to “large” functions, then to a set of slowly varying functions.

4.1. Large functions. For functions in class $\mathcal{S}$ that become “large” enough at their singularity, their dominant singular term being of the type $(1-z)^\alpha$ with $\alpha < -1$, the analyticity conditions of our previous theorems can be weakened. A corresponding form of Theorem 1 was given in Flajolet and Odlyzko [1982] (but we erroneously stated that form with the condition $\alpha < 0$ instead of the more restrictive $\alpha < -1$). For completeness, we state the corrected form and briefly sketch the proof.

THEOREM 4. Assume that $f(z)$ is analytic in $|z| < 1$. Assume further that as $z \rightarrow 1$ in this domain,

$$(4.1a) \quad f(z) = O(|1-z|^\alpha),$$

for some real number $\alpha < -1$. Then the n th Taylor coefficient of $f(z)$ satisfies

$$(4.1b) \quad f_n = [z^n]f(z) = O(n^{-\alpha-1}).$$

Proof. As our contour $\mathcal{C}$, we now choose

$$\mathcal{C} = \left\{ z/|z| = 1 - \frac{1}{n} \right\}.$$

For $z \in \mathcal{C}$, set $z = (1 - (1/n))e^{i\theta}$, where $-\pi < \theta \leq \pi$. Note that $|z|^{-n} \leq 2e$ for $n \geq 2$, and so

$$f_n = O\left(\int_0^\pi |1-z|^\alpha d\theta\right).$$

Now, for $\pi/2 \leq \theta \leq \pi$, $|1-z| > c$ for some constant $c > 0$, so that this region contributes a bounded quantity to f_n . We also have

$$\Re(1-z) = 1 - \left(1 - \frac{1}{n}\right) \cos \theta \geq \frac{1}{n},$$

and, since $\sin \theta \geq 2\theta/\pi$ for $0 \leq \theta \leq \pi/2$,

$$|\Im(1-z)| = \left(1 - \frac{1}{n}\right) \sin \theta \geq \frac{\theta}{10}.$$

Therefore, we get $|1-z| \geq \frac{1}{2}((1/n) + (\theta/10))$, and

$$\begin{aligned} \int_0^{\pi/2} |1-z|^\alpha d\theta &= O\left(\int_0^{\pi/2} \left(\frac{1}{n} + \frac{\theta}{10}\right)^\alpha d\theta\right) \\ &= O(n^{-\alpha-1}), \end{aligned}$$

which gives our estimate. $\square$

Appropriate transfer results of all types will follow under the conditions of Theorem 4.

4.2. Slowly varying functions. We notice that, in the general proofs of § 3, it is possible to use, for all α , a contour $\mathcal{C}$ whose outer circle is of radius $R_n = 1 + \log^2 n/n$, so that R_n here plays the role of η . We could have used the same contour as in the proofs of § 3, but it is also instructive to introduce yet another contour.

It then becomes possible to extend the range of asymptotic scales leading to O -, o - and $\sim$ -transfers to a scale that includes functions of the form $(1-z)^\alpha L((1-z)^{-1})$, provided $L(u)$ is of slow variation towards infinity. Such functions capture the features of functions like $\log$, $\log \log$, etc. Function $L(u)$ is said to be of *slow variation* at ∞ if it satisfies the following conditions:

V1. There exists a positive real number u_0 , and an angle ϕ with $0 < \phi < (\pi/2)$ such that $L(u)$ is $\neq 0$ and analytic in the domain

$$(4.2) \quad \{u / -(\pi - \phi) \leq \text{Arg}(u - u_0) \leq (\pi - \phi)\}.$$

V2. There exists a function $\varepsilon(x)$, defined for $x \geq 0$ with $\lim_{x \rightarrow +\infty} \varepsilon(x) = 0$, such that for all $\theta \in [-(\pi - \phi), \pi - \phi]$ and $u \geq u_0$, we have

$$(4.3) \quad \left| \frac{L(ue^{i\theta})}{L(u)} - 1 \right| < \varepsilon(u) \quad \text{and} \quad \left| \frac{L(u \log^2 u)}{L(u)} - 1 \right| < \varepsilon(u).$$

THEOREM 5. Assume that $L(u)$ is of slow variation at ∞ , then the conditions

$$\begin{aligned} f(z) &= O\left((1-z)^\alpha L\left(\frac{1}{1-z}\right)\right), \\ f(z) &= o\left((1-z)^\alpha L\left(\frac{1}{1-z}\right)\right), \quad f(z) \sim (1-z)^\alpha L\left(\frac{1}{1-z}\right), \end{aligned}$$

as $z \rightarrow 1$ in $\Delta \setminus \{1\}$, transfer into the corresponding estimates for coefficients:

$$f_n = O(n^{-\alpha-1} L(n)), \quad f_n = o(n^{-\alpha-1} L(n)), \quad f_n \sim \frac{n^{-\alpha-1}}{\Gamma(-\alpha)} L(n).$$

Proof. (A simple adaptation of the proof of Corollary 1.) Inequalities (4.3) permit us to estimate the contributions to Cauchy's integral "near" the singularity $z = 1$, as though the $L((1-z)^{-1})$ terms were not present. $\square$

This gives us a still wider range of functions such as

$$\exp(\sqrt{\log u}), \quad \log \log \log u, \dots$$

For instance, we have the transfer

$$f(z) = O\left(\frac{1}{\sqrt{1-z}} \exp\left(\sqrt{\log \frac{1}{1-z}}\right)\right) \Rightarrow f_n = O(n^{-1/2} e^{\sqrt{\log n}}).$$

The slow variation conditions will just exclude a few functions that are very nearly a power of $(1-z)$, such as

$$\exp\left(\frac{\log(1-z)^{-1}}{\log \log(1-z)^{-1}}\right).$$

5. A comparison with alternative methods. It is interesting to note first that Darboux's method and Tauberian theorems are in a sense complementary. The former applies to functions of the form $(1-z)^\alpha$ where α is a sufficiently large positive number, while the latter necessitates $\alpha \leq 0$. Transfer methods, which require somewhat different validity conditions, cover all values of α .

1. *Darboux's method.* There is a restricted form of Darboux's method which is most commonly encountered in combinatorial analysis. It applies to Taylor coefficients of functions of the form

$$(5.1) \quad f(z) = h(z)(1-z)^\alpha \quad \alpha \notin \{0, 1, 2, \dots\},$$

where $h(z)$ is analytic in $|z| < 1 + \eta$ for some $\eta > 0$. This is for instance the form given in Henrici [1977]. That form is directly covered by Corollary 4 and (2.27c).

The most general form is based on Darboux's lemma (a combination of integration by parts and the Riemann–Lebesgue lemma), also a classical result in Fourier analysis: *If $g(z)$ is analytic in $|z| < 1$ and k times continuously differentiable on $|z| = 1$, then*

$$(5.2) \quad [z^n]g(z) = o\left(\frac{1}{n^k}\right).$$

A typical application of the method to a function $f(x)$ therefore consists in finding a form

$$f(z) = \sigma(z) + g(z)$$

where $\sigma(z)$ is a simple singular function whose coefficients are known, and $g(z)$ is a remainder term that is amenable to treatment by Darboux's lemma.

We have already seen situations where the transfer approach applies while Darboux's method does not; this may owe to the very nature of the expansion (for example, (2.1a)), or to the fact that not enough terms can be obtained until a smooth enough error term (an instance is (2.28a)). Some further examples arising in applications are discussed in § 6 below. Conversely, it may be that a function is smooth on its circle of convergence, so that Darboux's method applies, but the circle is a natural boundary and no transfer like Theorem 1 is applicable. An artificial example³ is

$$(5.3) \quad g(z) = \sin(z) \sum_{n=1}^{\infty} \frac{\lfloor \sqrt{n} \rfloor}{n^5} z^n.$$

In this case, $g(z)$ is three times continuously differentiable on $|z| = 1$ and by Darboux, $g_n = o(n^{-3})$.

³ The natural boundary property for $f(z)$ follows from the classical Pólya–Carlson theorem: *If a function is represented by a Taylor series that has integer coefficients and radius of convergence 1, either it is rational or it admits the unit circle as a natural boundary.*

In cases where both Darboux and transfer methods are applicable, transfer methods tend to give better estimates. For instance, if we determine the expansion of a function until a term of the form $g(z) = O((1 - z)^{k+1/2})$ which is also k times continuously differentiable, we get $g_n = o(n^{-k})$ by Darboux, but a better bound of $O(n^{-k-3/2})$ by transfer.

2. *Tauberian theorems.* We already gave some indication in the introduction on this subject. In our terminology, a *real Tauberian theorem* asserts conditions under which an expansion

$$f(x) \sim \frac{c}{(1-x)^\beta}, \quad (x \rightarrow 1^-)$$

(with $\beta \geq 0$) that needs to be valid only along the *real* line, translates into an estimate for coefficients in the sense of a Cesàro average:

$$\frac{1}{n} \sum_{j=1}^n f_j \sim c \frac{n^{\beta-1}}{\Gamma(\beta)}.$$

A typical sufficient validity condition is $f_n > 0$. If, furthermore, the f_n are monotonic, then, we can infer that

$$f_n \sim c \frac{n^{\beta-1}}{\Gamma(\beta)}.$$

Application of Tauberian theorems therefore requires some a priori conditions—called Tauberian side conditions—like positivity, monotonicity, to be established on the coefficients.

Tauberian theorems are useful mostly for main terms in asymptotic expansions, and they may turn out to be the only applicable tool when the circle of convergence of the function is a natural boundary. Greene and Knuth [1983] have an interesting example of using a Tauberian theorem (complemented by “bootstrapping”), which gives the asymptotic form of coefficients of the function

$$(5.4) \quad f(z) = \prod_{k \geq 1} \left(1 + \frac{z^k}{j}\right).$$

Function f has the unit circle as a natural boundary and does not seem amenable to transfer methods.

Conversely, a function with somewhat erratic coefficients such as

$$\sin z \cos \left(\log \frac{1}{1-z} \right)$$

is easily treated by transfer methods (using transfers $(1 - z)^{\pm i}$), but the coefficients are not smooth enough to allow application of a Tauberian theorem.

On all those classical questions the reader is encouraged to refer to many excellent books like De Bruijn [1981], Olver [1974], Henrici [1977], and, for Tauberian theory, Hardy [1949] or Postnikov [1980].

6. Applications. In this section, we propose to review a few applications of transfer methods in combinatorial analysis and analysis of algorithms.

2-3 trees. The problem dealt with in Odlyzko [1982] was at the origin of our interest in transfer methods. It consisted of determining the number of balanced 2-3 trees with

n external nodes. This reduces to the determination of the coefficients of a generating function $f(z)$ that satisfies the functional equation

$$(6.1) \quad f(z) = z + f(z^2 + z^3).$$

It can be recognized that $f(z)$ is analytic for $|z| < \phi^{-1}$ where $\phi = (1 + \sqrt{5})/2$ is the golden ratio (ϕ^{-1} is a fixed point of $z^2 + z^3$). A detailed study of the iteration of polynomial $\sigma(z) = z^2 + z^3$ shows that there is a singular expansion at $z = \phi^{-1}$ which is of the form

$$(6.2a) \quad \begin{aligned} f(z) &= c \log \frac{1}{1 - \phi z} + \sum_{k=-\infty}^{k=+\infty} \Omega_k (1 - z)^{2ik\pi/\lambda} + O((1 - \phi z)) \\ &= c \log \frac{1}{1 - \phi z} + \Omega\left(\frac{1}{\lambda} \log(1 - \phi z)\right) + O((1 - \phi z)). \end{aligned}$$

There $\lambda = \log(4 - \phi)$ and the infinite sum $\Omega(x)$ in (6.2a) is a fast converging Fourier series with period 1. That singular expansion can be established in an angular sector around the singularity ϕ^{-1} , so that expansion (6.2a) transfers to coefficients,

$$(6.2b) \quad \begin{aligned} \phi^{-n} f_n &= \frac{c}{n} + \sum_{k=-\infty}^{+\infty} \omega_k n^{-1 - 2ik\pi/\lambda} + O\left(\frac{1}{n^2}\right) \\ &= \frac{c}{n} + \frac{1}{n} \omega\left(\frac{1}{\lambda} \log n\right) + O\left(\frac{1}{n^2}\right), \end{aligned}$$

where $\omega(x)$ is a Fourier series (with period 1 and mean value 0) that “corresponds” to $\Omega(x)$. Form (6.2b) gives the asymptotic number of 2-3 trees with n external nodes.

This example illustrates a direct extension of Corollary 3 to the situation where the singular expansion (6.2a) contains infinitely many terms whose complex exponents have a common real part.

Height of trees. The problem of estimating the expected height of a binary tree with n internal nodes (Flajolet and Odlyzko [1982]) reduces easily to finding an asymptotic expansion for the coefficients of function $f(z)$ given by

$$(6.3) \quad \begin{aligned} f(z) &= \sum_{h \geq 0} [y(z) - y_h(z)] \\ \text{where } y_0(z) &= 0; y_{h+1}(z) = 1 + zy_h(z); y(z) = y_\infty(z) = \frac{1 - \sqrt{1 - 4z}}{2z}. \end{aligned}$$

There is a somewhat delicate analysis to determine the behaviour of the quadratic recurrence at $z = \frac{1}{4}$, from which we obtain

$$(6.4a) \quad f(z) = c \log \frac{1}{1 - 4z} + c_0 + O((1 - 4z)^{1/4 + \epsilon}),$$

for any $\epsilon > 0$. By a direct application of Corollary 3, we derive

$$(6.4b) \quad 4^{-n} f_n = \frac{c}{n} + O(n^{-5/4 + \epsilon}).$$

From (6.4b), we find after normalization that the expected height of a binary tree with n internal nodes is $2\sqrt{\pi n} + O(n^{1/4 + \epsilon})$. Here transfer lemmas are useful since it seems difficult to obtain more terms in expansion (6.4a), and application of Darboux’s method (if at all feasible) would have required an expansion until terms of a higher order like

$O((1 - 4z)^{3/2})$. Computation of higher moments led us to develop the contour used in Theorem 4.

Multidimensional search. The analysis of partial match retrieval (Flajolet and Puech [1986]) in so-called k - d -trees requires expanding a function $f(z)$ that is a component of the solution to a linear differential system with regular singular points. Using the standard theory of regular singular points, we find a singular expansion

$$(6.5a) \quad f(z) = c(1 - z)^{-\beta} + O\left(\frac{1}{(1 - z)^2}\right)$$

valid at an angle outside the circle of convergence $|z| = 1$, where $2 < \beta < 3$. (For a 2- d search, $\beta = (\sqrt{17} + 1)/2$.) It would have been possible (though a little more lengthy) to push expansion (6.5a) further, but Theorem 1 or Theorem 4 provide all that is needed to deduce directly

$$(6.5b) \quad f_n = \frac{c}{\Gamma(\beta)} n^{\beta-1} + O(n).$$

For instance, a partial match search in a 2- d tree will have expected cost $\sim Kn^{(\sqrt{17}-3)/2}$.

Common subexpression problem. The analysis of the representation of trees by compact directed acyclic graphs (dags) in Flajolet, Sippala, and Steyaert [1987] requires finding the coefficient of z^n in

$$(6.6) \quad f(z) = \frac{1}{2z} \sum_{p \geq 0} B_p [\sqrt{1 - 4z + 4z^{p+1}} - \sqrt{1 - 4z}] \quad \text{where } B_p = \frac{1}{p+1} \binom{2p}{p}.$$

A singularity analysis of (6.6) around $z = \frac{1}{4}$ shows that

$$(6.7a) \quad f(z) = \frac{c}{\sqrt{(1 - 4z) \log(1 - 4z)^{-1}}} \left[1 + O\left(\frac{1}{\log(1 - 4z)^{-1}}\right) \right].$$

By Theorem 3A and a trivially amended form of Corollary 5, we get

$$(6.7b) \quad 4^{-n} f_n = \frac{c}{\sqrt{\pi n \log n}} \left[1 + O\left(\frac{1}{\log n}\right) \right].$$

This example (see also the discussion of (1.2a), (1.2b)) is interesting since it could not be attacked by Darboux's method. As we have already indicated, even by pushing the expansion further, there is no way of obtaining a differentiable error term in a more extensive form of (6.7a). Tauberian methods would be possible candidates for attacking (6.7a), but it seems to be quite difficult to establish Tauberian side conditions on the *error term* in (6.7a), and the situation would get even worse if higher order terms were to be found. By transfer methods, we are able to conclude easily that the expected size of the maximally compacted dag representation of a tree of size n is $\sim cn/\sqrt{\log n}$.

Longest cycle in permutations. This problem, solved by Shepp and Lloyd [1966], is equivalent to finding the asymptotic form of the coefficients of

(6.8)

$$f(z) = \sum_{k \geq 0} \left[\frac{1}{1 - z} - \exp(z/1 + z^2/2 + \cdots + z^k/k) \right] = \frac{1}{1 - z} \sum_{k \geq 0} \left[1 - \exp\left(-\sum_{j \geq k} \frac{z^j}{j}\right) \right].$$

Function $f(z)$ is singular at $z = 1$, and it is natural to set $z = e^{-t}$ so that $t \rightarrow 0$ as $z \rightarrow 1$ and $z \sim 1 - t$. Two successive applications of the Euler–Maclaurin summation to (6.8) provide the approximation

$$(6.9a) \quad (1 - z)f(z) \sim \frac{G}{t} \quad \text{where } G = \int_0^\infty \left[1 - \exp\left(-\int_x^\infty \frac{e^{-t}}{t} dt\right) \right] dx.$$

Thus $F(z) \sim G(1 - z)^{-2}$, and by transfer (Corollary 3), the longest cycle in a random permutation of n elements has expected length

$$(6.9b) \quad f_n \sim Gn.$$

We observe that Shepp and Lloyd's original derivation (see also Knuth [1973a, p. 181]) proceeds along quite different lines. They first prove a Poisson approximation and then use a Tauberian theorem. But Tauberian side conditions, though expected combinatorially, are not obvious to establish.

In Flajolet and Odlyzko [1990], we apply an analysis of this type to study random mappings and find the expected diameter of a random mapping of size n .

Odd-even merge. The problem of analyzing odd-even merge sorting networks was posed by Knuth [1973b, Ex. 5.2.2.16]. Knuth reduces it to finding the Taylor coefficients of a function closely related to

$$(6.10) \quad f(z) = \frac{y}{1+y^2} + \frac{y^2}{1+y^4} + \frac{y^4}{1+y^8} + \cdots \quad \text{where } y = \frac{1 - \sqrt{1-4z}}{1 + \sqrt{1-4z}}.$$

The problem was solved by Sedgewick in 1977 by expanding, then using real approximations on coefficients and finally applying Mellin transform techniques (Sedgewick [1977]). We present here the outline of an alternative approach based on Flajolet and Prodinger [1986]. Following our general strategy, we try to determine a singular expansion of $f(z)$ around the singularity $z = \frac{1}{4}$. We can set $y(z) = e^{-t}$, so that $t \rightarrow 0$ as $z \rightarrow \frac{1}{4}$, and the problem is to analyze

$$(6.11) \quad F(t) = \frac{e^{-t}}{1 + e^{-2t}} + \frac{e^{-2t}}{1 + e^{-4t}} + \frac{e^{-4t}}{1 + e^{-8t}} + \cdots \quad \text{as } (t \rightarrow 0).$$

We assume, to simplify the discussion, that t real and positive (general theorems guarantee that the expansion can be “continued” for complex t with $\Re(t) > 0$, hence for complex z at an angle from $\frac{1}{4}$). The Mellin transform of $F(t)$ is easily found from (6.11),

$$(6.12) \quad F^*(s) \stackrel{\text{def}}{=} \int_0^\infty F(t)t^{s-1} dt = \frac{\Gamma(s)L(s)}{1-2^{-s}} \quad \text{where } L(s) = \frac{1}{1^s} - \frac{1}{3^s} + \frac{1}{5^s} - \frac{1}{7^s} + \cdots.$$

From the familiar Mellin inversion formula and a computation by residues (Doetsch [1955]), we obtain an asymptotic expansion of $F(t)$ as $t \rightarrow 0$,

$$(6.13) \quad F(t) \sim \sum \text{Res}[F^*(s)t^{-s}],$$

where the sum is extended to all poles s of $F^*(s)$ satisfying $\Re(s) \leq 0$: There is a double pole at $s = 0$, simple imaginary poles at $s = 2ik\pi/\log 2$, and simple real poles at $s = -2, -4, -6, \dots$. Thus,

$$(6.14a) \quad F(t) \sim \frac{1}{2} \log_2 \frac{1}{t} + c_0 + \Omega(\log_2 t) + 2 \sum_{k \geq 1} \frac{E_{2k}}{1-2^{-k}} \frac{t^{2k}}{((2k)!)^2}.$$

There $\Omega(x)$ is a Fourier series in x and E_{2n} is the Euler number,

$$E_{2n} = (2n)! [z^{2n}] (1/\cos z).$$

Expansion (6.14a) is a *full* expansion in increasing powers of t which, by transformation $t = -\log y(z)$, yields a full asymptotic expansion of $F(t)$. From there, we can find a full expansion for the coefficients of $f(z)$ (by transfer) and, in particular, get a complete asymptotic expansion of f_n . The same method gives Sedgewick's result (and an infinite expansion): *The expected number of exchanges in odd-even merge applied to two sequences of length n is $\frac{1}{4}n \log_2 n + \dots$* . We refer the reader to Flajolet and Prodinger [1986] for a similar example treated in detail.

This "synthetic" method differs from the usual approach of De Bruijn, Knuth, and Rice [1972]: In accordance with our general principles, we operated only at the level of the generating function (using complex Mellin transforms to derive a singular expansion), and concluded directly by a transfer theorem.

Limit distributions in combinatorics. Consider a bivariate generating function of the form

$$P(z, u) = \exp(uG(z)).$$

Such functions arise naturally in counting combinatorial structures $\mathcal{P}$ that are decomposable as sets of basic building blocks (components) enumerated by $G(z)$. In this context, the polynomials $p_n(u) = [z^n]P(z, u)$ are the generating polynomials giving the distribution of the number of components in a random $\mathcal{P}$ structure of size n . Under the condition that $G(z)$ has a dominant singularity of a logarithmic type, methods of this paper may be used to estimate asymptotically the characteristic function $p_n(e^{it})$ which, once suitably normalised, tends to $e^{-t^2/2}$. From the continuity theorem for characteristic functions, we deduce that the number of components in a "large" ($n \rightarrow \infty$) random $\mathcal{P}$ structure tends to a limiting Gaussian distribution. A typical application is to cycles in permutations. There are extensions to Pólya's theory of counting, with the corresponding analytic scheme

$$P(z, u) = \exp\left(\frac{u}{1} G(z) \pm \frac{u^2}{2} G(z^2) + \frac{u^3}{3} G(z^3) \pm \frac{u^4}{4} G(z^4) + \dots\right).$$

We obtain, for instance: *The number of irreducible factors in a random polynomial with coefficients in $GF(q)$ is asymptotically Gaussian.*

Under different analytic conditions on the "generator" $G(z)$, Bender [1973] and Canfield [1977] have established similar asymptotic normality results. Detailed proofs of results cited here are presented in Flajolet and Soria [1988].

Acknowledgments. The authors express their gratitude to D. E. Knuth for encouragement to write this paper and for several useful suggestions regarding the presentation. The first author is thankful to D.E.K. for an invitation at Stanford University during which his work on the subject was done for a large part.

REFERENCES

- E. BENDER [1973], *Central and local limit theorems applied to asymptotic enumeration*, J. Combin. Theory Ser. A, 15, pp. 91–111.
 — [1974], *Asymptotic methods in enumerations*, SIAM Rev., 16, pp. 485–515.
 E. R. CANFIELD [1977], *Central and local limit theorems for coefficients of binomial type*, J. Combin. Theory Ser. A, 23, pp. 275–290.

- M. CAR [1982], *Factorisation dans $F_q[X]$* , C. R. Acad. Sci. Paris Série I, 294, pp. 147–150.
- [1984], *Ensembles de polynômes irréductibles et théorèmes de densité*, Acta Arith., 44, pp. 323–342.
- L. COMTET [1974], *Advanced Combinatorics*, Reidel, Dordrecht.
- N. G. DE BRUIJN [1981], *Asymptotic Methods in Analysis*, Dover, New York.
- N. G. DE BRUIJN, D. E. KNUTH, AND S. O. RICE [1972], *The average height of planted plane trees*, in Graph Theory and Computing, R.-C. Read, ed., Academic Press, New York, pp. 15–22.
- G. DOETSCH [1955], *Handbuch der Laplace Transformation*, Birkhäuser, Basel.
- P. FLAJOLET AND A. M. ODLYZKO [1982], *The average height of binary trees and other simple trees*, J. Comput. System Sci., 25, pp. 171–213.
- [1990], *Random mapping statistics*, in Proceedings Eurocrypt '89, J.-J. Quisquater Ed., Lecture Notes in Computer Science, to appear.
- P. FLAJOLET AND H. PRODINGER [1986], *Register allocation for unary-binary trees*, SIAM J. Comput., 15, pp. 629–640.
- P. FLAJOLET AND C. PUECH [1986], *Partial match retrieval of multidimensional data*, Journal of the ACM, 33, pp. 371–407.
- P. FLAJOLET, P. SIPALA, AND J.-M. STEYAERT [1987], *The analysis of tree compaction in symbolic manipulations*, preprint.
- P. FLAJOLET AND M. SORIA [1988], *Normal limiting distributions for the number of components in combinatorial structures*, INRIA Res. Report, 809, March 1988. J. Combin. Theory Ser. A, 1990, to appear.
- I. GOULDEN AND D. JACKSON [1983], *Combinatorial Enumerations*, John Wiley, New York.
- D. H. GREENE AND D. E. KNUTH [1982], *Mathematics for the Analysis of Algorithms*, 2nd ed., Birkhäuser, Boston.
- G. H. HARDY [1940], *Ramanujan, Twelve Lectures Suggested by His Life and Work*, Cambridge University Press. Reprinted by Chelsea Publishing Company, New York, 1978.
- [1949], *Divergent Series*, Oxford University Press, London.
- G. H. HARDY AND J. E. LITTLEWOOD [1914], *Tauberian theorems concerning power series and Dirichlet's series whose coefficients are positive*, Proc. London Math. Soc., 13, pp. 174–191.
- P. HENRICI [1977], *Applied and Computational Complex Analysis*, John Wiley, New York.
- R. JUNGEN [1931], *Sur les séries de Taylor n'ayant que des singularités algebriques-logarithmiques sur leur cercle de convergence*, Commentarii Mathematici Helvetici, 3, pp. 266–306.
- D. E. KNUTH [1973a], *The Art of Computer Programming Vol. 1: Fundamental Algorithms*, 2nd ed., Addison-Wesley, Reading, MA.
- [1973b], *The Art of Computer Programming Volume 3: Sorting and Searching*, Addison-Wesley, Reading, MA.
- [1981], *The Art of Computer Programming Volume 2: Semi-Numerical Algorithms*, 2nd ed., Addison-Wesley, Reading, MA.
- A. M. ODLYZKO [1982], *Periodic oscillations of coefficients of power series that satisfy functional equations*, Adv. in Math., 44, pp. 180–205.
- F. W. J. OLVER [1974], *Asymptotics and Special Functions*, Academic Press, New York.
- G. PÓLYA [1954], *Induction and Analogy in Mathematics*, Princeton University Press, Princeton, NJ.
- A. G. POSTNIKOV [1980], *Tauberian theory and its applications*, in Proc. Steklov Inst. Math., 144; English translation, Amer. Math. Soc. Transl.
- R. SEDGEWICK [1978], *Data movement in odd-even merging*, SIAM J. Comput., 7, pp. 239–272.
- A. SELBERG [1954], *Note on a paper by L. G. Sathe*, J. Indian Math. Soc., 18, pp. 83–87.
- L. A. SHEPP AND S. P. LLOYD [1966], *Ordered cycle lengths in a random permutation*, Trans. Amer. Math. Soc., 121, pp. 340–357.
- R. P. STANLEY [1986], *Enumerative Combinatorics*, Wadsworth and Brooks/Cole, Monterey.
- E. C. TITCHMARSH [1939], *The Theory of Functions*, 2nd ed., Oxford University Press, London.
- E. T. WHITTAKER AND G. N. WATSON [1927], *A Course of Modern Analysis*, 4th ed., Cambridge University Press, Cambridge.
- R. WONG AND M. WYMAN [1974], *The method of Darboux*, J. Approx. Theory, 10, pp. 159–171.
- D. A. ZAVE [1976], *A series expansion involving the harmonic numbers*, Inform. Process. Lett., 5, pp. 75–77.

REVERSIBLE TURING MACHINES AND POLYNOMIAL TIME REVERSIBLY COMPUTABLE FUNCTIONS*

G. JACOPINI†, P. MENTRASTI‡, AND G. SONTACCHI‡

Abstract. The reversible Turing machine (i.e., r -machine) was introduced initially by C. H. Bennett [*IBM J. Res. Develop.*, 6 (1973), pp. 525–532]. In the first part of the paper a convenient representation of r -machines is introduced by means of diagrams. By using this method the following theorem can be proved: the invertible partial functions are exactly those that can be computed without surplus information by r -machines. Therefore the following problem is pointed out: are the invertible functions that can be computed in polynomial time also r -computable in polynomial time? In the second part of the work this open question is connected with the problem $P = NP$ and with the problem of the existence of “one-way” bijections.

Key words. reversibility, Turing machine, one-way bijection

AMS(MOS) subject classifications. 68Q05, 68Q15, 03D10, 03D15

1. Introduction. The notion of reversible computation was initially developed with motivations related to physics [1], [5], [6]. Reversible computation is considered a process in which, given any instantaneous state, not only the future state but also the past one is unambiguously defined.

In this paper we are interested in reversible computation only from a mathematical point of view.

In 1973 Bennett [1] defined a reversible Turing machine and proved that any Turing machine (TM) with one tape and some limitations on the input could be simulated by a reversible Turing machine with three tapes. The alphabet of the three-tape Turing machine is variable, as it depends on the number of the 5-tuples of the simulated machine. Generally the simulation produces surplus information. He then proved that the surplus can be reduced to the simple reproduction of the input.

In our paper we use only a one-tape reversible Turing machine with fixed alphabet $\{0, 1\}$ (r -machine). The first result is as follows: expressive diagrams are introduced, which permits us to represent all and only the r -machines.

Then, we prove (Theorem 5.5) that the class of computable functions without superfluous information from an r -machine is identical with the class of invertible recursive functions. To obtain this result we make use of simulations in which alphabet and codification are the same (unlike Bennett) both in the simulating machine and the simulated one. The lack of simple correspondence between the 5-tuples of the two machines is not a problem. We do not represent the machines by means of 5-tuples but by means of diagrams.

Finally we prove that a polynomially honest bijection [3] is a two-way bijection if and only if it is poly-time computable with an r -machine, and that the one-way polynomial honest bijections (if they exist) are those that are poly-time computable from an ordinary Turing machine but not from one of our r -machines; their inverse would be computable in polynomial time by a nondeterministic Turing machine, but not by an ordinary Turing machine.

* Received by the editors October 17, 1988; accepted for publication (in revised form) March 9, 1989.

† Dipartimento di Matematica, Università degli Studi di Roma “La Sapienza,” Piazzale Aldo Moro 5, 00185 Roma, Italy. This work was partially supported by “Ministero Pubblica Istruzione”-“Progetto Nazionale Teoria de gli Algoritmi.”

‡ Dipartimento di Matematica Pura ed Applicata, Università dell’Aquila, Via Roma, 67100 L’Aquila, Italy.

2. Reversible Turing machine.

DEFINITION 2.1. A reversible Turing machine, r -machine, R is determined by a 5-tuple

$\langle Q_R, \Sigma_R, q_{0R}, D_R, T_R \rangle$ such that:

$Q_R = \{q_0, q_1, q_2, \dots, q_m\}$ is a finite set of states;

$\Sigma_R = \{s_0, s_1, s_2, \dots, s_n\}$ is a finite alphabet (of the tape);

$q_{0R} = q_0 (\in Q_R)$ is the initial-final state;

D_R is a function that associates a direction of movement to each state different from q_0 , i.e., $D_R : Q_R - \{q_0\} \rightarrow \{\rightarrow, \leftarrow\}$;

T_R is a set of transitions, i.e., a permutation on the set $K_R = Q_R \times \Sigma_R$, i.e.,

$T_R : K_R \leftrightarrow K_R$.

Let us give a short description of the behaviour of R , pointing out the differences regarding the classical TM described with 5-tuples.

A specific state is associated to the machine only in the moment in which it is halfway between two symbols. Instead, when it observes a symbol (and is replacing the symbol with another one) it does not have a specific state (but rather a pair of states: past state–future state).

$D_R(q_i) = \rightarrow (\leftarrow)$ means that while the machine moves along the tape between one symbol and another, being in the q_i state, the direction of its movement is always from left to right (from right to left).

The generic 4-tuple $\langle \langle q_i, s_j \rangle, \langle q_h, s_k \rangle \rangle \in T_R (i, h \neq 0)$ has the following meaning. If the machine, as it moves along the tape in state q_i , meets the symbol s_j , it replaces s_j with s_k and moves away in state q_h .

The 4-tuple $\langle \langle q_0, s_j \rangle, \langle q_h, s_k \rangle \rangle \in T_R$ has the following meaning. If the machine in the initial configuration¹ observes the symbol s_j , it replaces s_j with s_k and moves away in state q_h .

The 4-tuple $\langle \langle q_i, s_j \rangle, \langle q_0, s_k \rangle \rangle \in T_R$ has the following meaning. If the machine moves in state q_i and meets the symbol s_j , it replaces s_j with s_k and stops, as the final configuration has been reached.

The 4-tuple $\langle \langle q_0, s_j \rangle, \langle q_0, s_k \rangle \rangle \in T_R$ has the following meaning. If the machine in the initial configuration observes the symbol s_j , it replaces s_j with s_k and stops without any movement done.

Example 2.2. (Reversible Turing machine). $R1 = \langle Q_{R1}, \Sigma_{R1}, q_0, D_{R1}, T_{R1} \rangle$ where:

$$Q_{R1} = \{q_0, q_1, q_2, q_3\};$$

$$\Sigma_{R1} = \{0, | \};$$

$$D_{R1}(q_1) = \leftarrow, D_{R1}(q_2) = \rightarrow, D_{R1}(q_3) = \leftarrow;$$

$$\begin{aligned} T_R = \{ & \langle \langle q_0, 0 \rangle, \langle q_2, | \rangle \rangle, \langle \langle q_0, | \rangle, \langle q_1, 0 \rangle \rangle, \\ & \langle \langle q_1, 0 \rangle, \langle q_2, 0 \rangle \rangle, \langle \langle q_1, | \rangle, \langle q_3, 0 \rangle \rangle, \\ & \langle \langle q_2, 0 \rangle, \langle q_1, | \rangle \rangle, \langle \langle q_2, | \rangle, \langle q_3, | \rangle \rangle, \\ & \langle \langle q_3, 0 \rangle, \langle q_0, 0 \rangle \rangle, \langle \langle q_3, | \rangle, \langle q_0, | \rangle \rangle \} ; \end{aligned}$$

¹ By “configuration” we mean the content of the tape with indication of the observed cell (which will be indicated by underlining).

if the initial configuration is

$$\dots | \quad \underline{0} \quad 0 \dots$$

the computation goes on as

$$\begin{array}{ccc} \dots | & & \xrightarrow{q_2} 0 \dots \\ \dots | & & \xleftarrow{q_1} | \dots \\ \dots | & \xleftarrow{q_3} 0 & | \dots \end{array}$$

and the final configuration is

$$\dots | \quad 0 \quad | \dots$$

DEFINITION 2.3. For each r -machine R we define the r -machine

$$R^{-1} = \langle Q_{R^{-1}}, \Sigma_{R^{-1}}, q_{0R^{-1}}, D_{R^{-1}}, T_{R^{-1}} \rangle$$

in the following manner:

$$\begin{aligned} Q_{R^{-1}} &= Q_R; \\ \Sigma_{R^{-1}} &= \Sigma_R; D_{R^{-1}}(q_i) = \rightarrow \Leftrightarrow D_R(q_i) = \leftarrow; \\ q_{0R^{-1}} &= q_{0R} = q_0; \\ T_{R^{-1}} &= T_R^{-1}. \end{aligned}$$

We call R^{-1} the reverse of R .

PROPOSITION 2.4. If R transforms a configuration γ_1 into another γ_2 , then R^{-1} transforms γ_2 into γ_1 , always in the same amount of time. In fact R^{-1} operates the inverse computation moving backwards through all the intermediate configurations.

Example 2.5. The reverse machine of R_1 of Example 2.2 is

$$R_1^{-1} = \langle Q_{R_1^{-1}}, \Sigma_{R_1^{-1}}, q_0, D_{R_1^{-1}}, T_{R_1^{-1}} \rangle$$

where:

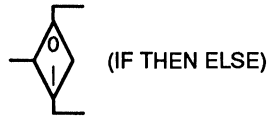
$$\begin{aligned} Q_{R_1^{-1}} &= Q_{R_1} = \{q_0, q_1, q_2, q_3\}; \\ \Sigma_{R_1^{-1}} &= \Sigma_{R_1} = \{0, |\}; \\ \{D_{R_1^{-1}}(q_i)\} &= \{\rightarrow, \leftarrow\} - \{D_{R_1}(q_i)\}, i = 1, 2, 3. \\ D_{R_1^{-1}}(q_1) &= \rightarrow, D_{R_1^{-1}}(q_2) = \leftarrow, D_{R_1^{-1}}(q_3) = \rightarrow. \\ T_{R_1^{-1}} &= \{\langle\langle q_0, 0 \rangle, \langle q_3, 0 \rangle\rangle, \langle\langle q_0, | \rangle, \langle q_3, | \rangle\rangle, \\ &\quad \langle\langle q_1, 0 \rangle, \langle q_0, | \rangle\rangle, \langle\langle q_1, | \rangle, \langle q_2, 0 \rangle\rangle, \\ &\quad \langle\langle q_2, 0 \rangle, \langle q_1, 0 \rangle\rangle, \langle\langle q_2, | \rangle, \langle q_0, 0 \rangle\rangle, \\ &\quad \langle\langle q_3, 0 \rangle, \langle q_1, | \rangle\rangle, \langle\langle q_3, | \rangle, \langle q_2, | \rangle\rangle\}. \end{aligned}$$

3. *R*-graphs.

3.1. It is well known that a deterministic Turing machine on the binary alphabet $\Sigma = \{0, |\}$ can also be described by a connected *H*-graph (Hermes graph) with the following nodes:

0 — (START) in one and only one copy,

— 0 (STOP) in one and only one copy,



(where for each IF THEN ELSE there is one CONFLUENCE) and the following nodes (representing elementary machines):

— → — (A STEP TO RIGHT)

— ← — (A STEP TO LEFT)

— 0 — (WRITE 0)

— | — (WRITE |).

Note 3.1.1. In this notation the strictly nonreversible components are: $\sqcap$ — (it is not known from where the computation originates after a confluence) and — 0 —, — | — (the previous symbol is not restorable).

It is our intention to give a notation in the form of a graph also for the *r*-machine on the binary alphabet $\Sigma = \{0, |\}$.

DEFINITION 3.2. We define a class of connected graphs which we call *r*-graphs and which have the following nodes:

0 — (START) in one and only one copy,

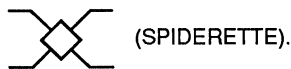
— 0 (STOP) in one and only one copy,

— → — (A STEP TO RIGHT),

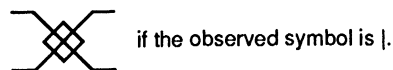
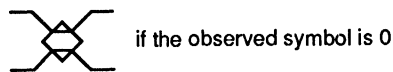
— ← — (A STEP TO LEFT),

and furthermore the new nodes

— A — (ALTER)



3.2.1. “A” represents the elementary reversible machine which exchanges the observed symbol and the “SPIDERETTE” means:



Note 3.2.2. Each component of the r -graphs is reversible. In particular, with regard to the “SPIDERETTE” we would point out that, once the symbol is known, the output edge determines in an unambiguous manner the input edge and vice versa.

THEOREM 3.3. *Each r -graph represents a r -machine on the alphabet $\Sigma = \{0, |\}$ and (vice versa) each r -machine on the alphabet $\Sigma = \{0, |\}$ has an r -graph which represents it.*

Proof. Let G be an r -graph with m movement nodes (i.e., $\rightarrow$, $\leftarrow$). To construct the r -machine R represented by G first at all we label 0 — and — 0 with q_0 and each movement node with a distinct q_i ($i = 1, 2, \dots, m$).

We define $Q_R = \{q_0\} \cup \{q_i \mid i = 1, 2, \dots, m\}$.

We define $D_R(q_i)$ equal to the symbol of the node labelled with q_i ($i = 1, 2, \dots, m$).

Now we define T_R as follows.

3.3.1. $\langle \langle q_i, s_j \rangle, \langle q_h, s_k \rangle \rangle \in T_R$ if and only if there is a path on G from q_i to q_h (directed to — 0 and from 0 —) that satisfies the following requests:

- (a) just out of the node labelled q_i , the symbol observed is s_j ;
- (b) when an A node is passed, the symbol changes;
- (c) when a SPIDERETTE node is passed, the path to be followed is established in agreement with § 3.2.1;
- (d) just before passing the node labelled q_h , the observed symbol is s_k .

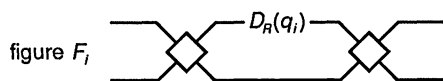
To verify that T_R is a permutation of $Q_R \times \Sigma_R$, the following must be verified:

- (i) for each $\langle q_i, s_j \rangle \in Q_R \times \Sigma_R$ there is one and only one $\langle q_h, s_k \rangle \in Q_R \times \Sigma_R$ such that § 3.3.1 holds, and
- (ii) for each $\langle q_h, s_k \rangle \in Q_R \times \Sigma_R$ there is one and only one $\langle q_i, s_j \rangle \in Q_R \times \Sigma_R$ such that § 3.3.1 holds.

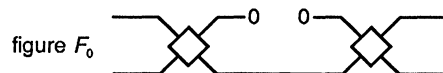
The uniqueness of (i) derives from the fact that, starting from any labelled node of the graph, the observed symbol determines unambiguously the future path until another labelled node has been reached; the uniqueness of (ii) is verified in a similar manner, observing that the backward path is determined symmetrically with the same rules.

The existence of (i) (and similarly for (ii)) is guaranteed by the fact that in the path from q_i to q_h (from q_h to q_i) the same edge can be covered at the most two times (the symbols of the alphabet being two).

3.3.2. Vice versa, given an r -machine, to have an r -graph which represents it, we draw for each state $q_i \neq q_0$ the following:



and for q_0 ,



Therefore the free edges must be connected as follows:

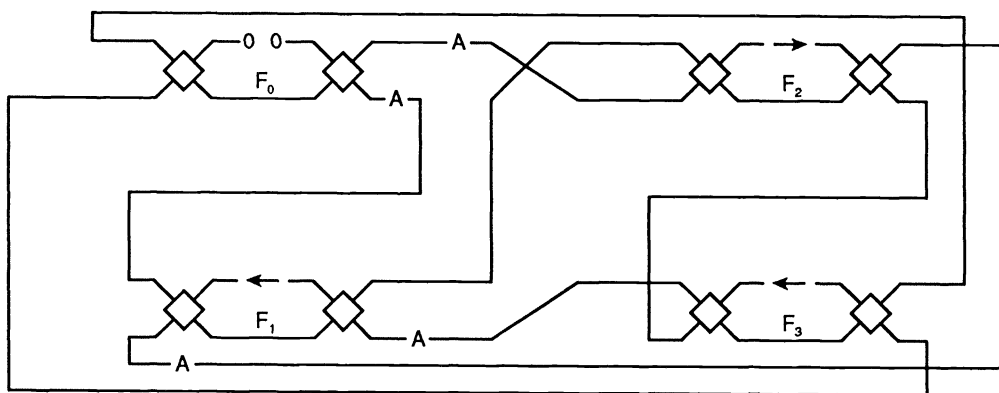
For each element $\langle \langle q_i, s_j \rangle, \langle q_h, s_k \rangle \rangle \in T_R$ an output edge of the figure F_i of the r -graph corresponding to q_i is connected with an input edge of the figure F_h of the r -graph corresponding to q_h in the following manner:

- (a') if $s_j = 0$ and $s_k = 0$ we connect directly the upper output of F_i with the upper input of F_h ;

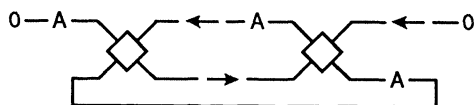
- (b') if $s_j = 0$ and $s_k = |$ we connect the upper output of F_i with the lower input of F_h placing on the path an A;
- (c') if $s_j = |$ and $s_k = 0$ we connect the lower output of F_i with the upper input of F_h placing on the path an A;
- (d') if $s_j = |$ and $s_k = |$ we connect directly the lower output of F_i with the lower input of F_h .

Note 3.3.3. The r -graph, which represents an r -machine, is not unique. If the following transformations $r\text{-graph} \rightarrow r\text{-machine}$ and $r\text{-machine} \rightarrow r\text{-graph}$ are operated one after the other generally we do not return to the initial r -graph but to a new r -graph which represents the same r -machine. Furthermore the r -graph that represents the r -machine R , obtained through the translation described in § 3.3.2, can result nonconnected, but in such a case R has inaccessible states (for any initial configuration) and it is equivalent to a machine that has fewer states, of which its translation in r -graph is the connected component of the first graph (necessarily unique) which contains 0 — and — 0 .

Example 3.4. The translation into r -graph of $R1$ of Example 2.2 is the following:



but the following r -graph also represents $R1$.



3.5. Given an r -graph as G , we consider the r -graph, which we call G^{-1} , obtained from G with a specular symmetry (right $\leftrightarrow$ left). G^{-1} represents an r -machine which is the reverse machine of the one represented by G .

3.6. It is known that a nondeterministic Turing machine on the binary alphabet $\{0, |\}$ can be described by a connected $\bar{H}$ -graph which has the same nodes as the H -graph (3.1) with the addition of the node $-\square$, its meaning being nondeterministic choice of the next operation to perform. We designate with $0-\overset{0}{|}-0$ the nondeterministic Turing machine, its computation consisting in the nondeterministic choice of writing 0 or $|$ in the observed cell. We designate with $0-\uparrow-0$ the deterministic Turing machine that does not provide any output for any input configuration.

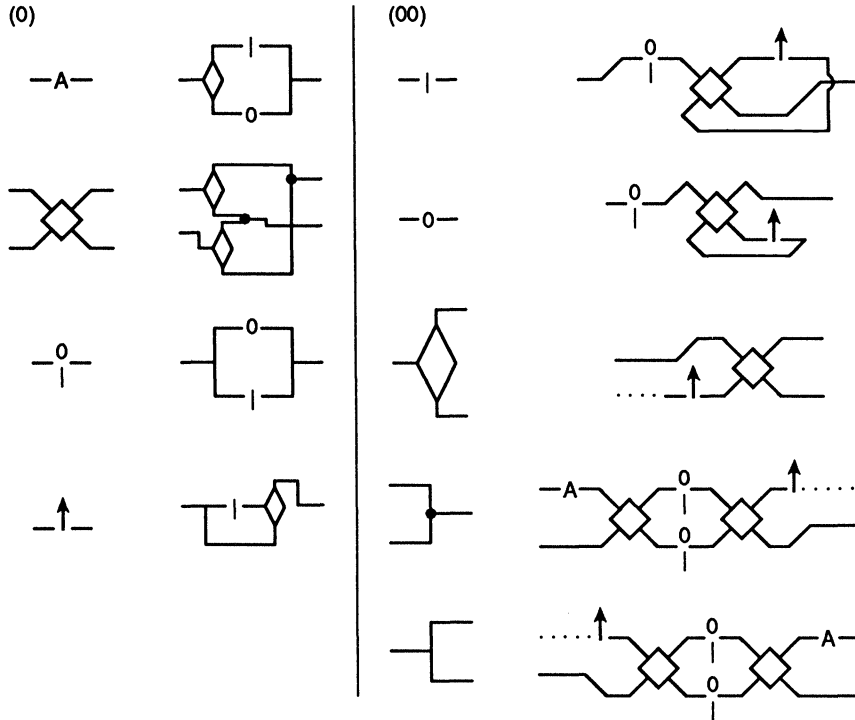
DEFINITION 3.7. We define $\bar{r}$ -graph as a connected graph, its nodes being the same as those of r -graphs, with the addition of $-\overset{0}{|}-$ and $-\uparrow-$.

3.8. All the deterministic reversible Turing machines can be represented by r -graphs (Definition 3.2), all the ordinary Turing machines can be represented by H -graphs (§ 3.1) (but not by r -graphs). Vice versa, the nondeterministic Turing machines can be

represented by $\bar{H}$ -graphs or, as well, (only in a manner formally reversible) by an $\bar{r}$ -graph (substantially for nondeterministic machines there is no difference between reversible and nonreversible). In fact it yields the following proposition.

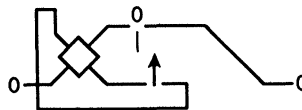
PROPOSITION 3.9. *$\bar{H}$ -graphs and $\bar{r}$ -graphs describe the same class of machines.*

Proof. To translate an $\bar{r}$ -graph into an $\bar{H}$ -graph we substitute the nodes on the left with the figures on the right in ⁽⁰⁾ and to translate an $\bar{H}$ -graph into an $\bar{r}$ -graph we substitute the nodes on the left with the figures on the right in ⁽⁰⁰⁾:²



Note 3.10. If we represent the nondeterministic Turing machine M with an r -graph, its specular copy is still an r -graph that represents the machine M^{-1} , the reverse of M . We can extend what was said in Proposition 2.4 to the nondeterministic machine M . That is, if there is a computation of M that transforms an initial configuration γ_1 into a final configuration γ_2 , then there is a computation of M^{-1} that operates the inverse transformation from γ_2 to γ_1 and the two computations take the same amount of time. For each nondeterministic Turing machine M there is a reverse machine M^{-1} : if M is reversible, M^{-1} is reversible also. Vice versa, a deterministic Turing machine has a reverse machine that is generally nondeterministic.

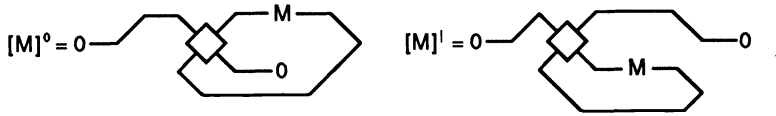
Example 3.11. The reverse machine of $0 \text{---} | \text{---} 0$ is



² The dotted edges are not used and can be connected to each other (we point out that they are equal in number).

4. Simulation of Turing machine with r -machine. We will now introduce a few notations which we will use further on.

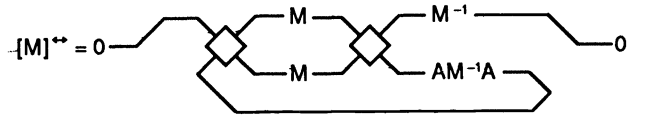
Notation 4.1. Let M be any r -machine; we put



The computation of $[M]^0$ ($[M]^1$) consists in the search of the first 0 (|) that is encountered repeating the transformation induced by M . If the observed symbol is initially | (0) the computation does not change the configuration.

Note 4.2. In the above figures the symbol M obviously stands for the graph representing the r -machine M without 0— and —0. This holds for all the pictures that follow.

Notation 4.3. We put



If M performs a movement, $[M]^*$, in what follows, is used to represent the exchange between the content of the observed cell and the content of the cell reachable with the movement performed by M , with return to the initially observed cell.

Notation 4.4. We put



$[M]^+$ is used later on to alter the content of the cell reachable by means of M only if the content of the initial cell (which is also the final one) is |.

Notation 4.5. We indicate with $\text{SYM}(M)$ the r -graph that is obtained from an r -graph representing M by substituting each $\rightarrow$ with $\leftarrow$ and vice versa ($\text{SYM}(M)$ and M^{-1} must not be confused).

Notation 4.6. $\rightarrow^n$ ($\leftarrow^n$) stands for $\rightarrow$ ($\leftarrow$) repeated n consecutive times.

DEFINITION 4.7. We call a semitape Turing machine a Turing machine such that the final cell is the initial one and each cell to the left of the initial-final cell is never observed during the computation.

THEOREM 4.8. Let M be a semi-tape Turing machine on the alphabet $\Sigma = \{0, |\}$ that transforms the initial generic configuration

$$\cdots \underline{i_0} i_1 i_2 i_3 \cdots^3$$

in the final one

$$\cdots \underline{\omega_0} \omega_1 \omega_2 \omega_3 \cdots.$$

³ $i_0, i_1, i_2, i_3, \cdots, \omega_0, \omega_1, \omega_2, \omega_3, \cdots \in \{0, |\}$, the underlined symbols ($\underline{i_0}, \underline{\omega_0}$) indicate the observed cell (in this case the same one).

Then there exists an r -machine M^G on the alphabet $\Sigma = \{0, |\}$ that transforms the initial configuration

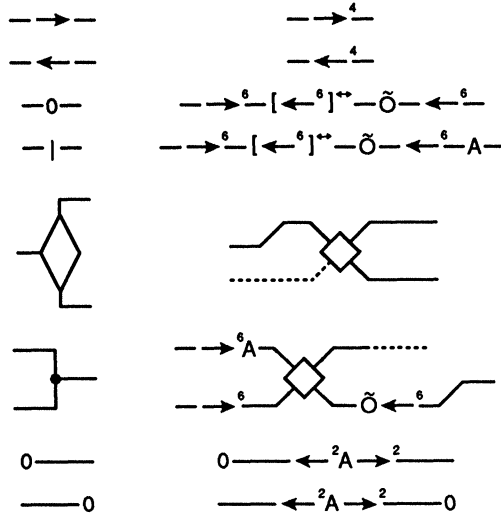
$$\cdots 00 \underline{i_0} 000 i_1 000 i_2 000 i_3 0 \cdots$$

in the final one

$$\cdots 00 \underline{\omega_0} ? 0 ! \omega_1 ? 0 ! \omega_2 ? 0 ! \omega_3 ? 0 ! \cdots,$$

where the content of the cells marked “?” or “!” denotes unspecified suitable bits. Moreover, during the computation, M^G can visit to the left of the initial-final cell two cells at most.

Proof. The graph of M^G can be obtained from the graph of M by substituting the nodes in the following manner:



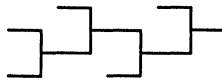
Where

$$\begin{aligned} \tilde{O} &= 0 \leftarrow {}^4 A \rightarrow {}^4 H [H^{-1}] \leftrightarrow \rightarrow {}^2 A \rightarrow {}^2 H^{-1} \leftarrow {}^4 A \rightarrow {}^4 0 \\ H &= 0 \leftarrow {}^4 [\leftarrow {}^4] \mid \rightarrow [\rightarrow {}^4] {}^0 \leftarrow {}^2 0 \end{aligned}$$

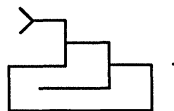
$\tilde{O}$ has the effect of printing a 0 in the observed cell. This cannot be done “sic et simpliciter” (the information cannot be destroyed), but such a result can be obtained by exchanging the content (?) of the observed cell with that of another cell which surely contains 0 and of which the new content, which we call garbage, will no longer be used; H (the initial observed cell is the same as that of $\tilde{O}$) searches for the first useful cell to store the garbage. The three intercalated cells (0 0 0) with the “ i ’s” of the input of M^G and with the “ ω ’s” of the output (? 0 !) are used by $\tilde{O}$ in the following manner: the first ones of each set of three cells are used, in sequence, for storing the garbage; the central ones always contain 0 (they are only temporarily altered by $\tilde{O}$ for operative needs of H); the third ones are altered in sequence by $\tilde{O}$ which assigns them the content |. Therefore with each intervention of $\tilde{O}$ a new set of three 0 0 0 is transformed in ? 0 !; in the final configuration the “!’s” in the third position (of each set of the three) will be | for some left segment and then 0. To the right of the last | also the bits indicated with ? will naturally be 0.

PROPOSITION 4.9. *The computation time of M^G , when the input changes, is at most proportional to the square of the computation time of M .*

Proof. In fact, $\tilde{O}$ requires a time at most proportional to the length of that part of tape observed up to then, which in turn is at most proportional to the computation time spent by M until then. The number of operations of $\tilde{O}$ during the simulation corresponds to the number of nodes $\text{—}0\text{—}$, $\text{—}| \text{—}$, $\sqsupset\text{—}$ that are crossed during the simulated computation and it is at the most proportional to the computation time of M . The constant of proportionality can depend on M , but not on the particular computation. In fact, it is true that the crossing of a node $\sqsupset\text{—}$ does not correspond to a computation step, i.e., it does not involve a change of state of the simulated machine. At the beginning or at the end of a path that crosses only $\sqsupset\text{—}$ nodes in fixed number,



there must be nodes of another kind (the length of the path will only influence the constant of proportionality). Otherwise a loop made up entirely of $\sqsupset\text{—}$ would correspond necessarily to a divergent computation



5. R -computable functions.

DEFINITION 5.1. Let X and Y be two sets. The function $f : X \rightarrow Y$ is invertible if and only if

$$x_1, x_2 \in X, \quad f(x_1) = f(x_2) \Rightarrow x_1 = x_2.$$

Note 5.2. It is known that if f is invertible there exists $f^{-1} : Y \rightarrow X$ such that:

$$(5.2.1) \quad x \in X \Rightarrow f^{-1}(f(x)) = x.$$

$$(5.2.2) \quad y \in Y \Rightarrow f(f^{-1}(y)) = y.$$

$$(5.2.3) \quad (f^{-1})^{-1} = f.$$

DEFINITION 5.3. $f \in \mathbf{I}$ if and only if f is a recursive partial invertible function of one variable.

Note 5.3.1. It is known that if $f \in \mathbf{I}$ then $f^{-1} \in \mathbf{I}$.

DEFINITION 5.4. Let $n \in \mathbf{N}$ and $n = b_0 b_1 b_2 \cdots b_{|n|-1}$ be its binary notation, $b_i \in \{0, 1\}$ ($b_0 = 1$ if and only if $n > 0$, $b_0 = 0$ and $|n| = 1$ if and only if $n = 0$). Further on we will say that the following tape configuration

$$\cdots 000 b_0 | b_1 | b_2 | \cdots b_{|n|-1} | 00 \cdots,$$

which we indicate with $\langle\langle n \rangle\rangle$, represents n .

Let $X \subseteq \mathbf{N}$ and $Y \subseteq \mathbf{N}$, we say that $f : X \rightarrow Y$ is r -computable if and only if there is an r -machine R_f (that computes f) such that for each $n \in X$ transforms the initial configuration representing n in the final one representing $f(n)$ (without superfluous information).

Note 5.4.1. We point out that the “|’s” that intercalate the b_i ’s are necessary to indicate the end of the number represented. They would not be necessary if the alphabet of the tape contained three symbols, for example, $\{0, |, \#\}$.

5.5. Main theorem. *Let $X \subseteq \mathbb{N}$ and $Y \subseteq \mathbb{N}$, $f : X \rightarrow Y$,*

$$f \in \mathbf{I} \Leftrightarrow f \text{ is } r\text{-computable}.$$

Proof. ($\Leftarrow$) The r -machines are Turing machines, therefore f is recursive. If for $x_1 \neq x_2$, $f(x_1) = f(x_2)$ then (see § 2.4) the machine R_f starting from $\langle\langle y \rangle\rangle$ would have two distinct final configurations.

($\Rightarrow$) Consider (see §§ 5.3.1 and 4.8) the r -machines M_f^G , $M_{f^{-1}}^G$. To define R_f we need to define two suitable r -machines, i.e., EXPANDED and MIRROR. EXPANDED transforms the initial configuration

$$\cdots ???0000b_0 | b_1 | b_2 | \cdots b_n | 000 \cdots$$

into the final

$$\cdots ???0000b_0000 | 000b_1000 | 000b_2000 | \cdots 000b_n000 | 000 \cdots$$

without ever visiting the cells containing ?.

MIRROR writes on the left semi-tape, which is initially empty, a specular copy of the right semi-tape only with regard to every fourth cell, the others remaining 0. More precisely MIRROR transforms the initial configuration

$$\cdots 00000?0!b_0?0! | ?0!b_1?0! | ?0! \cdots$$

into the final

$$\cdots | 000b_1000 | 000b_00000000?0!b_0?0! | ?0!b_1?0! | ?0! \cdots$$

$$\begin{aligned} \text{EXPANDED} = 0 \text{---} [\rightarrow^2]^0 [[\rightarrow^3] \leftrightarrow \rightarrow^5 A [\rightarrow^8]^0 [[\rightarrow^6] \leftrightarrow \leftarrow^4 \\ [\rightarrow^6] \leftrightarrow \leftarrow^4]^0 A [\rightarrow^6] \leftrightarrow \leftarrow^4 [\rightarrow^6] \leftrightarrow \leftarrow^3]^0 \leftarrow^3 \text{---} 0 \end{aligned}$$

and

$$\begin{aligned} \text{MIRROR} = 0 \text{---} \leftarrow^2 A \leftarrow^4 A \rightarrow^8 A \rightarrow^6 [A \rightarrow^2 A \leftarrow^8 A \rightarrow^2 [\rightarrow^6 [\leftarrow^4]^\dagger [\leftarrow^4]^\dagger \leftarrow^2]^+ \rightarrow^6 \\ [\leftarrow^4]^\dagger [\leftarrow^4]^\dagger A \leftarrow^6 A \leftarrow^2 [\rightarrow^4]^\dagger [\rightarrow^4]^\dagger \rightarrow^6 A]^0 A \leftarrow^6 A \rightarrow^6 \\ [\leftarrow^8]^0 A \leftarrow^{10} [\leftarrow^4]^\dagger A \leftarrow^6 [\rightarrow^8]^0 \rightarrow^2 A \rightarrow^2 \text{---} 0. \end{aligned}$$

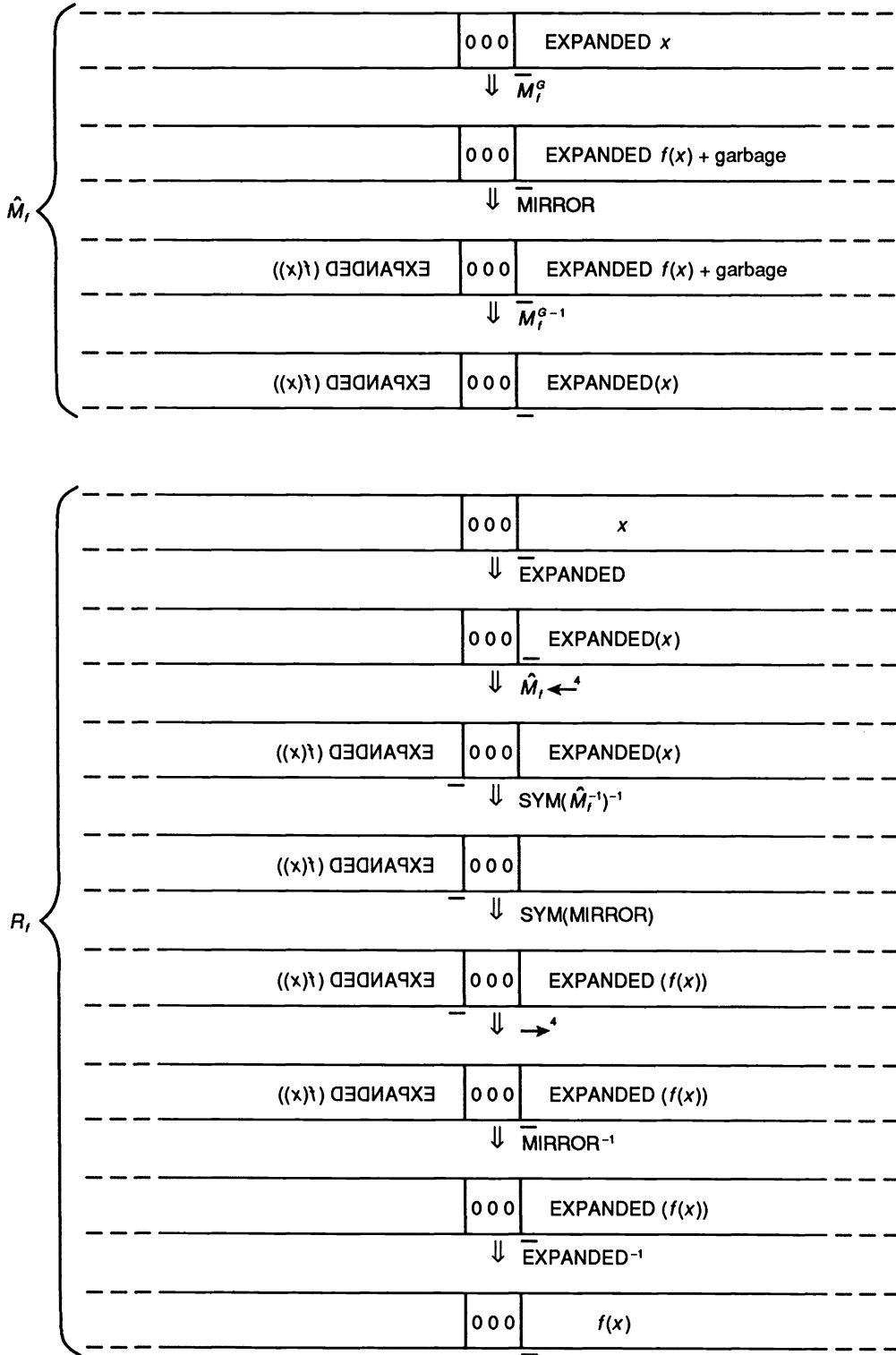
Suppose

$$(5.5.1) \quad \hat{M}_f = 0 \text{---} M_f^G \text{MIRROR} (M_f^G)^{-1} \text{---} 0.$$

We can define

$$(5.5.2) \quad R_f = 0 \text{---} \text{EXPANDED} \hat{M}_f \leftarrow^4 \text{SYM} (\hat{M}_{f^{-1}})^{-1} \text{SYM} (\text{MIRROR}) \rightarrow^4 \text{MIRROR}^{-1} \text{EXPANDED}^{-1} \text{---} 0.$$

We can represent $\hat{M}_f$ and R_f in the following manner:



Note 5.6. If we construct the reversible machine that computes f as suggested in (5.5.1) and (5.5.2) the computation time of R_f is about two times the computation time of M_f^G and $M_{f^{-1}}^G$ (plus the computation time of EXPANDED and MIRROR at the most proportional to $|x|^2$ or $|f(x)|^2$).

6. Existence of one-way bijections. We will first define the HB set of the polynomially honest bijections.

DEFINITION 6.1. $f \in \text{HB} \Leftrightarrow f \in \text{I}$, f is total and there is a polynomial p such that

$$(6.1.1.) \quad |f(x)| \leq p(|x|), p(|x|) \leq |f(x)| \text{ for each } x \in \mathbb{N}.$$

Notation 6.1.2. If $X \subseteq \text{HB}$, with X^{-1} we indicate the set of inverse bijections:

$$f^{-1} \in X^{-1} \Leftrightarrow f \in X.$$

We now define three subsets of HB: HBP, HBRP, HBNP.

DEFINITION 6.2. One bijection $f \in \text{HB}$ belongs also to HBP [HBRP, HBNP] if there is an ordinary [reversible, nondeterministic] Turing machine M_f and a polynomial p such that:

- (1) for each $\langle x, y \rangle \in \mathbb{N}^2$ there is a computation C of M_f that transforms the input $\langle\langle x \rangle\rangle$ in the output $\langle\langle y \rangle\rangle$ if and only if $y = f(x)$;
- (2) the computation time is less than or equal to $p(|x|)$ for at least a computation C .

Note 6.2.1. $\langle\langle x \rangle\rangle$ is to be understood as defined in 5.4. The “if and only if” at point (1) indicates that M_f (even if nondeterministic) must not compute two distinct values starting from the same x . The condition “for at least one computation” at point (2) is necessary (meaningful) only if M_f is nondeterministic.

Obviously we have

$$(6.3.1) \quad \text{HBRP} \subseteq \text{HBP},$$

$$(6.3.2) \quad \text{HBP} \subseteq \text{HBNP},$$

and furthermore

$$(6.3.3) \quad \text{HBRP} = \text{HBRP}^{-1}.$$

$$(6.3.4) \quad \text{HBNP} = \text{HBNP}^{-1}$$

(in fact $M_{f^{-1}} = M_f^{-1}$ if M_f is a reversible or nondeterministic machine, see Proposition 2.4, and for Definition 6.1 there is no difference between polynomiality in regard to the range and the value of f .)

On the other hand, we do not know whether

$$(6.3.5) \quad \text{HBP} = \text{HBP}^{-1}?$$

The set $\text{B2V} = \text{HBP} \cap \text{HBP}^{-1}$ is known as the class of the two-way bijections, i.e., computable in two directions in polynomial time.

$\text{B1V} = \text{HBP} - \text{HBP}^{-1}$ is known as the class of one-way bijections, i.e., computable in polynomial time in one direction but not in the other.

THEOREM 6.4. $\text{B2V} = \text{HBRP}$.

The two-way bijections are all and only those computable in polynomial time by a reversible Turing machine.

Proof. if $f \in \text{HBRP}$ for 6.3.1 and 6.3.3 then $f \in \text{HBP} \cap \text{HBP}^{-1}$; if $f \in \text{HBP} \cap \text{HBP}^{-1} = \text{B2V}$, for § 5.5, Note 5.6, and Proposition 4.9 then $f \in \text{HBRP}$.

The problem of the existence of one-way bijections ($\text{B1V} \neq \emptyset$?) is still open (it is also connected to the problem $P = NP$). For us it is not solved, but can be reduced to

the following: “HBP $\neq$ HBRP?” (Are there polynomially honest bijections polytime computable by an ordinary Turing machine but not by a reversible Turing machine?), which in turn is equivalent to the problem (6.3.5).

The practical meaning of Theorem 6.4 can be illustrated as follows: let us suppose that f is a bijection easy to compute in one direction, obviously $f \in \text{HBP}$, but difficult to compute (at least intuitively) in the other direction, so that one doubts its bipolynomiality: $f^{-1} \in \text{HBP}$? We therefore do not know whether $f \in \text{B2V}$ or $f \in \text{B1V}$. To try to establish it, it is not necessary to deal with the difficult inverse problem, i.e., to establish whether $f^{-1} \in \text{HBP}$ or $f^{-1} \in \text{HBNP} - \text{HBP}$. In fact to try to prove that $f \in \text{B2V}$ all that is necessary is to try to program the direct problem by means of the construction⁴ of a reversible Turing machine that computes it in polynomial time. If one succeeds in this, apart from having resolved such a problem, it is possible to obtain with only the aid of a mirror a polynomial procedure for the inverse problem. Obviously to prove the contrary ($f \in \text{B1V}$) it will be necessary to establish that a function f , computable in polynomial time by a deterministic Turing machine, is not computable, always in polynomial time, by any reversible Turing machine. This would no doubt be more difficult, but if it were done the result would be striking. In fact if it resulted

$$f \in \text{HBP} - \text{HBRP} \quad (f \in \text{B1V} \neq \emptyset),^5$$

furthermore for problems 6.3.2 and 6.3.4 and Theorem 6.4, one would have

$$f^{-1} \in \text{HBNP} - \text{HBP}.$$

Let Π be the predicate so defined

$$\Pi[(2x+1)2^y] = \begin{cases} \text{true} \\ \text{false} \end{cases} \Leftrightarrow \lfloor f^{-1}(x)/2^y \rfloor \text{ is } \begin{cases} \text{odd} \\ \text{even}. \end{cases}$$

It would result $\Pi \in \text{NP} \cap \text{CoNP} - \text{P}$ (result even stronger than $\Pi \in \text{NP} - \text{P}$, i.e., $\text{P} \neq \text{NP}$). In fact, it is easy to prove

$$f^{-1} \in \text{HBNP} \Rightarrow \Pi \in \text{NP} \cap \text{CoNP}$$

$$\Pi \in \text{P} \Rightarrow f^{-1} \in \text{HBP}.$$

REFERENCES

- [1] C. H. BENNETT, *Logical reversibility of computation*, IBM J. Res. Develop., 6 (1973), pp. 525–532.
- [2] H. HERMES, *Enumerability, Decidability, Computability*, Springer-Verlag, Heidelberg, New York, 1969.
- [3] K. KO, *On some natural complete operators*, TCS, 37 (1985), pp. 1–30.
- [4] H. R. LEWIS AND C. H. PAPADIMITRIOU, *Element of the Theory of Computation*, Prentice-Hall, Englewood Cliffs, NJ, 1981.
- [5] R. P. FEYNMAN, *Quantum Mechanical Computers*, California Institute of Technology, Pasadena, CA, 1984.
- [6] T. TOFFOLI, *Computation and construction universality of reversible cellular automata*, J. Comput. System Sci., 15 (1977), pp. 213–231.

⁴ Obviously one cannot use the general method of the proof of the Theorem 5.5, because it assumes the knowledge of $M_{f^{-1}}^G$, but one must directly construct R_f .

⁵ It is known that $\text{B1V} \neq \emptyset \Rightarrow \text{P} \neq \text{NP} \cap \text{CoNP}$.

MONOTONE CIRCUITS FOR CONNECTIVITY REQUIRE SUPER-LOGARITHMIC DEPTH*

MAURICIO KARCHMER†‡ AND AVI WIGDERSON†

Abstract. It is proved here that every monotone circuit which tests st -connectivity of an undirected graph on n nodes has depth $\Omega(\log^2 n)$. This implies a superpolynomial ($n^{\Omega(\log n)}$) lower bound on the size of any monotone formula for st -connectivity.

The proof draws intuition from a new characterization of circuit depth in terms of communication complexity. Within the same framework, a very simple and intuitive proof is given of a depth analogue of a theorem of Khrapchenko concerning formula size lower bounds.

Key words. circuit complexity, communication complexity, monotone circuits, graph connectivity

AMS(MOS) subject classifications. 06E30, 94A05, 94C10

1. Introduction. The circuit complexity of Boolean functions has been studied for the past forty years, but its main problem remains unsolved: we have no example of a simple function (say in NP) that requires super linear circuit size or super logarithmic (bounded fan-in) circuit depth. The reason is, perhaps, that although the circuit model is elegantly simple, our understanding of the way it computes is, at the most, vague.

In the last years, however, advance has been surprisingly fast. On the one hand, results of Andreev [An] and Razborov [Ra], improved by Alon and Boppana [AB], give exponential size lower bounds for monotone circuits. On the other hand, results of a long list of authors (e.g., [Aj], [FSS], [Y1], [H]) give exponential size lower bounds for constant depth circuits. More than the results themselves, perhaps the main contribution of the mentioned papers has been the development of some general techniques for proving lower bounds, such as random restrictions and circuit approximation. These techniques, however, turn out to be difficult to apply to other problems so that new ideas have been sought.

In this paper we show the equivalence between circuit depth and the communication complexity of a certain related problem.¹ We believe that the later model is much more appealing for both showing and understanding upper bounds, as well as for proving lower bounds. This characterization is reminiscent of, but somehow more explicit and intuitive than, the well known relationship between circuits and alternating machines [Ru]. This characterization allows us to view computation top-down (from output to input) and apply such techniques as random restrictions in that direction (rather than the common bottom-up approach). We argue the relevance of this model by presenting a very simple proof of a theorem of Khrapchenko, and by proving the first super-logarithmic (in the size of the circuit) depth lower bound for monotone circuits.

Though the mentioned results of Andreev and Razborov give exponential (in $\log n$) depth lower bounds for monotone circuits computing certain functions, the depth lower bound is always logarithmic in the size bound. That is, the techniques apply to size rather

* Received by the editors March 1, 1989; accepted for publication in revised form June 9, 1989.

† Institute of Mathematics and Computer Science, Hebrew University, Jerusalem, Israel 91904. Part of this work was done while the authors were visiting the University of California at Berkeley and was supported by National Science Foundation grant DCR-8612563. The second author also wishes to acknowledge support from the Alon Fellowship and an American-Israeli Binational Science Foundation grant.

‡ Present address, Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

¹ Yannakakis independently discovered this equivalence [KPPY].

than to depth. We present a technique which captures, in a strong way, the essence of circuit depth. We give here a tight $\Omega(\log^2 n)$ depth bound for st -connectivity,² a function which has $O(n^2 \log n)$ size, $O(\log^2 n)$ depth monotone circuits. As a consequence, we get nonpolynomial ($n^{\Omega(\log n)}$) size lower bounds for monotone formulas computing st -connectivity and hence separate the monotone analogues of NC^1 and AC^1 .

While our proof bears no obvious similarity to the methods of Razborov and Andreev, we point out the important role that (different) nontrivial results from extremal set theory play in both cases.

It is interesting to note here the different character of the connectivity and majority functions in the Boolean and arithmetic monotone circuits models. Shamir and Snir [ShS] showed an $\Omega(\log^2 n)$ depth bound for both functions in the arithmetic model. The difficulty in applying these techniques to Boolean circuits are the axioms $x \vee xy = x$ and its dual, which do not hold in rings. Indeed, Valiant [V] (by probabilistic methods) and Ajtai, Komlos, and Szemerédi [AKS] (by explicit constructions) showed that these axioms make a difference for the majority function which admits $O(\log n)$ depth monotone Boolean circuits. Our result says that, unlike for majority, for connectivity the situation in the Boolean case is very similar to the arithmetic one.

It is worthwhile to mention that our results apply to undirected graph st -connectivity, a function that, in some models, is easier than its directed version. For example, see [AKLLR] for some relevant evidence. More recently, Ajtai and Fagin [AF] show that, while undirected st -connectivity is definable in monadic second order logic, the directed case is not.

The paper is organized as follows: In § 2 we define the communication game and show its equivalence to circuit depth; in § 3 we give a simple proof of a theorem of Khrapchenko. In § 4 we give the lower bound for connectivity.

2. Communication complexity and circuit depth. In this section we show the equivalence between circuit depth and a problem in communication complexity. We will be considering circuits over the basis $\{\vee, \wedge, \neg\}$ where $\{\vee, \wedge\}$ -gates have fan-in 2 and $\neg$ -gates are only applied to input variables. For a function f , $d(f)$ is the minimum depth of a circuit computing f .

Let $B_0, B_1 \subseteq \{0, 1\}^n$ such that $B_0 \cap B_1 = \emptyset$. Consider the following game between players I and II: Player I gets $x \in B_1$ while player II gets $y \in B_0$; their goal is to find a coordinate i such that $x_i \neq y_i$. Let $C(B_1, B_0)$ be the minimum number of bits they have to communicate in order for both to agree on such a coordinate. Note that unlike standard problems in communication complexity [Y1], the task of the players here is to solve a search, rather than a decision, problem.

THEOREM 2.1. *For every function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ we have*

$$d(f) = C(f^{-1}(1), f^{-1}(0)).$$

Proof. The proof follows from the following two lemmas. $\square$

LEMMA 2.1. *For all functions f and all $B_0, B_1 \subseteq \{0, 1\}^n$ such that $B_0 \subseteq f^{-1}(0)$ and $B_1 \subseteq f^{-1}(1)$ we have*

$$C(B_1, B_0) \leq d(f).$$

Proof. The proof is shown by induction on $d(f)$.

If $d(f) = 0$ then f is either x_i or $\bar{x}_i$. In either case, we have that for all $x \in B_1$ and $y \in B_0$, $x_i \neq y_i$ so that i is always an answer and $C(B_1, B_0) = 0$.

² We present here an improved and simplified version of an early result of ours giving a $\Omega(\log^2 n / \log \log n)$ bound. This was possible after Hastad formulated and proved Lemma 4.1. A similar improvement was independently discovered by Boppana.

For the induction step we suppose that $f = f_1 \wedge f_2$ (the case $f = f_1 \vee f_2$ is treated similarly) so that $d(f) = \max(d(f_1), d(f_2)) + 1$. Let $B_0^j = B_0 \cap f_j^{-1}(0)$ for $j = 1, 2$. By induction we have that $C(B_1, B_0^j) \leq d(f_j)$ for $j = 1, 2$. Consider the following protocol for B_1 and B_0 : player II sends a 0 if $y \in B_0^1$, otherwise he sends a 1; the players then follow the best protocol for each of the subcases. We have

$$C(B_1, B_0) \leq 1 + \max_{j=1,2} (C(B_1, B_0^j)) \leq 1 + \max_{j=1,2} (d(f_j)) = d(f). \quad \square$$

The converse is as follows.

LEMMA 2.2. *Let $B_0, B_1 \subseteq \{0, 1\}^n$ such that $B_0 \cap B_1 = \emptyset$. Then, there exists a function f with $B_0 \subseteq f^{-1}(0)$ and $B_1 \subseteq f^{-1}(1)$ such that*

$$d(f) \leq C(B_1, B_0).$$

Proof. The proof is shown by induction on $C(B_1, B_0)$.

If $C(B_1, B_0) = 0$ then there exists an i such that for all $x \in B_1$ and for all $y \in B_0$, $x_i \neq y_i$. It is clear that for all $x', x'' \in B_1$ we have $x'_i = x''_i$ and the same holds for all $y', y'' \in B_0$. Without loss of generality $x_i = 1$ so that letting $f = x_i$ we have $B_0 \subseteq f^{-1}(0)$ and $B_1 \subseteq f^{-1}(1)$.

To prove the induction step, we assume that player II sends the first bit (the other case is treated similarly). For some partition $B_0 = B_0^1 \cup B_0^2$, player II sends a 0 if $y \in B_0^1$, a 1 otherwise; the players then continue with the best protocol for each of the subcases and

$$C(B_1, B_0) = 1 + \max_{j=1,2} (C(B_1, B_0^j)).$$

By induction, there exist f_1, f_2 so that $B_0^j \subseteq f_j^{-1}(0)$, $B_1 \subseteq f_j^{-1}(1)$ and $d(f_j) \leq C(B_1, B_0^j)$ for $j = 1, 2$. Taking now $f = f_1 \wedge f_2$ we have

$$B_1 \subseteq f_1^{-1}(1) \cap f_2^{-1}(1) = f^{-1}(1)$$

$$B_0 = B_0^1 \cup B_0^2 \subseteq f_1^{-1}(0) \cup f_2^{-1}(0) = f^{-1}(0)$$

and

$$d(f) \leq 1 + \max_{j=1,2} (d(f_j)) \leq 1 + \max_{j=1,2} (C(B_1, B_0^j)) = C(B_1, B_0). \quad \square$$

For monotone circuits we can give a modified version of Theorem 2.1 that captures, in a nice way, the restrictions of monotone computation. A *minterm* (*maxterm*) of a monotone function f is a minimal set of variables where, if we set to 1 (0), f will be equal to 1 (0) regardless of the other variables. Define $\min(f)$, $\text{Max}(f)$ as the set of minterms, respectively, maxterms, of f . It is easy to see that every minterm intersects every maxterm. We will look at minterms and maxterms as subsets of $[n]$.³ For a monotone function f , let $d_m(f)$ be the minimum depth of a monotone circuit computing f .

Consider the following communication game (the *monotone game*) between players I and II. Let $P, Q \subseteq 2^{[n]}$ be such that for all $p \in P$ and for all $q \in Q$ we have $p \cap q \neq \emptyset$. Player I gets a $p \in P$ while player II gets a $q \in Q$; their goal is to find an element in $p \cap q$. Let $C_m(P, Q)$ be the minimum number of bits they have to communicate in order to find such an element.

THEOREM 2.2. *For every monotone function f we have*

$$d_m(f) = C_m(\min(f), \text{Max}(f)).$$

³ $[n] = \{1, \dots, n\}$.

Proof. Note that in the base case of Lemma 2.1, if the circuit is monotone, we always find a coordinate i such that $x_i = 1$ while $y_i = 0$. On the other hand, if the protocol always gives a coordinate i with the above property, Lemma 2.2 gives a monotone circuit.

Let $x \in f^{-1}(1)$ be the characteristic vector of a subset $p \subseteq [n]$. Similarly, let $y \in f^{-1}(0)$ be the characteristic vector of the complement of a subset $q \subseteq [n]$ (i.e., $i \in q \Leftrightarrow y_i = 0$). By the above argument, it is clear that the answer of the protocol will be an element of $p \cap q$. The theorem follows by noticing that it is enough to give a protocol for $(\min(f), \text{Max}(f))$ because the players, in case they get inputs p' and q' , can always behave as if they got $p \subseteq p'$ and $q \subseteq q'$ where $p \in \min(f)$ and $q \in \text{Max}(f)$. $\square$

For proving lower bounds for the communication game, it may be convenient to have more structure in the way the players behave. We would like to synchronize the protocol so that the players communicate in rounds where players I and II send messages of fixed lengths.

THEOREM 2.3. *For any function f , there exists a protocol D where at each round player II sends 2^a bits while player I responds with a bits and such that the number k of rounds satisfies*

$$k \leq \frac{d(f)}{a}$$

for the general game and

$$k \leq \frac{d_m(f)}{a}$$

for the monotone one.

Proof. Let C be the best circuit for f . The idea is to simulate a layers of C with a round of D . Divide C into stages of depth a each and look at the subcircuits of each stage. Each one computes a function which depends on at most 2^a wires and, thus, can be represented in Conjunctive Normal Form (CNF) with at most 2^{2^a} clauses, each of length at most 2^a . Following the proof of Lemma 2.1, it is easy to see that such a CNF representation can be simulated by a round where player II sends 2^a bits and player I sends a bits. The same holds for the monotone case. $\square$

Of course, in Theorem 2.3, the roles of players I and II can be switched so that both players send 2^a and a bits per round, respectively.

3. Khrapchenko's bound. As a nice application of Theorem 2.1, we give a simple proof of a depth analogue of a theorem of Khrapchenko [K]. Let C_n be the graph of the n -cube with vertex set $\{0, 1\}^n$ and two nodes adjacent if and only if they differ in one coordinate. Any subset A of edges induces a graph G_A of C_n in the natural way. For a graph G_A and a node x , we denote $d_A(x)$, $N_A(x)$ as the degree of x in A and the set of neighbors of x in A , respectively. We drop the subindex A if no confusion arises. Let E denote expectation with uniform distribution.

THEOREM 3.1 (Khrapchenko). *Let $B_0, B_1 \subseteq \{0, 1\}^n$ such that $B_0 \cap B_1 = \emptyset$. Let $A = C_n \cap (B_0 \times B_1)$. Then, for every function f with $B_0 \subseteq f^{-1}(0)$ and $B_1 \subseteq f^{-1}(1)$ we have*

$$d(f) \geq \log \frac{|A|^2}{|B_0| |B_1|}.$$

Proof. Fix a protocol D for the communication game and let $C(x, y)$ be the number of bits D uses on inputs x, y . We will prove that for (x, y) taken uniformly from A

we have

$$(*) \quad E(C(x, y)) \geq \log \frac{|A|^2}{|B_0| |B_1|}$$

so that by Lemma 2.1, we get $d(f) \geq E(C(x, y))$.

We view $(*)$ as follows: Write

$$\log \frac{|A|^2}{|B_0| |B_1|} = \log \frac{|A|}{|B_0|} + \log \frac{|A|}{|B_1|}$$

and note that $|A|/|B_0|$ and $|A|/|B_1|$ are the average degrees of nodes in B_0 and B_1 , respectively. In what follows, we will claim that the number of bits player I sends is at least the logarithm of the average degree of nodes in B_1 (similarly with player II). Intuitively, this is so because even if player I knows y , he needs $\log d(y)$ bits to tell player II which x he has.

We now proceed formally. For $(x, y) \in A$, let $b_I(x, y)$, $b_{II}(x, y)$ be the number of bits player I and II send when the input to the protocol is (x, y) . We have

$$\begin{aligned} E(C(x, y)) &= \frac{1}{|A|} \left[\sum_{(x, y) \in A} (b_I(x, y) + b_{II}(x, y)) \right] \\ &= \frac{1}{|A|} \left[\sum_{x \in B_1} \sum_{y \in N(x)} b_{II}(x, y) + \sum_{y \in B_0} \sum_{x \in N(y)} b_I(x, y) \right]. \end{aligned}$$

We claim:

- For any $x \in B_1$, $\sum_{y \in N(x)} b_{II}(x, y) \geq d(x) \log d(x)$. This is so because, even if player II knows x , he has to tell player I which y he has.
- Similarly, for all $y \in B_0$ we have $\sum_{x \in N(y)} b_I(x, y) \geq d(y) \log d(y)$.

We now conclude that

$$\begin{aligned} E(C(x, y)) &\geq \frac{1}{|A|} \left[\sum_{x \in B_1} d(x) \log d(x) + \sum_{y \in B_0} d(y) \log d(y) \right] \\ &\geq \frac{1}{|A|} \left[\sum_{x \in B_1} \frac{|A|}{|B_1|} \log \frac{|A|}{|B_1|} + \sum_{y \in B_0} \frac{|A|}{|B_0|} \log \frac{|A|}{|B_0|} \right] = \log \frac{|A|^2}{|B_1| |B_0|} \end{aligned}$$

where the last inequality follows from the convexity of $x \log x$. $\square$

4. A lower bound for connectivity. In this section we give a $\Omega(\log^2 n)$ depth lower bound for monotone circuits computing undirected graph st -connectivity. This section is organized as follows: In § 4.1 we give some intuition and we state the main theorem; in § 4.2 we give some definitions and useful lemmas; finally, in § 4.3 we give the proof of the theorem.

4.1. Intuition. The function st -connectivity receives as input the adjacency matrix of an undirected graph with two distinguished vertices s and t , and tests whether there is a path from s to t or not. The function is obviously monotone with minterms corresponding to minimal st -paths and maxterms corresponding to minimal st -cuts. We view st -paths as ordered sets of vertices excluding s and t . We view st -cuts as partitions of the set of vertices into two subsets, one containing s and the other containing t . The minimal cut contains all edges between the two subsets. A partition may be regarded as a coloring $q: V \rightarrow \{0, 1\}$ where $q(s) = 0$ and $q(t) = 1$. The game is as follows: Player I gets an st -

path while player II gets a coloring of the nodes. Their goal is to find a bichromatic edge in the path.

Let us look at the protocol based on the idea of raising the adjacency matrix of the graph to the n th power: Player I sends the name of the middle vertex on his path; player II responds with the color of that vertex. The players then continue recursively on the half path where a bichromatic edge is ensured to exist. Note that the protocol requires $O(\log n)$ rounds in each of which player I sends $\log n$ bits and player II sends just one.

The crucial observation is that, even if player II were allowed to send $O(n^\epsilon)$ bits each round (instead of one bit as in the protocol), the players will still need many rounds. Basically, this is because player II doesn't know much about the nodes in player I's path. If he sends $O(n^\epsilon)$ bits and the path is of length $O(n^\epsilon)$, then the probability that player I gets valuable information from player II is negligible. If we could prove a $\Omega(\log n)$ lower bound for the number of rounds needed, we will be able to use Theorem 2.3 to get the promised $\Omega(\log^2 n)$ depth lower bound for circuits.

Note the asymmetry between players I and II. Indeed, if the roles of both players were switched so that player I would be the one who sends $O(n^\epsilon)$ bits per round, they would be able to solve the problem in a constant number of rounds. This is consistent with the intuition in Shamir and Snir in [ShS].

Define $stconn(l)$ as the restriction of st -connectivity to the case where player I gets a path of length l . We state the main theorem of this section.

THEOREM 4.1. *Suppose $l \leq n^{1/10}$. There exists an $0 < \epsilon < \frac{1}{2}$ such that if D is a k -round protocol for $stconn(l)$ where at each round player I sends $\epsilon \log n$ bits and player II sends n^ϵ bits, then $k \geq \log l$.*

COROLLARY 4.1. *The monotone depth complexity of st -connectivity is $\Omega(\log^2 n)$.*

Proof. The proof follows from Theorems 2.3 and 4.1 by taking $l = n^{1/10}$. $\square$

COROLLARY 4.2. *The monotone formula size complexity of st -connectivity is $n^{\Omega(\log n)}$.*

Proof. The proof follows by noting that the relation $d(f) = O(\log L(f))$ holds also in the monotone case [S]. $\square$

Theorem 4.1 is a consequence of the following theorem. Let $vector(l)$ be the game where player I gets an $(l+2)$ -vector p with $p[0] = s$ and $p[l+1] = t$ and other entries from $[n]$ and player II gets a coloring $q \in \{0, 1\}^n$ of $[n]$ extended so that $q(s) = 0$ and $q(t) = 1$. The goal of the players is to find an element $v \in [n]$ such that for some index i , $p[i] = v$ and either $q(p[i]) \neq q(p[i+1])$ or $q(p[i-1]) \neq q(p[i])$.

THEOREM 4.2. *Suppose $l \leq n^{1/10}$. There exist an $0 < \epsilon < \frac{1}{2}$ such that if D is a k -round protocol where at each round player I sends $\epsilon \log n$ bits and player II sends n^ϵ bits, and D solves the game $vector(l)$ for a fraction $\frac{1}{8}n^{-\epsilon}$ of the possible vectors, then $k \geq \log l$.*

Note that in $stconn(l)$ the players are asked to find a bichromatic edge, while in $vector(l)$ they are asked to find an endpoint of a bichromatic edge. Also, in $stconn(l)$, player I receives a simple path which can be viewed as a vector without repetitions. Given this, Theorem 4.1 follows by noting that, by our choice of l , a protocol for $stconn(l)$ solves the game $vector(l)$ for a fraction $1 - o(1)$ of the vectors.

To prove Theorem 4.2, we will assume, for contradiction, the existence of a k -round protocol ($k < \log l$) good for a large family of all possible vectors and a large family of all possible colorings. We will pick a large subset of the vectors and colorings for which players I and II sent the same message in the first round. We will give some extra information (by applying a random restriction to the coloring of the nodes) to both players so as to get smaller, yet nicer, subsets which are in 1-1 correspondence with a family of vectors shorter in length (but of higher quality) and a family of colorings of fewer nodes. The fact that the original protocol had $(k-1)$ rounds to go will allow us to find a $(k-1)$ -round protocol for the smaller families. Repeating this k times will give us a

protocol without communication that solves a problem which cannot be solved without any messages.

Note the top-down structure of the proof; essentially, the argument shows that, if the output of a shallow circuit depending on a set of wires is in some sense complex, then there is a wire which computes a complex subfunction. This is similar to the argument used in [KPPY].

4.2. Notation and definitions. Let $[n]^l$ denote the set of all l -vectors with entries from $[n]$. An *interval* $I \subseteq [l]$ is a subset of consecutive integers. For a vector $p \in [n]^l$ and an interval $I \subseteq [l]$, p_I is the *projection* of p into the interval I . For $P \subseteq [n]^l$, $P_I = \{p_I : p \in P\}$ is the projection of P into I . Note that $P_I \subseteq [n]^{|I|}$. Conversely, for $p \in [n]^{|I|}$, $P \subseteq [n]^l$ and an interval I , let $\text{Ext}_{P,I}(p) = \{\tilde{p} \in P : \tilde{p}_I = p\}$ be the set of extensions of p in P within I . We will drop the subindices P and I if no confusion arises. For $p \in [n]^l$, the *support* of p , $\text{supp}(p)$, is defined as the set of nodes contained in p . When no confusion arises, we will denote $\text{supp}(p)$ by p . Given a partition of $[l]$ into two intervals L and R , we will denote a vector $p \in [n]^l$ by (p_L, p_R) where each entry is the projection of p into the respective interval.

Similarly, for a coloring $q \in \{0, 1\}^n$ and a subset $T \subseteq [n]$, q_T is the projection of q into T and for $Q \subseteq \{0, 1\}^n$, $Q_T = \{q_T : q \in Q\}$ is the projection of Q into T . For $q \in \{0, 1\}^{|T|}$, $Q \subseteq \{0, 1\}^n$ and a subset $T \subseteq [n]$, let $\text{Ext}_{Q,T}(q) = \{\tilde{q} \in Q : \tilde{q}_T = q\}$ be the set of extensions of q in Q within T (again, we drop the subindices Q and T whenever possible). For a restriction $\rho : [n] \mapsto \{0, 1, *\}$, we will denote by Q^ρ the set of colorings in Q consistent with ρ , (i.e., $\{q \in Q : \rho(i) \neq * \Rightarrow \rho(i) = q_i\}$).

For a subset A of a universe Ω , the density of A , $\mu_\Omega(A)$, is defined as $|A|/|\Omega|$. In what follows, we will work with densities rather than with cardinalities. If Ω is clear from the context, we will drop the subindex and write $\mu(A)$. The reader should be aware that we may mix densities with respect to different universes in the same equation.

We will need the following combinatorial lemma due to Hastad: Let $H \subseteq A_1 \times \cdots \times A_k$ and for $v \in A_i$, let $\text{Ext}_{A_i}(v) = \{u \in H : u_i = v\}$. Note that, though $\text{Ext}_{A_i}(v) \subseteq H$, $\text{Ext}_{A_i}(v)$ will be considered as a subset of $H/A_i = A_1 \times \cdots \times A_{i-1} \times A_{i+1} \times \cdots \times A_k$ and, in what follows, its density will be defined with respect to H/A_i .

LEMMA 4.1. *Let $H \subseteq A_1 \times \cdots \times A_k$. Let $B_i = \{u \in A_i : \mu(\text{Ext}_{A_i}(u)) \geq \mu(H)/2k\}$. Then*

$$\prod_{i=1}^k \mu(B_i) \geq \frac{\mu(H)}{2}.$$

Proof. Say that a member $(u_1, \dots, u_k)$ of H is bad if for some i , $u_i \notin B_i$. Let $\bar{H}$ be the set of bad elements in H . We have

$$\mu(\bar{H}) \leq \sum_{i=1}^k \mu\left(\bigcup_{u \notin B_i} \text{Ext}_{A_i}(u)\right) < \sum_{i=1}^k \frac{\mu(H)}{2k} = \frac{\mu(H)}{2};$$

the lemma follows immediately, by noting that

$$\prod_{i=1}^k \mu(B_i) \geq \mu(H) - \mu(\bar{H}). \quad \square$$

COROLLARY 4.3. *If $k = 2$, then there exists an i such that $\mu(B_i) \geq (\mu(H)/2)^{1/2}$.*

COROLLARY 4.4. *$\Pr(\mu(B_i) < (\mu(H)/2)^{2/k}) < \frac{1}{2}$ for i chosen randomly from $\{1, \dots, k\}$.*

4.3. The proof.

Proof of Theorem 4.2. In what follows, all our protocols will be synchronized so that at each round player I sends $\varepsilon \log n$ bits and player II responds with n^ε bits. The existence of ε will be clear from the proof, though we can check that $\varepsilon = \frac{1}{10}$ suffices. We will define a sequence of problems of different sizes as follows: We first define the parameters of the problems, let $t_{\max} = \log l - 1$.

Let $n_0 = n$ and $n_{t+1} = n_t - 4n_t^{1/2}$. Note that

$$(1) \quad n/2 \leq n_t \leq n \quad \text{for } t \leq t_{\max}.$$

Let $l_0 = l$ and $l_{t+1} = l_t/2$ and note that

$$(2) \quad 2 \leq l_t \leq l \quad \text{for } t \leq t_{\max}.$$

Consider the following property:

H(t, k). *There exist a collection of vectors $P^t \subseteq [n_t]^{l_t}$ of length l_t , and a collection of colorings $Q^t \subseteq \{0, 1\}^{n_t}$ of $[n_t]$, with $\mu(P^t) \geq \frac{1}{8}n^{-\varepsilon}$ and $\mu(Q^t) \geq 2^{-2t\varepsilon}$ such that there exists a k -round protocol D^t good for (P^t, Q^t) .*

We will prove the following two claims:

CLAIM 4.1. *For $t \leq t_{\max} \neg H(t, 0)$.*

CLAIM 4.2. *For $t \leq t_{\max} H(t, k) \rightarrow H(t+1, k-1)$.*

It is clear that the two claims imply $\neg H(0, t_{\max})$ which in turn implies our theorem.

The first claim follows easily by noticing that there is not a single node (other than s and t) which appears in every vector of P^t so that player II cannot know the answer. To see this, note that the fraction of vectors of length l_t which contain a given node is $1 - (1 - 1/n_t)^{l_t} \leq \frac{1}{8}n^{-\varepsilon}$. This is enough for proving the claim as both players must know the answer. However, it can also be shown that, for most input pairs, player I will not know the color of a single node in its vector.

The second claim will be proved by assuming $H(t, k)$ and constructing P^{t+1} , Q^{t+1} , and D^{t+1} so as to satisfy $H(t+1, k-1)$. Take P^t , Q^t , and D^t which satisfy $H(t, k)$. Let us look at the protocol after the first round. By the pigeonhole principle, there exist $P \subseteq P^t$ with $\mu(P) \geq \frac{1}{8}n^{-2\varepsilon}$ such that for every vector in P , player I sent the same message. Similarly, there exists $Q \subseteq Q^t$ with $\mu(Q) \geq 2^{-(2t+1)\varepsilon}$ so that for every coloring of Q , player II sent the same message.

Let $L = \{1, \dots, l_t/2\}$ and $R = \{l_t/2 + 1, \dots, l_t\}$ be a partition of the vector's coordinates into left and right intervals of the same length. We say that P is L -good if many left projections of P have, each, many extensions to the right; that is, if

$$\mu(\{p_L : \mu(\text{Ext}_{P,L}(p_L)) \geq n^{-2\varepsilon}/32\}) \geq \frac{1}{4}n^{-\varepsilon}.$$

R -goodness is defined similarly. The following lemma states that if we shrink the length of the vectors to half and we restrict our family P to one of the intervals, then we can improve the quality of our collection. This is one of our main ideas: Although we cannot raise the absolute size of P , by reducing the size of the universe we can increase its density.

LEMMA 4.2. *P is either L -good or R -good.*

Proof. The lemma follows using Corollary 4.3 and viewing P as a subset of $[n_t]^{l_{t+1}} \times [n_t]^{l_{t+1}}$. $\square$

Without loss of generality, assume that P is L -good and let A be the set of vectors in P_L with many extensions. The next lemma is the heart of our argument.

LEMMA 4.3. *There exists a restriction $\rho : [n_t] \mapsto \{0, 1, *\}$ with $|\rho^{-1}(*)| = n_{t+1}$ such that the following properties hold:*

G1: $\mu(Q_{\rho^{-1}(*)}^\rho) \geq 2^{-2(t+1)n^\epsilon}$.

G2: $\exists \bar{P} \subseteq P$ such that

- $\forall p \in \bar{P}, p_L \subseteq \rho^{-1}(*)$ and $p_R \subseteq \rho^{-1}(1)$
- $\forall p, p' \in \bar{P}, p_L \neq p'_L$
- $\mu(\bar{P}_L) \geq \frac{1}{8}n^{-\epsilon}$.

Assuming the lemma is true, we will finish the proof of the second claim.

Let $Q^{t+1} = Q_{\rho^{-1}(*)}^\rho$ and $P^{t+1} = \bar{P}_L$ and rename the coordinates so that $[n_{t+1}] = \rho^{-1}(*)$. Note that there is a natural 1-1 correspondence between Q^{t+1} and Q^ρ and between P^{t+1} and $\bar{P}$. Also note that for all $q \in Q^\rho$ and for all $p \in \bar{P}$ any bichromatic edge lies in the interval L . The protocol D^{t+1} on (P^{t+1}, Q^{t+1}) simulates the remaining rounds of the protocol D^t on $(\bar{P}, Q^\rho)$ by following the behavior of the associated vector and coloring. $\square$

Proof of Lemma 4.3. In what follows, we denote $V = [n_t]$, $v = n_t$, $l = l_t$, $l' = l_{t+1}$ for simplicity. The existence of a good restriction will be shown by probabilistic methods. We will pick ρ uniformly from the set of all restrictions with $|\rho^{-1}(*)| = v - 4\sqrt{v}$ and $\Pr(\rho(x) = 0 | \rho(x) \neq *) = \frac{1}{2}$, and show that, with positive probability, the conditions of the lemma are fulfilled. Specifically, we will show that $\Pr(\neg \mathbf{G1}) + \Pr(\neg \mathbf{G2}) \leq \frac{1}{2} + o(1)$.

Let us start with **G1**: Intuitively, the following lemma says that, with high probability, ρ does not give player I too much information about the colors of nodes in $\rho^{-1}(*)$.

LEMMA 4.4. $\Pr(\mu(Q_{\rho^{-1}(*)}^\rho) < 2^{-(2t+1)n^\epsilon}) \leq \frac{1}{2} + o(1)$.

Proof. Let $\alpha = 2^{-(2t+1)n^\epsilon}$. Picking ρ uniformly from all restrictions with $|\rho^{-1}(*)| = v - 4\sqrt{v}$, is equivalent to picking randomly $T = \rho^{-1}(1) \cup \rho^{-1}(0)$ among all $4\sqrt{v}$ -subsets of V , and then picking the restriction of ρ to T randomly among all vectors x in $\{0, 1\}^{4\sqrt{v}}$. Let $k = \sqrt{v}/4$. Say T is *bad* if

$$\mu\left(\left\{x : \mu(\text{Ext}_{Q,T}(x)) \geq \frac{\alpha}{2k}\right\}\right) < \left(\frac{\alpha}{2}\right)^{2/k},$$

T is *good* otherwise. We have

$$\Pr\left(\mu(Q_{\rho^{-1}(*)}^\rho) < \frac{\alpha}{2k}\right) \leq \Pr(T \text{ is bad}) + \Pr\left(\mu(Q_{\rho^{-1}(*)}^\rho) < \frac{\alpha}{2k} \mid T \text{ is good}\right).$$

Note that $Q_{\rho^{-1}(*)}^\rho = \text{Ext}_{Q,T}(x)$. By the definition of goodness, and the choice of k , the second term is bounded by $1 - (\alpha/2)^{2/k} = o(1)$. Also note that $\alpha/2k \geq 2^{-2(t+1)n^\epsilon}$. It remains to bound the first term: We pick a random T by first picking a random partition of V into $4\sqrt{v}$ -subsets and then picking a random subset from the partition. We must show that for any partition, $\Pr(T \text{ is bad}) < \frac{1}{2}$ for a random T in the partition. But this is precisely the content of Corollary 4.4. $\square$

Now we take care of **G2**.

Let $A^* = \{p \in A : p \subseteq \rho^{-1}(*)\}$. We say that ρ *kills* a vector $p_L \in A$ if there is no $p_R \in \text{Ext}(p_L)$ with $p_R \subseteq \rho^{-1}(1)$. We will show that for every choice of ρ , $\mu(A^*)$ is large and hence $\Pr(\neg \mathbf{G2}) \leq \Pr(\exists p_L \in A \text{ killed by } \rho)$.

CLAIM 4.3. *For every ρ , $\mu(A^*) \geq \frac{1}{8}n^{-\epsilon}$.*

Proof. Recall that $|\rho^{-1}(*)| = v - 4\sqrt{v}$ so that $|V \setminus \rho^{-1}(*)| = 4\sqrt{v}$. It is easy to see that at most a fraction $1 - [(v - 4\sqrt{v})/v]^{l'} \leq \frac{1}{8}n^{-\epsilon}$ of the vectors in $[v]^{l'}$ intersect $V \setminus \rho^{-1}(*)$ so that at least a fraction $\frac{1}{4}n^{-\epsilon} - \frac{1}{8}n^{-\epsilon} \geq \frac{1}{8}n^{-\epsilon}$ of them are in A^* . $\square$

We now use some combinatorics to bound the probability that there exists a vector in A killed by ρ .

CLAIM 4.4. $\Pr(\exists p_L \in A \text{ killed by } \rho) = o(1)$.

Proof. We have

$$\Pr(\exists p_L \in A \text{ killed by } \rho) \leq |A| \cdot \max_{p_L \in A} \{\Pr(p_L \text{ is killed by } \rho)\}$$

so let us look at the worst possible $p_L \in A$. Let $F \equiv (p_L \text{ is killed by } \rho)$. Note that $\Pr(F)$ depends only on $\rho^{-1}(1)$. We pick $\rho^{-1}(1)$ as follows: Pick a number t between 0 and $4\sqrt{v}$ according to the binomial distribution (i.e., $\Pr(t = i) = \binom{4\sqrt{v}}{i} 2^{-4\sqrt{v}}$). If $t < \sqrt{v}$, we assume that F fails. Otherwise, we pick a subset $T = \rho^{-1}(1)$ where $|T| = t$ by choosing $\sqrt{v}/l'$ independent random vectors $y_1 \cdots y_{\sqrt{v}/l'}$ from $[v]^{l'}$, putting all nodes in these vectors in T , and adding enough random nodes so that $|T| = t$. It is clear that we are simulating our original distribution on $\rho^{-1}(1)$. We can now estimate $\Pr(F)$ by

$$\begin{aligned} \Pr(F) &\leq \Pr(t < \sqrt{v}) + \Pr(F | t \geq \sqrt{v}) \\ &\leq (2/e)^{\sqrt{v}} + \Pr(\forall i y_i \notin \text{Ext}(p_L)) \\ &\leq (2/e)^{\sqrt{v}} + (1 - \mu(\text{Ext}(p_L)))^{\sqrt{v}/l'} \\ &\leq \exp(-n^{1/5}) \end{aligned}$$

where we are using Chernoff's bound to estimate $\Pr(t < \sqrt{v})$ [Ch].

Recalling that $|A|$ is less than $n^{n^{1/10}}$, we easily conclude our calculations and get

$$\Pr(\exists p \in A \text{ killed by } \rho) \leq n^{n^{1/10}} \cdot \exp(-n^{1/5}) = o(1).$$

We have $\Pr(\neg G1) + \Pr(\neg G2) \leq \frac{1}{2} + o(1)$ implying the existence of a good restriction. Take any consistent extension of each $p_L \in A^*$ not killed by ρ to form $\bar{P}$. We have $\mu(\bar{P}_L) \geq \frac{1}{8} n^{-\epsilon}$ and Lemma 4.3 is proved. $\square$

Acknowledgments. We are very grateful to Miklós Ajtai for a conversation which led to the proof of Theorem 4.1. We are indebted to Johan Hastad for allowing us to present his improvements of our early results, and to Ravi Boppana for simplifying Claim 4.4. We are also thankful to Uri Zwick for helping us improve a previous draft.

REFERENCES

- [AB] N. ALON AND R. BOPPANA, *The monotone circuit complexity of Boolean functions*, Combinatorica, 7 (1987), pp. 1–22.
- [An] A. E. ANDREEV, *On a method for obtaining lower bounds for the complexity of individual monotone functions*, Sov. Math. Dokl., 31 (1985), pp. 530–534.
- [Aj] M. AJTAI, Σ_1^1 -formulae on finite structures, Ann. Pure Appl. Logic, 24 (1983), pp. 1–48.
- [AF] M. AJTAI AND R. FAGIN, *Reachability is harder for directed than for undirected finite graphs*, Proceedings of 29th FOCS, White Plains, NY, 1988, pp. 358–367.
- [AKLLR] R. ALELIUNAS, R. M. KARP, R. J. LIPTON, L. LOVASZ, AND C. RACKOFF, *Random walks, universal traversal sequences, and the complexity of maze problems*, Proceedings of 20th FOCS, San Juan, Puerto Rico, 1979, pp. 218–223.
- [AKS] M. AJTAI, J. KOMLÓS, AND E. SZEMERÉDI, *An $O(n \log n)$ sorting network*, Proceedings 15th STOC, Boston, MA, 1983, pp. 1–9.
- [Ch] H. CHERNOFF, *A measure of the asymptotic efficiency for tests of a hypothesis based on the sum of observations*, Ann. Math. Statist., 23 (1952), pp. 493–509.
- [FSS] M. FURST, J. B. SAXE, AND M. SIPSER, *Parity circuits and the polynomial time hierarchy*, Proceedings of 22nd FOCS, Nashville, TN, 1981, pp. 260–270.

- [H] J. HASTAD, *Improved lower bounds for small depth circuits*, Proceedings of 18th STOC, Berkeley, CA, 1986, pp. 6–20.
- [KPPY] M. KLAWE, W. J. PAUL, N. PIPPENGER, AND M. YANNAKAKIS, *On Monotone formulae with restricted depth*, Proceedings of 16th STOC, Washington, DC, 1984, pp. 480–487.
- [K] V. KHRAPCHENKO, *A method of determining lower bounds for the complexity of Π schemes*, Math. Notes Acad. Sci. USSR, (2) 1971, pp. 474–479.
- [Ra] A. A. RAZBOROV, *Lower bounds for the monotone complexity of some Boolean functions*, Sov. Math. Dokl., 31 (1985), pp. 354–357.
- [Ru] W. L. RUZZO, *Tree-size bounded alternation*, J. Comput. System Sci., 21 (1980), pp. 218–235.
- [ShS] E. SHAMIR AND M. SNIR, *On the depth complexity of formulas*, Math. Systems Theory, 13 (1980), pp. 301–322.
- [S] P. M. SPIRA, *On time-hardware complexity tradeoffs for Boolean functions*, Proceedings of 4th Hawaii Symp. on System Sciences, 1971, pp. 525–527.
- [V] L. G. VALIANT, *Short monotone formulae for the majority function*, J. Algorithms, 5 (1984), pp. 363–366.
- [Y1] A. C.-C. YAO, *Some complexity questions related to distributive computing*, Proceedings of 11th STOC, Atlanta, GA, 1979, pp. 209–213.
- [Y2] ———, *Separating the polynomial-time hierarchy by oracles*, Proceedings of 26th FOCS, 1985, pp. 1–10.

MEDIANS AND MAJORITIES IN SEMIMODULAR LATTICES*

BRUNO LECLERC†

Abstract. Given a ν -tuple $(x_1, \dots, x_\nu)$ in a metric space (X, d) , a median is an element m of X minimizing the sum $\sum_i d(m, x_i)$. One of the basic facts about medians in a finite space is that their obtainment by the majority rule is characteristic of a distributive-like structure on X , called a median semilattice. It is shown here that, in the case where X is a finite lattice, the semimodularity property is characterized by a weaker relation between the medians and the majority rule. Some consequences of this result are investigated.

Résumé. Etant donné un ν -uple $(x_1, \dots, x_\nu)$ d'éléments d'un espace métrique (X, d) , une médiane est un élément m de X minimisant la quantité $\sum_i d(m, x_i)$. Pour X fini, un fait essentiel est que l'obtention des médianes par la règle majoritaire est caractéristique d'une structure semilatticielle de type distributif sur X . On montre que, si X est un treillis fini, la propriété de semimodularité se caractérise par un lien plus faible entre les médianes et la règle majoritaire. On aborde quelques conséquences de ce résultat.

Key words. semimodular lattice, median, majority rule

AMS(MOS) subject classifications. O6C10, 62H30

1. Introduction. Given a metric space (X, d) and a ν -tuple (profile) $P = (x_1, \dots, x_\nu)$ of elements of X , a *median* is any element m of X minimizing the remoteness $r(m) = \sum_i d(m, x_i)$. Medians are considered in statistics as central values; in operation research as solutions to optimal location problems; in social choice theory for the aggregation of preferences; and in mathematical taxonomy for the classification consensus problem (Barthélemy and Monjardet (1981), Barthélemy, Leclerc, and Monjardet (1986)).

The research of medians and the study of their properties generally lead to difficult problems, except in some especially good cases where the medians are strongly related with a classical aggregation method called the Condorcet, or majority rule. As a first example, we obtain the following fact used in statistics: if X is a linearly ordered set and d a metric on X such that $x < y < z$ implies $d(x, z) = d(x, y) + d(y, z)$, then a slight change of the place of an element $x \in X$ decreases the remoteness $r(x)$ if and only if x moves towards a majority of the elements of P . The precise determination of the medians then follows from this.

Two natural extensions of this situation are the case of trees (Jordan (1869), Zelinka (1968), and Slater (1978)) and the case where X is a finite distributive lattice, with the minimum path length metric on its (unvalued, undirected) covering graph as metric d (Barbut 1961). In turn, these two kinds of graphs are special classes of median graphs, while median graphs are the covering graphs of median semilattices (Avann (1961), Bandelt (1984)). The complete recognition of these cases has been achieved by Bandelt and Barthélemy (1984): the structure of median semilattice is exactly that required for the obtainment of medians by the majority rule. Several sets of preference models, or sets of classification models, may be endowed with such a structure. As noted above, they correspond to especially good instances since, until now, they are the only ones where the determination of medians is a theoretically (and frequently practically) solved problem (see the recent review of Barthélemy and Monjardet (1988)).

Indeed, in these results the important structure is that of distributive lattice: the properties of the median semilattices are due to the possibility of embedding them in distributive lattices with preservation of all their lattice and metric properties (Sholander

* Received by the editors September 28, 1988; accepted for publication (in revised form) August 15, 1989.

† Centre d'Analyse et de Mathématique Sociales, Ecole des Hautes Etudes en Sciences Sociales 54, Bd Raspail-F-75270 Paris, Cedex 06 France.

(1954), Bandelt and Barthélemy (1984)). Then the question arises of what properties of the medians remain in lattice structures that are less strong than the distributive ones. Here, an answer is given for semimodular lattices. From a theoretical point of view, the semimodularity property is a classical weakening of the distributivity. Concerning applications, the lattices of partitions and of ultrametrics are semimodular and the research of median partitions is now a classical problem, especially important for the clustering of categorical data (see the references given in § 7 and the bibliography by Leclerc and Cucumel (1988)). We obtain the following result (Theorem 5.3): a finite lattice L is (upper) semimodular if and only if the majority rule provides a (lower) bound for the medians of any profile of elements of L . This result, a weakening of Barbut's theorem, is a new characterization of semimodular lattices. It may be useful for the research of medians since it restricts the domain of their possible values. It is a new illustration of the relevance of ordinal theoretic tools in the aggregation problems, with the works quoted above, the recent unifying work on the axiomatic approach by Monjardet (1987), and the general formal approach of Barthélemy and Janowitz (1988).

In § 2 we give some general definitions on finite lattices. In particular, join-irreducible elements are considered. The classes of lattices considered in the paper are briefly described in § 3. Section 4 presents lattice formalizations for some classical consensus methods, including the majority rule. The main result is proved in § 5; in fact, we give a proof of the same type as that outlined above, which allows us to obtain the medians in a linear order. A straightforward characterization of modular lattices is given in § 6, with some examples. As an application, a new and simple proof of the median characterization of distributive lattices is presented in the same section. In the concluding section, we compare our results with some previous works, especially those about central partitions, and we point out some problems.

2. Some definitions on lattices. We consider now the case where X is a finite lattice $(L, \leq, \wedge, \vee)$. Recall that a finite lattice is a finite partially ordered set L such that any pair x, y of elements of L admits a greatest lower bound (g.l.b.) denoted $x \wedge y$ (the *meet* of x and y) and a least upper bound (l.u.b.) denoted $x \vee y$ (the *join* of x and y). Then, any subset Y of L also has a g.l.b. and an l.u.b., respectively, denoted $\wedge Y$ and $\vee Y$. As is usually done, we set $\wedge L = 0$ and $\vee L = 1$.

The *covering relation* $\prec$ associated with L is defined as follows: for any $x, y, z \in L$,

$$x \prec y \text{ if and only if } x \leq y \quad \text{and} \quad x \leq z < y \text{ imply } x = z.$$

The undirected covering graph $G(L)$ is the undirected graph corresponding to this (directed) covering relation. That is, the vertex set of $G(L)$ is L and an unordered pair xy of elements of L is an edge if and only if $x \prec y$ or $y \prec x$. The lattice metric d on L that is naturally induced by the lattice structure of L is the minimum path length metric on $G(L)$: for any $x, y \in L$, $d(x, y)$ is the minimum number of edges of a path of $G(L)$ between x and y . The computation of $d(x, y)$ involves the determination of a geodesic (minimum length) path between x and y . This may be difficult when $|L|$ is a large number.

An element $j \in L$ is said to be *join-irreducible* if and only if $\vee Y = j$ implies $j \in Y$. Since we adopt the usual convention $\vee \emptyset = 0$, this means that zero is not join-irreducible. Equivalently, j is join-irreducible if and only if it covers a unique element of L , denoted by $p(j)$ (the predecessor of j). Let J be the set of all the join-irreducible elements of L and, for any $x \in L$, set $J(x) = \{j \in J : j \leq x\}$. Then $x = \vee J(x)$ is an expression of x as the join of a subset of J .

A subset K of J is a *down-subset* if and only if $j \in K, j' \in J$ and $j' \leq j$ imply $j' \in K$. The following lemma is needed in the rest of the paper.

LEMMA 2.1. *Let K be a down-subset of J . For any $x \in L$ such that $\vee K \not\leq x$, there exists $k \in K$ such that $k \not\leq x$ and $p(k) \leq x$.*

Proof. Since $\vee K \not\leq x$, there exists $k_0 \in K$ such that $k_0 \not\leq x$. If

$$p(k_0) = \vee J(p(k_0)) \not\leq x,$$

choose $k_1 \in J(p(k_0))$ such that $k_1 \not\leq x$. One has $k_1 \in K$, since K is a down-subset of J . Then, see if $p(k_1) \leq x$ holds, and so on. Since L is finite and $0 \leq x$, one must find an element k in K as required. $\square$

More information on lattices, especially on the four classes of lattices briefly described in the next section, can be found in the classical books of Birkhoff (1967) and Grätzer (1978).

3. Four classes of lattices.

3.1. Graded lattices. A finite lattice L is said to be *graded* if and only if there exists an integer-valued *height function* h on L such that, for $x, y \in L$, $x < y$ implies $h(y) = h(x) + 1$. In a graded lattice, we have a simple expression of the distance between any two comparable elements x and y . Specifically, $d(x, y) = |h(x) - h(y)|$, the length of any maximal chain between x and y . Otherwise, the geodesic paths between x and y may remain difficult to recognize when x and y are not comparable.

3.2. Semimodular lattices. Among the many equivalent definitions of the semimodularity property for a finite lattice (some of them extend this notion to the infinite case), we shall use the following three in this paper.

DEFINITION. A finite lattice L is said to be *upper semimodular* if and only if it satisfies one of the three following conditions:

- (i) For every $x, y \in L$, $x \wedge y < x$ and $x \wedge y < y$ imply $x < x \vee y$ and $y < x \vee y$.
- (ii) For every $x, y \in L$, $x \wedge y < x$ implies $y < x \vee y$.
- (iii) L is graded and its lattice metric is given, for every $x, y \in L$, by

$$d(x, y) = 2h(x \vee y) - h(x) - h(y),$$

where h is a height function on L .

These very classical conditions may be found in the book of Birkhoff (1967, and previous editions). The metric characterization (iii) means that (upper) semimodular lattices are exactly those graded lattices where a geodesic path between x and y may be always obtained by the concatenation of a maximal chain between x and $x \vee y$ and a maximal chain between $x \vee y$ and y .

The property of lower semimodularity is dually defined. When lower semimodularity is considered in the sequel, it will be so stated, while we will generally omit “upper” in the case of upper semimodularity.

Now we will point out a property of semimodular lattices: given $x, y \in L$, it is possible to find some minimal moves of x such that $d(x, y)$ also has a minimal variation.

PROPOSITION 3.1. *Let L be a finite semimodular lattice. For any $x, y \in L$ and $j \in J$ such that $j \not\leq x$ and $p(j) \leq x$, the following equalities hold:*

$$\text{If } j \leq x \vee y, \text{ then } d(x \vee j, y) = d(x, y) - 1.$$

$$\text{If } j \not\leq x \vee y, \text{ then } d(x \vee j, y) = d(x, y) + 1.$$

Proof. With the above assumptions, we have $x \wedge j = p(j) < j$. So, by condition (ii), $x < x \vee j$ and $h(x \vee j) = h(x) + 1$. If $j \leq x \vee y$, then $x = (x \vee j) \wedge (x \vee y)$; by (ii) again, $x \vee j \vee y$ covers $x \vee y$ and $h(x \vee j \vee y) = h(x \vee y) + 1$. Otherwise, $x \vee$

$j \vee y = x \vee y$. The proposition follows by direct use of the expression of distances given in condition (iii). $\square$

3.3. Modular lattices. A lattice L is *modular* if and only if it is both upper semimodular and lower semimodular. We just recall here the three expressions of the lattice metric in the case of a finite modular lattice: For every $x, y \in L$,

$$\begin{aligned} d(x, y) &= 2h(x \vee y) - h(x) - h(y) \\ &= h(x) + h(y) - 2h(x \wedge y) \\ &= h(x \vee y) - h(x \wedge y). \end{aligned}$$

3.4. Distributive lattices. A lattice L is *distributive* if and only if it satisfies the distributivity laws: for every $x, y, z \in L$,

$$\begin{aligned} (x \vee y) \wedge z &= (x \wedge z) \vee (y \wedge z), \\ (x \wedge y) \vee z &= (x \vee z) \wedge (y \vee z). \end{aligned}$$

It is well known that a distributive lattice is modular.

Many identities on lattice polynomials may be derived from the distributivity laws, especially concerning the consensus functions defined in § 4.

We end this section with the remark that the assumption of distributivity makes it possible to state Proposition 3.1 differently. If L is distributive, the following implications hold:

$$j \leq x \vee y \Leftrightarrow (x \vee y) \wedge j = (x \wedge j) \vee (y \wedge j) = j,$$

then, since j is a join-irreducible element, we have either $j = x \wedge j$ (that is, $j \leq x$) or $j = y \wedge j$ ($j \leq y$). So, when $j \not\leq x$ is assumed, $j \leq x \vee y$ is equivalent to $j \leq y$.

4. Quota rule polynomials. Given an integer ν , let $P = (x_1, \dots, x_\nu)$ be a ν -tuple of elements of L and $V = \{1, \dots, \nu\}$ be the corresponding set of indices. P will be said to be a ν -*profile*, or simply a *profile* of L .

To construct a consensus function $f : L^\nu \rightarrow L$, it is natural to consider the lattice polynomials c_w , $1 \leq w \leq \nu$, defined as follows: For every $P \in L^\nu$,

$$c_w(P) = \bigvee \{ \bigwedge_{i \in W} x_i : W \subseteq V, |W| \geq w \}.$$

The functions c_w correspond to the so-called w -quota rules, since $c_w(P)$ is the l.u.b. of all the elements of L which are lower bounds of w , or more, of the x_i 's. These functions have at least two properties (U) of unanimity and (I) of isotony which seem desirable for a consensus function:

$$(U) \quad x_1 = x_2 = \dots = x_\nu = x \text{ implies } c_w(P) = x;$$

$$(I) \quad P, P' \in L^\nu \text{ and } P \leq P' \text{ (according to the pointwise order) imply } c_w(P) \leq c_w(P').$$

Note also that $w < w'$ implies $c_w(P) \geq c_{w'}(P)$.

We set $c_w = c$ in the case where $w = [\nu/2]^*$, the least integer strictly superior to $\nu/2$. The consensus function c constitutes the lattice formalization of Condorcet's majority rule.

The consensus functions c'_w corresponding to the dual w -quota rules, with the dual majority rule c' being the special case where $w = [\nu/2]^*$, are dually defined: For every $P \in L^\nu$,

$$c'_w(P) = \bigwedge \{ \bigvee_{i \in W} x_i : W \subseteq V, |W| \geq w \}.$$

For large values of ν , and middle values of w , the computation of the lattice polynomial c_w by the use of the above formula can become very long. The alternative expression given below may provide a more efficient computation in such cases. This expression is also used in some proofs in what follows.

For $P \in L^\nu$ and $1 \leq w \leq \nu$, set $K_w(P) = \{j \in J : |\{i \in V : j \leq x_i\}| \geq w\}$. For $w = \lfloor \nu/2 \rfloor^*$, set $K_w(P) = C(P)$ and, for even ν and $w = \nu/2$, set $K_w(P) = D(P)$. An element of $C(P)$ (respectively, of $D(P) - C(P)$) will be said to be *majority* (respectively, *almost majority*) *join-irreducible*. Then we have the following proposition.

PROPOSITION 4.1. *In a finite lattice L , for any ν -profile P and integer w , $1 \leq w \leq \nu$, the following equality holds:*

$$c_w(P) = \bigvee K_w(P).$$

Proof. We simply write K_w for $K_w(P)$. If $j \in K_w$, then there exists $W \subseteq V$ such that $|W| \geq w$ and $j \leq x_i$ for any $i \in W$. So, $j \leq \bigwedge \{x_i : i \in W\} \leq c_w(P)$, and $\bigvee K_w \leq c_w(P)$.

For the reversed inequality, recall, for $W \subseteq V$, the obvious fact that $J(\bigwedge_{i \in W} x_i) = \bigcap J(x_i)$. This is a subset of K_w , if $|W| \geq w$ is assumed, and we have $\bigwedge x_i \leq \bigvee K_w$. The inequality $c_w(P) \leq \bigvee K_w$ now follows. $\square$

Note that $c(P) = \bigvee C(P)$, and, for even ν , $c_{\nu/2}(P) = \bigvee D(P)$. In the case where L is a distributive lattice, we may verify that $c'_w(P) = c_{\nu-w+1}(P)$ for any ν -profile P and w such that $1 \leq w \leq \nu$. In particular, for odd ν , the majority and dual majority rules are identical:

$$c(P) = c'(P) = \bigvee C(P),$$

while, for even ν , $c'(P) = c_{\nu/2}(P) = \bigvee D(P)$.

Lemma 2.1 and Proposition 4.1 imply the following proposition.

PROPOSITION 4.2. *Let L be a finite lattice, P be a profile of L , and $x \in L$ such that $c_w(P) \leq x$. Then, there exists a join-irreducible $j \in K_w(P)$ such that $j \not\leq x$ and $p(j) \leq x$.*

Proof. Obviously, K_w is a down-subset of J as defined in Lemma 2.1. So the proposition is the direct application of this lemma to the expression of $c_w(P)$ just given above. $\square$

5. A median characterization of semimodularity. Let L be a finite lattice and $P \in L^\nu$. The *remoteness function* r_P (or simply r) associated with P is the integer valued function on L defined, for every $x \in L$, by $r(x) = \sum_{1 \leq i \leq \nu} d(x, x_i)$. As mentioned above, the elements of L with minimal remoteness are the medians.

DEFINITION. A *P -median* (or simply a median) is an element m of L such that $r(m) = \min \{r(x) : x \in L\}$.

The set of all the medians will be denoted by $M(P)$, or simply by M .

PROPOSITION 5.1. *Let L be a finite semimodular lattice. For any profile $P \in L^\nu$, and for any median $m \in M(P)$, the inequality $c(P) \leq m$ holds. Moreover, if ν is even, there is at least one median m such that $c_{\nu/2}(P) \leq m$.*

Proof. The result is a consequence of Propositions 3.1 and 4.2. Let x be an element of L such that $c(P) \not\leq x$. Then, by Proposition 4.2, there exists $j \in C(P)$ such that $j \not\leq x$ and $p(j) \leq x$. Set $y = x \vee j$ and compare $r(x)$ with $r(y)$. By Proposition 3.1, $r(y) = r(x) - \alpha + \beta$, where α (respectively, β) is the number of indices $i \in V$ such that $j \leq x \vee x_i$ (respectively, $j \not\leq x \vee x_i$). Since j is a majority join-irreducible, there are more than $\nu/2$ indices i such that $j \leq x_i$; this implies $\alpha > \beta$, so $r(y) < r(x)$ and x cannot be a median. The inequality $c(P) \leq m$ now follows.

Assume ν is even and m_0 is a median such that $c_{\nu/2}(P) \not\leq m_0$. Let $j \in D(P)$ such that $j \not\leq m_0$ and $p(j) \leq m_0$. Since $m_0 \geq c(P) = \bigvee C(P)$, j is almost majority join-

irreducible. Set $m_1 = m_0 \vee j$. By the same argument as above, $r(m_1) = r(m_0) - \alpha + \beta$, with $\alpha \geq \nu/2$. Indeed $\alpha = \beta = \nu/2$ since m_0 is a median. Thus, m_1 is also a median. This procedure may be repeated until a median m such that $c_{\nu/2}(P) \leq m$ is found. We have also shown there exists a maximal chain of L between m_0 and m all whose elements belong to M . $\square$

Proposition 5.1 implies the Barthélemy (1981) result that in a semimodular lattice the median procedure has the Pareto property. That is, any median m satisfies the inequality

$$\bigwedge_{i \in V} x_i = c_{\nu}(P) \leq m.$$

The first part of Proposition 5.1 has a converse.

PROPOSITION 5.2. *Let L be a finite lattice. If L is not semimodular, there exists a 3-profile $P = (x_1, x_2, x_3)$ and a P -median m such that $c(P) \not\leq m$.*

Proof. Here, we use characterization (i) of the semimodularity property given in § 3.2. If L is not semimodular, we may choose two distinct elements x_1 and x_2 in such a way that both x_1 and x_2 cover $x_1 \wedge x_2$ and $x_1 \vee x_2$ does not cover both x_1 and x_2 . Let x_3 be an element of L such that $x_3 \geq x_1 \vee x_2$ and $d(x_1, x_3) + d(x_2, x_3)$ is minimum under that condition. Then, $c(P) = x_1 \vee x_2$.

We may assume, without loss of generality, that $q = d(x_1, x_3) \leq q' = d(x_2, x_3)$. Then, $q' \geq 2$ and $r(x_3) = q + q' \geq r(x_1) = q + 2$. No element x of L such that $c(P) \leq x$ has a remoteness inferior to that of x_3 . So, there is no median m such that $c(P) \leq m$ except if x_1 is a median and $r(x_3) = r(x_1)$. But in that case, we have $c(P) > x_1$. $\square$

Propositions 5.1 and 5.2 assemble into the following result.

THEOREM 5.3. *A finite lattice L is upper semimodular if and only if, for any profile P and for any P -median m , the inequality $c(P) \leq m$ holds.*

6. The case of modular or distributive lattices. The median characterization of the modularity property for a finite lattice is straightforward, using directly Theorem 5.3 and its dual.

COROLLARY 6.1. *A finite lattice L is modular if and only if, for any profile $P \in L^{\nu}$ and for any P -median m , the inequalities $c(P) \leq m \leq c'(P)$ hold.*

Notice the inequalities $c(P) \leq m \leq c'(P)$ for any 3-profile are sufficient to imply that a lattice is modular.

Whenever $c(P) = c'(P)$, the P -median is unique and given by the majority rule. This is the case if ν is odd and the elements of P generate a distributive sublattice L' of L .

We present four illustrations of the results on medians in semimodular lattices. In these examples, L is the modular lattice with 13 elements given by Fig. 1.

Example 1. $\nu = 3$, $P = (x_1, x_2, x_3)$. Then, $c(P) = 0$, $c'(P) = y_1$, and $M = \{0, y_1\}$.

Example 2. $\nu = 5$, $P = (y_1, y_2, y_3, y_4, y_5)$. Then, $c(P) = c'(P) = y_1$, the unique P -median. Notice the elements of P do not generate a distributive sublattice of L .

Example 3. $\nu = 4$, $P = (x_1, z_1, y_4, y_5)$. Then, $c(P) = 0$ and $c'(P) = 1$, while $c_2(P) = c'_2(P) = y_1$.

Neither $c(P)$ nor $c'(P)$ are medians: the unique median is y_1 . It satisfies the second part of Proposition 5.1 and the dual result.

Example 4. $\nu = 4$, $P = (x_2, x_3, y_2, y_3)$. Then, $c(P) = 0$, $c'(P) = c_2(P) = c'_2(P) = y_1$. We find $M = \{0, x_2, x_3, y_1\}$. Notice $\{0, x_2, y_1\}$ and $\{0, x_3, y_1\}$ are two chains of medians as described in the proof of the second part of Proposition 5.1.

Now we consider the case of a distributive (then modular) lattice L . If ν is odd, distributivity implies that $c(P) = c'(P)$ is the unique median; if ν is even, the following

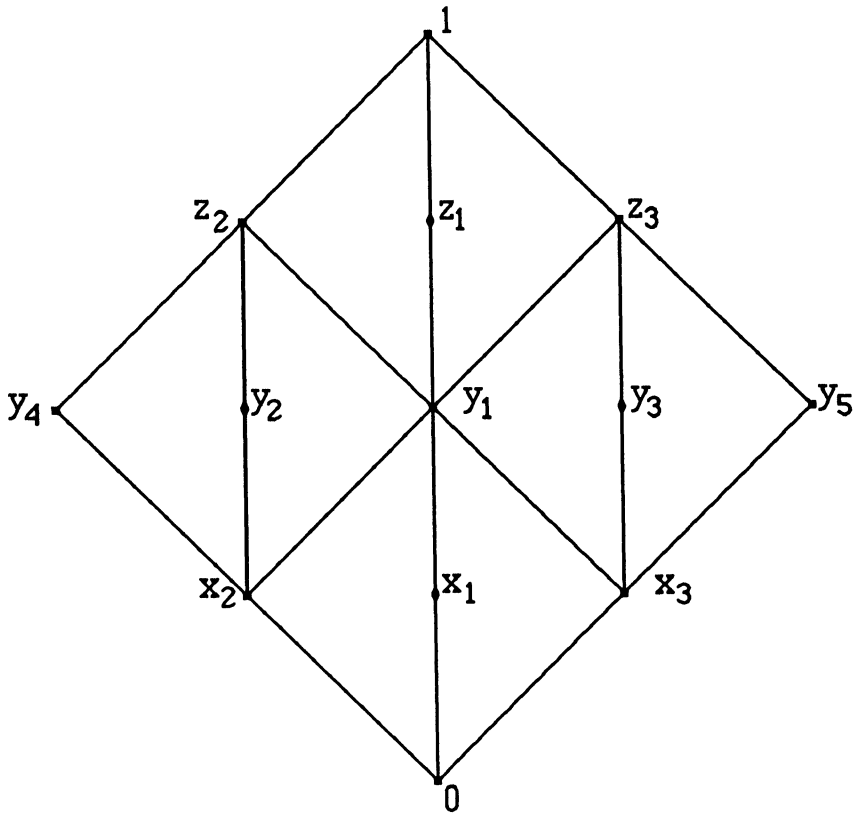


FIG. 1

result implies that the remoteness function r_P is constant on the interval $[c(P), c'(P)]$, which is the set M of all the medians.

PROPOSITION 6.2. *Let ν be an even integer and $P \in L^\nu$ a ν -profile of a finite distributive lattice L . For any $x, y \in L$ such that $c(P) \leq x < y \leq c'(P)$, we have $r(x) = r(y)$.*

Proof. By Lemma 2.1, with $D(P) \cap J(y)$ as a down-subset of J , we know there exists a join-irreducible $j \in D(P)$ such that $j \not\leq x, j \leq y$ and $p(j) \leq x$. Indeed, since $x \leq \vee C(P)$, j is almost majority, that is, there are exactly $\nu/2$ of the x_i 's such that $j \leq x_i$. We have $y = x \vee j$, and, for $i \in V$, by Proposition 4.1 and the remark in § 3.4, the equality $d(y, x_i) = d(x, x_i) - 1$ (respectively, $d(x, x_i) + 1$) if $j \leq x_i$ (respectively, $j \not\leq x_i$). The equality $r(x) = r(y)$ follows. $\square$

As a converse, we obtain the following proposition.

PROPOSITION 6.3. *Let L be a finite, not distributive, lattice. Then there exists a 3-profile $P = (x_1, x_2, x_3)$ such that the set M of all the P -medians differs from the interval $[c(P), c'(P)]$.*

Proof. If L is not upper (lower) semimodular, Proposition 5.2 (or its dual) implies that there is a 3-profile with a median on the outside of the interval. Assume L is modular but not distributive. Then, it is well known that it has three distinct elements x_1, x_2 , and x_3 such that $x_1 \wedge x_2 = x_1 \wedge x_3 = x_2 \wedge x_3 = y$ and $x_1 \vee x_2 = x_1 \vee x_3 = x_2 \vee x_3 = y'$. They generate the sublattice of the so-called M_3 type represented in Fig. 2.

Let h be a height function on L , and $h_i = h(x_i) - h(y)$, for $i = 1, 2, 3$. We have, with $P = (x_1, x_2, x_3)$, $c(P) = y$, $c'(P) = y'$, $r(y) = h_1 + h_2 + h_3$ and, using one of the

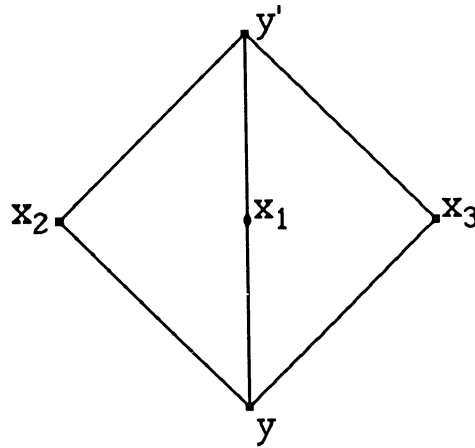


FIG. 2

expressions of the lattice metric in a modular lattice given in § 3.3, $r(x_1) = 2h_1 + h_2 + h_3$. So, x_1 is an element of the interval $[c(P), c'(P)]$ which is not a median. $\square$

In the following median characterizations of distributive lattices, the implication of (iv) by (i) is an early result of Birkhoff and Kiss (1947), extended to $[(i) \Rightarrow (iii)]$ by Barbut (1961) and to $[(i) \Rightarrow (ii)]$ by Monjardet (1980). The above results make it possible to give a new simple proof of the converse implications. Previously, they were derived from a strong result of Bandelt and Barthélemy (1984): for a finite lattice, the distributivity property is characterized by the uniqueness of the median of any odd profile. The proof of this fact, based upon an Avann (1961) result, requires many more theoretical arguments than those used in this paper.

THEOREM 6.4. *Let L be a finite lattice. The four following conditions are equivalent:*

- (i) *L is distributive.*
- (ii) *For any profile $P \in L^v$, the set $M(P)$ of all the P -medians is the interval $[c(P), c'(P)]$.*
- (iii) *For any v -profile P with odd v , $c(P)$ is the unique P -median.*
- (iv) *For any 3-profile P , $c(P)$ is the unique P -median.*

Proof. By Propositions 6.2 and 6.3, conditions (i) and (ii) are equivalent. They imply (iii), while (iii) $\Rightarrow$ (iv) is obvious. If (iv) is true, by Corollary 6.1, L is at least modular. But it cannot admit the M_3 sublattice of Fig. 2: setting $h'_i = h(y') - h(x_i)$ for $i = 1, 2, 3$, and using the metric properties of modular lattices given in § 3.3, we would have in such a sublattice

$$d(x_1, x_2) + d(x_1, x_3) + d(x_2, x_3) = 2h'_1 + 2h'_2 + 2h'_3 = 2h_1 + 2h_2 + 2h_3.$$

Then, $r(y) = h_1 + h_2 + h_3 = r(y') = h'_1 + h'_2 + h'_3$ and $y = c(P)$ would not be the unique median. So, if (iv) is true, L is a distributive lattice. $\square$

7. Some remarks and open questions. The results of § 5 give more information on medians in semimodular lattices. Especially, they show the median procedure has a good property as an aggregation method: it respects the majority consensus, in the sense that, whenever an element j is majority join-irreducible, we have $j \leq m$ for every median m . This is the concrete significance of Proposition 5.1. The other general properties of the median procedure in semimodular lattices remain to be studied. Barthélemy and Janowitz (1988) have recently given axiomatic characterizations of the median procedure (and of

some other related consensus methods) in distributive lattices. A question is to generalize their results to semimodular lattices.

It would be interesting to have an upper bound for the medians in an upper semimodular lattice. The dual unanimity consensus $c_1(P) = c'_v(P) = \bigvee_i x_i$ might be such a bound, but we have neither succeeded to show that property nor found a counter-example. In order to restrict the domain where the medians may be searched, it may be noted, from simple considerations based upon the expression of the metric d , that, if m is a P -median, $h(m) \geq h(c_1(P))$ imply $m = c_1(P)$.

Another question concerns the structure of the set M of all the medians. It was recalled in the previous section that, for any profile P in a distributive lattice L , M is an interval of L . By a simple and elegant argument, Barthélemy (1981) has shown M to be a sublattice of L whenever L is modular. We are then tempted to suppose that, if L is semimodular, then M is always a sub-semilattice of L . For $n \geq 3$, the family of lattices L_n of Fig. 3 disproves this conjecture.

L_n has $3n + 1$ elements $0, x_1, \dots, x_n, y_1, \dots, y_n, z_1, \dots, z_n = 1$. For $i = 2, \dots, n$, $x_i, y_i, z_{i-1} < z_i$; $x_{i-1} < x_i$; $y_{i-1} < y_i$; $x_1, y_1 < z_1$; $0 < x_1, y_1$; and there are no other covering pairs in L . Then, for the 3-profile $P = \{0, x_n, y_n\}$, we have $M(P) = \{x_n, y_n\}$, an antichain of L . Moreover, in the above example, $c(P) = 0$, and $d(0, x_n) = n$, while the total height of the lattice L_n is $n + 2$. So a median and the majority rule element may be very far from each other.

Indeed, the determination of the medians in a semimodular lattice does not seem to be an easy question. Its computational complexity is an open problem, with the especially interesting instance of the lattice of partitions, where the lattice metric d was first considered by Boorman and Arabie (1972). In that case, there is an efficient computation (in $O(n)$ time) of $d(x, y)$ for any partitions x and y of the same set E with n elements (Day 1981).

It is worth noticing that lattices of partitions are not only semimodular: they are geometric (or matroid) lattices, while it may be verified that lattices of ultrametrics are examples of generalized matroid lattices, as defined by Stern (1978). Then, the question arises of the possible implications concerning the medians of these stronger structural properties.

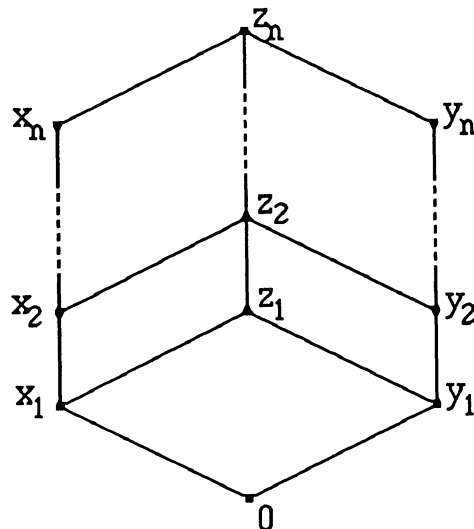


FIG. 3

As was noted in the introduction, the literature about median partitions is important (Régnier (1965), Marcotorchino and Michaud (1982), Schader (1981), Schader and Tüshaus (1986), Wakabayashi (1986), Grötschel and Wakabayashi (1987), among others). These authors do not use the lattice metric d , but the metric δ defined as follows: for any partitions x and y of the same finite set E , $\delta(x, y) = |R_x \Delta R_y|$, the cardinality of the symmetric difference between the equivalence relations R_x and R_y on E , respectively, associated with x and y . Wakabayashi has shown the problem of finding median partitions using metric δ to be NP-complete.

Let us consider, as an illustration, the profile $P = (x_1, x_2, x_3, x_4, x_5)$ of five partitions of the set $E = \{a, b, c, d, e, f\}$; $x_1 = /ad/b/ce/f/$, $x_2 = /abcd/e/f/$, $x_3 = /abf/c/d/e/$, $x_4 = /abc/def/$, $x_5 = /abd/ce/f/$. In this example, the majority rule partition is $c = /abd/c/e/f/$, whose remoteness (with the lattice metric) is equal to 10, while the partition $m = /abd/ce/f/$, which is the unique median, has a remoteness equal to 9. With the δ metric, there are two medians c and $x = /ab/c/d/e/f/$.

Indeed, the metric δ may be extended to any finite lattice L , by setting, for any $x, y \in L$, $\delta(x, y) = |J(x) \Delta J(y)|$. δ is the restriction to the set $\{J(x) : x \in L\}$ of the lattice metric on the Boolean (distributive) lattice $(\mathcal{P}(J), \cup, \cap)$.

The uses of the metrics d and δ appear to be almost incompatible in the study of medians. Assume ν is odd, and let $P = (x_1, \dots, x_\nu)$ be a ν -profile of L . The median in the lattice $\mathcal{P}(J)$ of the corresponding profile $K = (J(x_1), \dots, J(x_\nu))$ is given by the majority rule. It is nothing but the set $C(P)$ of the majority join-irreducibles. But generally, there is no element x of L such that $J(x) = C(P)$. Proposition 5.1 simply ensures $C(P) \subseteq J(c(P))$. Define the δ -remoteness function ρ by $\rho(x) = \sum_i \delta(x, x_i)$ for any $x \in L$, and consider $y \in L$ such that $y > c(P)$. From $C(P) \subseteq J(c(P)) \subset J(y)$, we may show the inequality $\rho(y) > \rho(c(P))$ and y cannot be a median for the δ metric. The conclusion is that only $c(P)$, the majority rule element, may eventually be a median for both metrics d and δ . The situation is analogous, but more complicated in the case of even ν .

Acknowledgments. The author thanks J. P. Barthélemy and B. Monjardet for helpful discussions and advice during the preparation of this work, and the anonymous referees for many corrections and relevant suggestions.

REFERENCES

- S. P. AVANN (1961), *Metric ternary distributive semi-lattices*, Proc. Amer. Math. Soc., 12, pp. 407–414.
- H. J. BANDELT (1984), *Discrete ordered sets whose covering graphs are median*, Proc. Amer. Math. Soc., 9, pp. 6–8.
- H. J. BANDELT AND J. P. BARTHELEMY (1984), *Medians in median graphs*, Discrete Applied Math., 8, pp. 131–142.
- M. BARBUT (1961), *Médiane, distributivité, éloignements*, reprint, Math. Sci. Humaines, 70 (1980), pp. 5–31.
- J. P. BARTHELEMY (1981), *Trois propriétés des médianes dans un treillis modulaire*, Math. Sci. Humaines, 75, pp. 83–91.
- J. P. BARTHELEMY AND M. F. JANOWITZ (1988), *A formal theory of consensus*, SIAM J. Discrete Math., submitted.
- J. P. BARTHELEMY, B. LECLERC, AND B. MONJARDET (1986), *On the use of ordered sets in problems of comparison and consensus of classifications*, J. Classification, 3, pp. 187–224.
- J. P. BARTHELEMY AND B. MONJARDET (1981), *The median procedure in cluster analysis and social choice theory*, Math. Social Sci., 1, pp. 235–268.
- (1988), *The median procedure in data analysis: new results and open problems*, in Classification and Related Methods of Data Analysis, H. H. Bock, ed., North-Holland, Amsterdam, pp. 309–316.
- G. BIRKHOFF (1967), *Lattice Theory*, Third edition, American Mathematical Society, Providence, RI.
- G. BIRKHOFF AND S. A. KISS (1947), *A ternary operation in distributive lattices*, Bull. Amer. Math. Soc., 53, pp. 749–752.

- S. A. BOORMAN AND P. ARABIE (1972), *Structural measures and the method of sorting*, in *Multidimensional Scaling*, Vol. 1, Theory and Applications in the Behavioural Sciences, R. N. Shepard, A. K. Romney, and S. B. Nerlove, eds., Seminar Press, New York, pp. 226–249.
- W. H. E. DAY (1981), *The complexity of computing metric distances between partitions*, *Math. Social Sci.*, 1, pp. 269–287.
- G. GRÄTZER (1978), *General Lattice Theory*, Birkhäuser Verlag, Basel.
- M. GRÖTSCHEL AND Y. WAKABAYASHI (1987), *A cutting plane algorithm for a clustering problem*, Research Report no. 9, Schwerpunktprogramm der Deutschen Forschungsgemeinschaft, Institut für Mathematik, Universität Augsburg.
- C. JORDAN (1869), *Sur les assemblages de lignes*, *J. Reine Angew. Math.*, 70, pp. 185–190.
- B. LECLERC AND G. CUCUMEL (1987), *Consensus en classification: une revue bibliographique*, *Math. Sci. Humaines*, 100, pp. 109–128.
- F. MARCOTORCHINO AND P. MICHAUD (1982), *Agrégation de similarités en classification automatique*, *Rev. Statist. Appl.*, 30, pp. 21–44.
- B. MONJARDET (1980), *Théorie et application de la médiane dans les treillis distributifs finis*, *Ann. Discrete Math.*, 9, pp. 87–91.
- (1981), *Metrics on partially ordered sets, a survey*, *Discrete Math.*, 35, pp. 173–184.
- (1987), *Arrowian characterization of latticial federation consensus functions*, Research Report CMS-P030, Centre d'Analyse et de Mathématique Sociales, Ecole des Hautes Etudes en Sciences Sociales, (CAMS-EHESS), Paris, to appear.
- S. REGNIER (1965), *Sur quelques aspects mathématiques des problèmes de classification automatique*, *I.C.C. Bulletin* 4, pp. 175–191; reprint, *Math. Sci. Humaines*, 82 (1983), pp. 13–29.
- M. SCHADER (1981), *Scharfe und unscharfe klassifikation qualitative daten*, Athenäum, Königstein.
- M. SCHADER AND U. TUSHAUS (1986), *Subgradient Methods for Analyzing Qualitative Data, Classification as a Tool of Research*, W. Gaul and M. Schader, eds., North Holland, Amsterdam, pp. 397–403.
- M. SHOLANDER (1954), *Medians, lattices and trees*, *Proc. Amer. Math. Soc.*, 5, pp. 808–812.
- P. J. SLATER (1978), *Center to centroid in graphs*, *J. Graph Theory*, 2, pp. 209–222.
- M. STERN (1981), *Generalized matroid lattices, algebraic methods in graph theory*, (Proc. Conf. Szeged 1978), *Colloq. Math. Soc. Janos Bolyai*, Vol. 25, North-Holland, Amsterdam, pp. 727–748.
- Y. WAKABAYASHI (1986), *Aggregation of binary relations: algorithmic and polyhedral investigations*, Ph.D. Thesis, Universität Augsburg.
- B. ZELINKA (1968), *Medians and peripherians of trees*, *Arch. Math.*, (Brno) 4, pp. 87–95.

LOCALLY SYMMETRIC GRAPHS OF PRESCRIBED GIRTH AND COXETER GROUPS*

ALEXANDER LUBOTZKY†

Abstract. Kupitz and Perles showed that for $g = 3$ or 4 and fixed $k \geq 3$, every connected, k -regular locally symmetric graph is finite and there are only finitely many such graphs. It is shown that the situation is completely different for every even $g \geq 6$. The proof uses Coxeter groups and some suitable quotients.

Key words. locally symmetric graphs, girth, Coxeter groups

AMS(MOS) subject classifications. 05C25, 05C38

Introduction. Let $G = (V, E)$ be a graph. A vertex $x \in V$ is called *locally symmetric* if every permutation of the neighbours of x can be extended to an automorphism of G which fixes x . The graph G is called *locally symmetric* if every vertex is locally symmetric. The *girth* of G is the length of the shortest closed path in G . It is not difficult to see that a connected k -regular graph of girth three must be the complete graph on $k + 1$ vertices. Kupitz and Perles analyzed the structure of locally symmetric graphs of girth four and concluded with the following theorem.

THEOREM (Kupitz–Perles). *Let G be a connected locally symmetric k -regular graph ($k \geq 3$) of girth $g = 4$.*

Then:

1) G is finite.

2) G is one of the following:

(a) $K_{k,k}$ the complete bi-partite graph with k points on each side.

(b) $K_{k+1,k+1} - (k+1)K_{1,1}$, i.e., $K_{k+1,k+1}$ minus a perfect matching.

(c) C^k , the graph of the k -cube.

(d) $C^k/2$ the graph of the k -cube with opposite points identified.

If $k = 4$, then (d) = (a), but there is another graph:

(d') The nonadjacency graph of points versus lines of the projective plane over Z_2 .

If $k = 3$, these graphs are not all necessarily distinct, and type (d) has girth 3 (in fact it is equal to K_4).

For $g = 5$, they know only finitely many examples, two with $k = 3$ and two with $k = 4$, while for $g = 6$ they knew infinitely many examples with $k = 3$. They naturally raised the question for general g and k . Here we show the following theorem.

THEOREM. *Let $g = 2m$, $m \geq 3$ and $k \geq 3$ fixed. Then there is an infinite connected k -regular locally symmetric graph of girth g , and there are infinitely many such finite graphs.*

The proof of the theorem is based on the following observation: Let G be a group generated by a set of generators $S = \{s_1, s_2, \dots, s_k\}$ each one of order two and with a “symmetric presentation” (i.e., if $w(s_1, \dots, s_k)$ is a relation so is $w(s_{\sigma(1)}, \dots, s_{\sigma(k)})$ for every permutation σ of $\{1, \dots, k\}$). Then the Cayley graph of G with respect to S is a locally symmetric graph.

This observation suggests considering Coxeter groups with defining relations $(s_i s_j)^{m_{ij}} = 1$ where all $m_{ij} = m$ (when $i \neq j$). The group W so obtained is of girth $2m$. It is a finite group if $m = 2$ (which is compatible with Kupitz–Perles’ Theorem), but infinite

* Received by the editors June 7, 1988; accepted for publication (in revised form) July 14, 1989.

† Department of Mathematics, Yale University, Box 2155 Yale Station, New Haven, Connecticut 06520.
Permanent address, Institute of Mathematics, Hebrew University, Jerusalem, 91904, Israel.

and residually finite if $m \geq 3$. So the Cayley graph of W gives the infinite model and suitable finite quotients of it provide infinitely many examples.

The proof is an easy application of standard results on Coxeter groups which are summarized in § 1. Some other combinatorial applications of the residual-finiteness of Coxeter groups can be found in [V1] and [V2]. Groups with symmetric presentations were studied recently by Campbell and Robertson.

1. Coxeter groups. In this section we review some facts about Coxeter groups needed later. Proofs can be found in [B].

DEFINITIONS. Let S be a set of k elements.

(1) A Coxeter matrix is a map $M: S \times S \rightarrow N \cup \{\infty\}$ such that

(i) $M(s, t) = M(t, s)$ for every $s, t \in S$.

(ii) $M(s, s) = 1$ for every $s \in S$.

(iii) $M(s, t) \geq 2$ for $s \neq t \in S$.

(2) The Coxeter group $W = W(M)$ associated with the Coxeter matrix M is the group given by the set of generators S and the relations $(st)^{M(s,t)} = 1$ if $M(s, t) \neq \infty$. (In particular $s^2 = 1$.)

(3) The bilinear form $B = B(M)$ associated with M is the form on the k -dimensional real vector space R^k , with basis $\{e_s | s \in S\}$ given by $B(e_s, e_t) = -\cos(\pi/M(s, t))$.

(4) An element $w \in W$ is said to have length τ , denoted $l(w) = \tau$, if $w = s_1 \cdots s_\tau$, $s_i \in S$, and τ is the smallest integer with this property. Note that $l(w)$ is just the distance of w from the identity element e on the Cayley graph $X(W, S)$.

The following property is very useful.

PROPOSITION 1.1 (The Exchange Lemma [B, Chap. IV, § 1.4, Lemma 3]). *Let $w \in W$ and $s \in S$ such that $l(sw) < l(w) = q$. Then if $w = s_1 \cdots s_q$ then there exists j , $1 \leq j \leq q$ such that $ss_1 \cdots s_{j-1} = s_1 \cdots s_{j-1}s_j$.*

PROPOSITION 1.2 [B, Chap. V, § 4.3, Prop. 4]. *For every $s, t \in S$, the subgroup of W generated by s and t is the dihedral group of order $2M(s, t)$.*

THEOREM 1.3 [B, Chap. V, § 4.8, Thm. 2]. *The group W is finite if and only if B is a positive definite form.*

THEOREM 1.4 [B, Chap. V, § 4.4, Cor. 2]. *The group W has a faithful linear representation into $GL_k(R)$.*

DEFINITION. A group is *residually-finite* if the intersection of its finite index normal subgroups is trivial.

Finitely generated linear groups are residually finite [W, § 4]. Theorem 1.4, therefore, implies the following proposition.

PROPOSITION 1.5. *W is residually finite.*

2. Proof of the theorem. Let $g = 2m$ and $W = W_{g,k}$ the Coxeter group defined by a set S of k generators and the Coxeter matrix

$$(2.1) \quad \begin{aligned} M(s, t) &= m \quad \text{for } s \neq t \in S \\ M(s, s) &= 1 \quad \text{for every } s \in S. \end{aligned}$$

A sequence $u = (s_1, \cdots, s_q)$, $s_i \in S$, is called *admissible* if $s_i \neq s_{i+1}$ for every $i = 1, \cdots, q-1$. Admissible sequences are corresponded naturally to paths without back-tracking on the Cayley graph $X(W, S)$ starting at e . We denote by $[u]$ the product in W , i.e., $[u] = s_1 \cdots s_q$.

LEMMA 2.1. *If $(s_1, \cdots, s_q)$ is an admissible sequence and $q < g = 2m$, then $s_1 \cdots s_q \neq e$.*

Proof. Let q be the minimal positive integer for which there is an admissible sequence $u = (s_1, \cdots, s_q)$ with $s_1 \cdots s_q = e$. Since all relations in (2.1) contain every generator

$s \in S$ an even number of times, this is also true in u . In particular q must be even, say $q = 2r$. We will first show that for every $1 \leq i \leq q - 2$, $s_i = s_{i+2}$. To see this, observe first that every cyclic permutation of u is also admissible. Otherwise, $s_1 = s_q$ which implies that $s_2 \cdots s_{q-1} = e$ in contradiction to the minimality of q . By a similar reasoning, for every cyclic permutation u' of u , $[u'] = e$. We can therefore assume that $i = r - 1$ and $i + 2 = r + 1$.

Now let $w = s_{r+1} \cdots s_q$. The minimality of q shows that $l(w) = r$ while $l(s_r \cdot s_{r+1} \cdots s_q) = r - 1$. We can apply Proposition 1.1 to deduce that there is l , $r + 1 \leq l \leq q$ such that $s_r s_{r+1} \cdots s_{l-1} = s_{r+1} \cdots s_l$. Thus,

$$s_l \cdot s_{l-1} \cdots s_{r+1} s_r s_{r+1} \cdots s_{l-1} = e.$$

The sequence $(s_l, s_{l-1}, \dots, s_{r+1}, s_r, s_{r+1}, \dots, s_{l-1})$ is admissible, so the minimality of q implies $l = q$. But then:

$$e = s_1 \cdots s_{r-1} s_r s_r s_{r+1} \cdots s_{q-1} = s_1 \cdots s_{r-1} s_{r+1} \cdots s_{q-1}$$

which is again a contradiction to the minimality of q unless $s_{r-1} = s_{r+1}$, which is what we wanted to prove.

This shows that u is of type $(s, t, s, t, \dots, s, t)$. The group generated by s and t is the dihedral group of order $2m$ (Proposition 1.2), so we can deduce that $q = 2m = g$. $\square$

Remark. Lemma 2.1 could be proven also by an application of the method of diagrams of relations (cf. [R]).

COROLLARY 2.2. *The girth of $X(W_{g,k}, S)$ is g .*

Proof. For Cayley graphs, the girth is determined only by the closed paths passing through e . So Lemma 2.1 gives the girth. $\square$

Let R be the subset of W consisting of all $w \in W$ with $l(w) \leq g$. Let A_1 be the family of normal subgroups $N \leq W$ such that $N \cap R = \{e\}$.

COROLLARY 2.3. *If $N \in A_1$, then the girth of $X(W/N, S)$ is g . (By S we mean, of course, the image of S in W/N).*

Proof. The shortest possible “admissible” relation in W/N is $(st)^m = 1$ for $s, t \in S$. $\square$

To get locally symmetric graphs we argue as follows: The symmetric group $\text{Sym}(k)$ acts on the set $S = \{s_1, \dots, s_k\}$ in the obvious way. Every such $P \in \text{Sym}(k)$ preserves the relations (2.1) and thus defines an automorphism of the group W , and together an action of $\text{Sym}(k)$ on W . Denote by A_2 the set of normal subgroups of W which are invariant under the action of $\text{Sym}(k)$.

PROPOSITION 2.4. *If $I \in A_2$, then the graph $X = X(W/I, S)$ is locally symmetric.*

Proof. Cayley graphs are homogeneous, so it suffices to check the condition at one point, say at e . The permutation $P \in \text{Sym}(k)$ also induces an automorphism $\tilde{P}$ of the Cayley graph X . (Warning: not every automorphism of a group induces an automorphism of a Cayley graph; it will do so if it preserves, as in our case, the set of generators). This automorphism $\tilde{P}$ fixes the identity and acts on its neighbours just like P . $\square$

The Coxeter matrix M defined by (2.1) gives rise to the bilinear form

$$B(X_1, \dots, X_k) = \sum_{i=1}^k X_i^2 - 2 \cos\left(\frac{\pi}{m}\right) \sum_{1 \leq i < j \leq k} X_i X_j.$$

For $m = 2$, i.e., $g = 4$, this is a positive definite form and by Theorem 1.3, $W_{4,k}$ is finite for every k (in fact equal to Z_2^k). On the other hand, for $m \geq 3$,

$$B(1, 1, \dots, 1) < 0$$

so $W_{g,k}$ is infinite.

We conclude that for $g \geq 6$, $X(W_{g,k}, S)$ is an infinite, connected, k -regular, locally symmetric graph of girth g . To find finite examples, we continue as follows.

Let A_3 be the family of all finite index normal subgroups of W , and

$$A = A_1 \cap A_2 \cap A_3.$$

LEMMA 2.5. *If $g \geq 6$, then A is an infinite family of subgroups of W of unbounded indices.*

Proof. The second part follows from the first, since W is finitely generated and has, therefore, only finitely many subgroups of each given index. To see the first part, observe that if $N \in A_3$ then

$$I(N) = \cap \{P(N) \mid P \in \text{Sym}(k)\}$$

is in $A_3 \cap A_2$ and $I(N) \subseteq N$. Also if $N \in A_3 \cap A_1$ then $I(N) \in A$. By (1.5) and the fact that R is finite we know that $A_3 \cap A_1$ is infinite and so is A . $\square$

Finally, if $I \in A$ then $X(W/I, S)$ is connected (since S generates W/I), k -regular (since $|S| = k$), locally symmetric (by 2.4), finite (since $A \subseteq A_3$) and of girth g (by 2.3). This is an infinite family of examples of sizes going to infinity which proves the theorem.

Acknowledgments. The author thanks M. Perles and Y. Kupitz for bringing the problem to his attention and thanks G. Kalai and W. M. Kantor for some suggestions for an improved presentation.

REFERENCES

- [B] N. BOURBAKI, *Groupes et Algebres de Lie*, Herman, Paris, 1968, Chapitres 4, 5, et 6.
- [KP] Y. KUPITZ AND M. PERLES, *Classification of locally symmetric regular graphs of girth four*, in preparation.
- [R] E. RIPS, *Oral communication*.
- [V1] A. VINCE, *Combinatorial maps*, J. Combin. Theory Ser. B, 34 (1983), pp. 1–21.
- [V2] ———, *Regular combinatorial maps*, J. Combin. Theory Ser. B, 35 (1983), pp. 256–277.
- [W] B. WEHRFRITZ, *Infinite Linear Groups*, Springer-Verlag, Berlin, 1973.

ON A LOWER BOUND FOR SHORT NONCONTRACTIBLE CYCLES IN EMBEDDED GRAPHS*

T. PRZYTYCKA† AND J. H. PRZYTYCKI‡

Abstract. In this paper, a technique is developed that allows the construction of a triangulation of a closed orientable surface of genus g by an n -vertex graph in such a way that the triangulation does not have short noncontractible cycles. Using this technique, a counterexample is constructed to a conjecture by Hutchinson that the length of the shortest noncontractible cycle in any such triangulation is $O(\sqrt{n/g})$. The presented technique can also be used to show that the function $\sqrt{n/g} \log^* g$ provides a lower bound for the shortest noncontractible cycle in a triangulation of a surface of genus g .

Key words. embedded graph, covering spaces, orientable surface, noncontractible cycle

AMS(MOS) subject classifications. 05C10, 05C38, 57M10

1. Introduction. An important problem in topological graph theory is how to find short noncontractible cycles in an embedded graph. This implies a natural question: What is the lower bound for the length of the shortest noncontractible cycle in a triangulation of an orientable surface? Usually questions about lower bounds are difficult to answer and there is a need to find general methods that can be used to obtain nontrivial lower bounds. This paper develops a technique which allows us to construct a triangulation of a closed orientable surface of genus g by an n -vertex graph which does not have short noncontractible cycles.

Consider the family $T_{g,n}$ of all triangulations of a surface of genus g by an n -vertex graph. Denote by $C(g, n)$ the maximum length of a shortest noncontractible cycle over all elements $T \in T_{g,n}$. Hutchinson [2] has proved that $C(g, n) = O(\sqrt{n/g} \log g)$.¹ This means that in any triangulation of a surface of genus g with a graph of n vertices there exists a noncontractible cycle of length at most $c\sqrt{n/g} \log g$ where c is some constant. In the same paper, Hutchinson conjectures that $C(g, n) = O(\sqrt{n/g})$ and that for $n < g$, $C(g, n) = O(1)$. Using our technique we have been able to construct a counterexample to this conjecture. The technique is derived from algebraic topology and uses covering spaces. In the conclusions we outline an extension of our technique to show that $C(g, n) = \Omega(\sqrt{n/g} \log^* g)$ where

$$\log^* n = k \quad \text{iff} \quad \underbrace{\log \log \cdots \log n}_{k \text{ times}} \leq 1 < \underbrace{\log \log \cdots \log n}_{k-1 \text{ times}}.$$

It means that $C(n, g)$ grows at least as fast as $\sqrt{n/g} \log^* g$. We omit the detailed proof of this fact for two reasons:

- 1) It involves technical details that are not essential for the technique that we present.
- 2) We conjecture that the result obtained by Hutchinson is tight up to a constant (i.e., we conjecture that $C(g, n) = \Theta(\sqrt{n/g} \log g)$).

* Received by the editors January 21, 1988; accepted for publication (in revised form) August 15, 1989.

† Department of Computer Science, University of British Columbia, Vancouver, British Columbia. This research was supported by a University of British Columbia graduate Fellowship. The author is on leave from Department of Mathematics, Warsaw University, Warsaw, Poland.

‡ Department of Mathematics, University of British Columbia, Vancouver, British Columbia. This author is on leave from Department of Mathematics, Computer Science, and Mechanics, Warsaw University, Warsaw, Poland.

¹ In the paper we denote $\log_2 n$ by $\log n$.

The method we use to solve the problem is novel for showing lower bounds in graph theory, is interesting in its own right, and can potentially be applied to other related problems.

In the next section we present basic facts and definitions used in the construction. For proofs and more formal presentation see, for example, [3] or [4]. In the third section we prove topological theorems that suggest a method for constructing a triangulation without short noncontractible cycles. In the fourth section we present the construction. In the Appendix, we prove several algebraic theorems used in the paper.

2. Basic facts and definitions. Throughout the paper we use *surface* to describe a closed, connected, orientable surface without boundary. Informally, this describes a sphere with g handles where g is the genus of the surface. We denote a double torus (a sphere with two handles) by F_2 .

A graph is said to be *embedded* on an orientable surface of genus g ($g \geq 0$) if the graph can be drawn on the surface such that no two edges cross. If the graph G is embedded in a surface S , the complement of G relative to S is a collection of open sets called *faces*. If all of the faces are open discs, we say that the embedding is a *2-cell embedding*. Throughout the rest of the paper we will consider only a 2-cell embedding.

An embedding is called a *triangulation* if every face is bounded by three edges.

We also use some facts and definitions from algebraic topology such as the following.

DEFINITION 2.1. Let X' and X be two surfaces. A continuous mapping $p: X' \rightarrow X$ is said to be a *covering map* if each point $x \in X$ has an open neighborhood U_x such that $p^{-1}(U_x)$ is the disjoint sum of open subsets of X' each of which is mapped homeomorphically onto U_x by p . The surface X' is called *the covering surface*; the surface X is called *the basic surface*.

DEFINITION 2.2. The covering $p: X' \rightarrow X$ is called a *k-fold covering* if for any $x \in X$ $|p^{-1}(x)| = k$.

It is sometimes convenient to imagine that in a k -fold covering for each U_x from Definition 2.1 there exist k copies of U_x in the covering space X' , each of them mapped onto U_x by p .

Having chosen a point $x \in X$ we can consider the set of all closed paths from x to x . Point x is called a *base point* for those paths.

DEFINITION 2.3. Two closed paths with a base point x are *equivalent relative to the base point x* if they are homotopic relative to the base point x .

Informally two paths are equivalent relative to the base point x if one can be continuously transformed to another in such a way that endpoints are not moved.

THEOREM 2.4. *The set of equivalence classes of closed paths based at $x \in X$ forms a group.*

The group formed by the set of equivalence classes of closed paths based at $x \in X$ is denoted by $\pi(X, x)$ and called *the fundamental group* of X with the base point x .

We use a special kind of covering called a *regular covering*. This covering has a number of nice properties that will be useful in the construction. However, to define regular covering we need a few more facts.

THEOREM 2.5. *If $p: X' \rightarrow X$ is a covering with $x'_0 \in X'$, $x_0 \in X$ such that $p(x'_0) = x_0$ then the induced homomorphism $p^*: \pi(X', x'_0) \rightarrow \pi(X, x_0)$ is a monomorphism.*

Note that $p^*(\pi(X', x'_0))$ is a subgroup of $\pi(X, x_0)$. We define regular covering as follows.

DEFINITION 2.6. A covering $p: X' \rightarrow X$ is said to be *regular* if the group $p^*(\pi(X', x'_0))$ is a normal subgroup of $\pi(X, x_0)$ (that is, for each $g \in p^*(\pi(X', x'_0))$ and $h \in \pi(X, x_0)$ $hgh^{-1} \in p^*(\pi(X', x'_0))$).

We present two examples of regular coverings that will be used later.

Example 1. Let $X = F_2$ and X' be a surface of genus $k + 1$ ($k \geq 2$). We construct a k -fold regular covering p as in Fig. 1. Covering p transforms each of A'_i onto A and each of B'_i onto B (where A and B are two parts of the surface F_2 each corresponding to one handle and A'_i and B'_i are parts of the covering surface). As a consequence of this, the noncontractible cycles α'_i are mapped onto α and each part of γ' such that $\gamma' \in B'_i$ is mapped to γ . We use line $-\cdots-\cdots-$ to show graphically where parts of surfaces (here A , B or A'_i , B'_i) are glued together.

It is easy to see that we can generalize the above construction to a surface of genus g greater than two (see Fig. 2). Then, the genus of the covering surface is $k(g - 1) + 1$.

Let γ be a closed path on the surface X which begins at x and ends at y . Let $x' \in p^{-1}(x)$. γ defines uniquely the path from x' to some y' where $y' \in p^{-1}(y)$. Assume that γ is a closed path with a base point x . Let γ^k denote the closed path composed of the sequence of k paths γ . Consider the smallest number r such that the path in the covering space which starts at x' and corresponds to γ^r is closed. The number r is called the *developing number* of γ with base points x, x' , in the given covering. The closed path which corresponds to γ^r in the covering space is called a *generalized lift* of γ (we also call it an *r -developing lift* of γ). Note that for a regular covering the developing number does not depend on the choice of base points $x \in X$ and $x' \in p^{-1}(x)$.

In the first example γ' is a generalized lift of γ and its developing number is k . The path α has k generalized lifts $\alpha'_1, \dots, \alpha'_k$ each of them having developing number equal to one.

Property 2.7. In a regular covering, generalized lifts of equivalent closed paths have equal developing numbers.

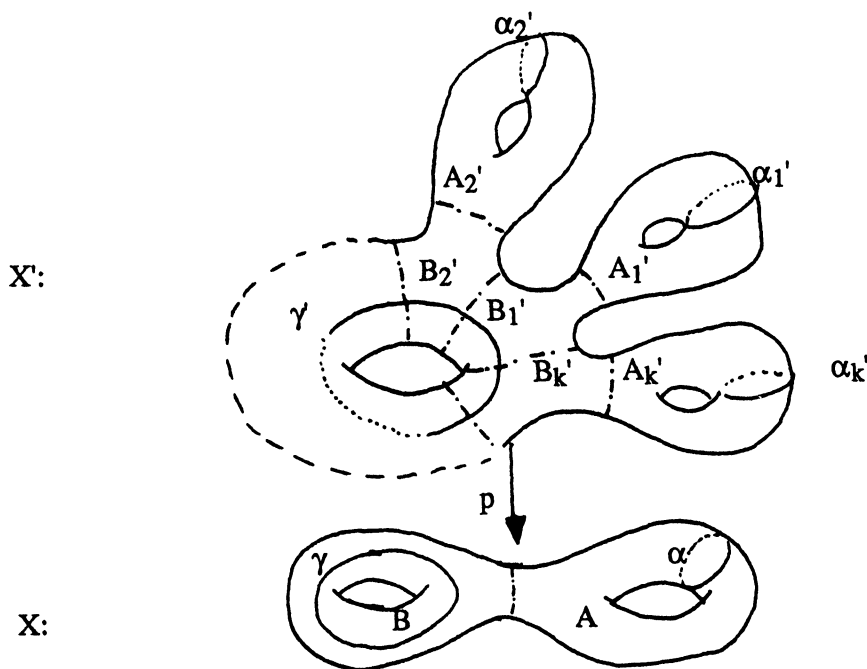
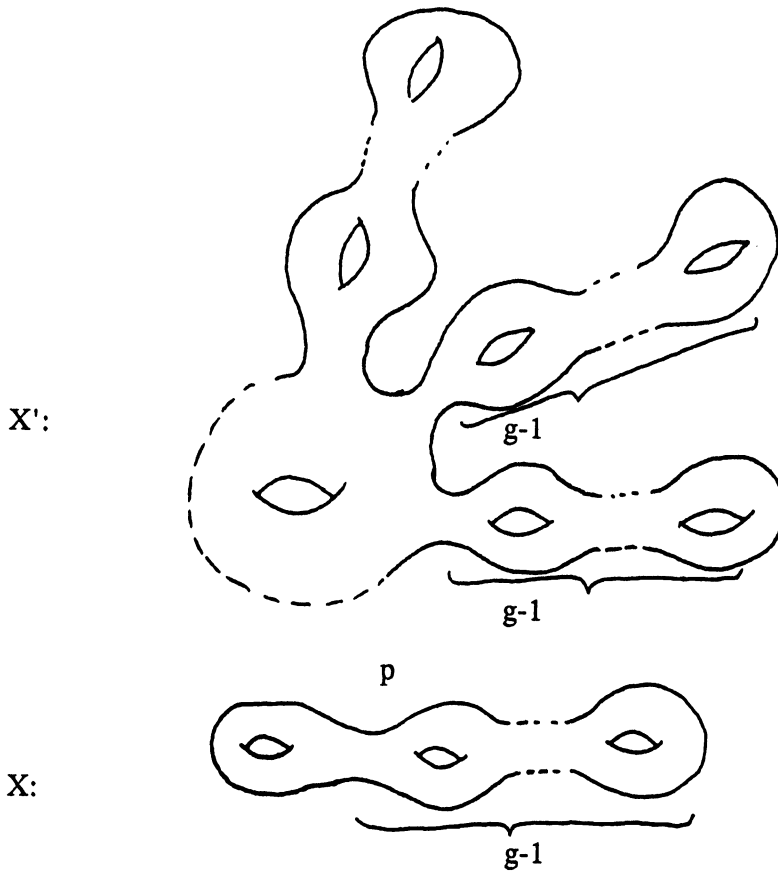


FIG. 1. A k -fold covering of F_2 .

FIG. 2. The k -fold covering of a surface of genus g .

We can divide all simple closed paths into two classes.

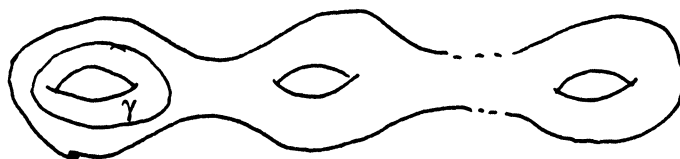
DEFINITION 2.8. A closed path γ on surface X is called a *separating path* if $X - \gamma$ is disconnected; otherwise it is called a *nonseparating path*.

It is well known that a nonseparating closed path is unique up to homeomorphism of X and therefore it can be presented as in Fig. 3(a). Similarly, if a separating closed path is noncontractible then it divides the surface of genus g into two surfaces of genus g_1 and g_2 such that $g_1, g_2 > 0$ and $g_1 + g_2 = g$ and it can be presented (up to homeomorphism of the surface) as in Fig. 3(b).

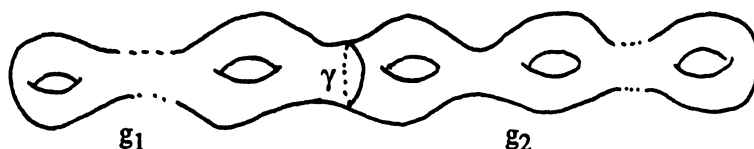
The following example shows that for any separating noncontractible closed path there exists a 2-fold regular covering such that the preimage of the given path consists of two nonseparating closed paths.

Example 2. Consider the covering $p: X' \rightarrow F_2$ where X' is a surface of genus 3 as shown in Fig. 4. (In particular, $p(A'_i) = A$, $p(B'_i) = B$ where A and B are two parts of the surface separated by noncontractible separating path γ .) In the example, the preimage of the simple closed noncontractible separating path γ is composed of two nonseparating simple paths γ'_1, γ'_2 .

We can generalize the above construction to the case of any simple, separating, noncontractible closed path γ on a surface. In this way, we obtain a regular 2-fold covering



a) a simple nonseparating closed path



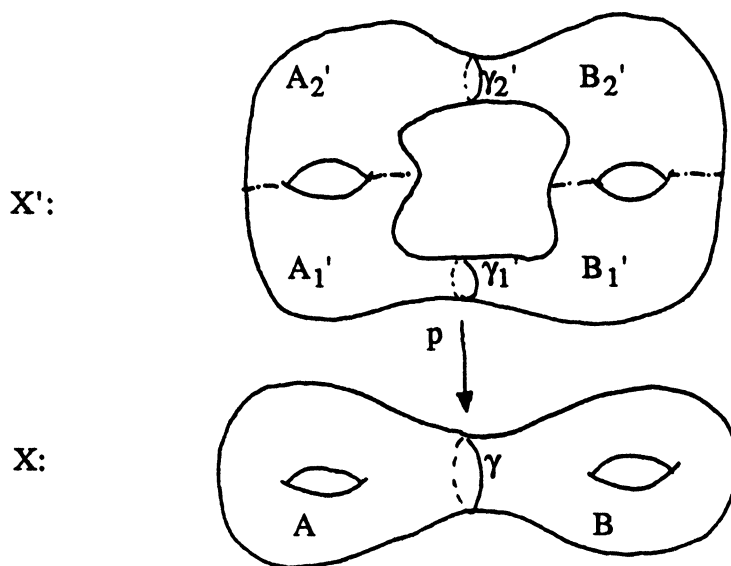
b) a simple separating closed path

FIG. 3

such that the preimage of γ is composed of two nonseparating simple paths γ'_1, γ'_2 (see Fig. 5).

We observe, in the presented examples, that in a k -fold covering there is a certain relationship between the genera of the covering and the base surfaces. This is a well known relationship which can be easily proved with an Euler characteristic argument and can be formulated as follows.

THEOREM 2.9. *In a k -fold covering of a base surface of genus $g > 0$ the covering surface has genus $k(g - 1) + 1$.*

FIG. 4. The 2-fold covering of F_2 .

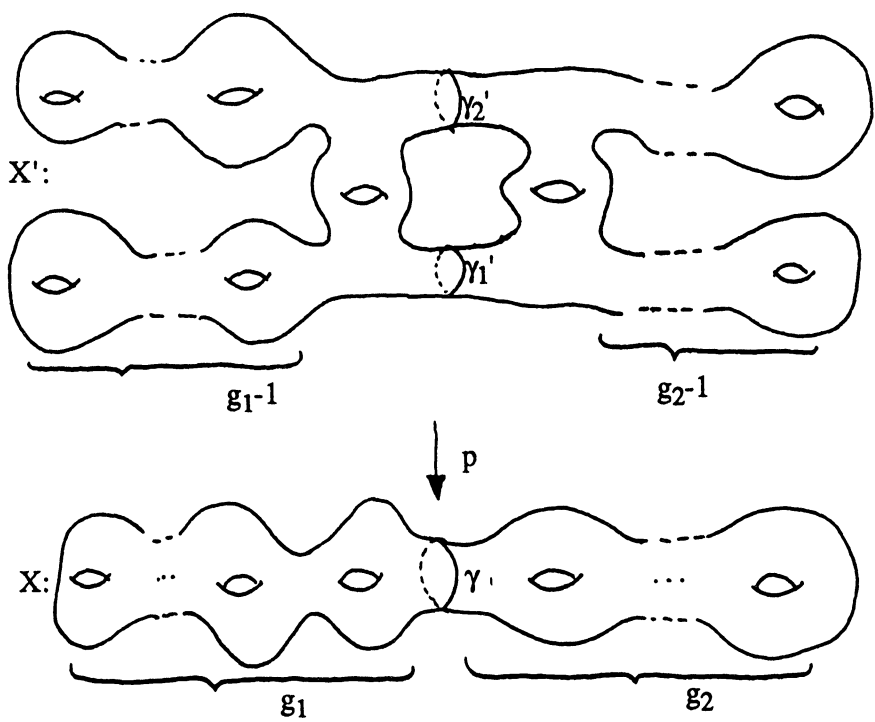


FIG. 5. The 2-fold covering of a surface of genus g .

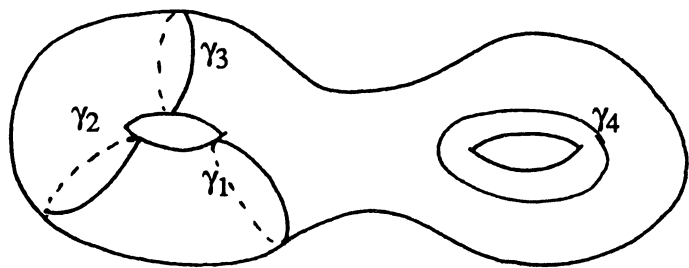


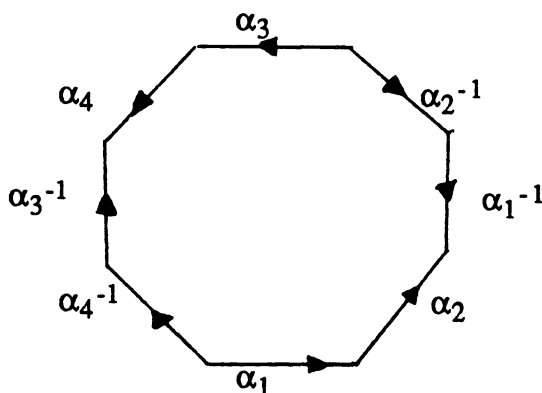
FIG. 6

In Definition 2.3, we introduced an equivalence relation on closed paths with a base point. This was the basic definition needed to define fundamental group and then regular covering. In the rest of the paper we use another equivalence relation on closed paths. Now we do not fix a base point.

DEFINITION 2.10. Two closed paths are equivalent if they are homotopic (one can be transformed continuously to another).

In Fig. 6, paths γ_1 , γ_2 , γ_3 are equivalent (we say also that they belong to the same homotopy class) and paths γ_1 , γ_4 are not.

3. The topological motivation for the construction. Our goal is to show how to construct a triangulation without short noncontractible cycles. As mentioned earlier, the idea of the construction is derived from elements of algebraic topology. In this section, we consider properties of families of closed paths on basic and covering surfaces. Starting

FIG. 7. The presentation of F_2 as a regular octagon.

with a family of noncontractible paths on F_2 we show how to construct a regular covering of F_2 in which all generalised lifts of the paths from this family are “long.”

Consider the presentation of F_2 as a regular octagon, where edges α_i and α_i^{-1} are pairwise identified as on Fig. 7.

Each edge α_i defines a noncontractible cycle on F_2 . We associate with each path γ on F_2 a value called complexity (which can be treated as a measure of its length) defined as follows.

DEFINITION 3.1. Let γ be a closed path on F_2 and $[\gamma]$ its homotopy class. Denote by $l(\gamma, \alpha)$ the number of crossings between closed paths α and γ . The *complexity* of γ is defined as follows:

$$\text{complexity}(\gamma) = \min_{\gamma' \in [\gamma]} \sum_{i=1}^4 l(\gamma', \alpha_i).$$

Informally, if we measure the length of a path by the number of crossings of the given path with edges α_i then the complexity of a closed path γ is the length of the shortest path in the homotopy class $[\gamma]$. Note that

$$\text{complexity}(\gamma) = \min_{\gamma' \in [\gamma]} \sum_{i=1}^4 l(\gamma', \alpha_i)$$

is realised by a closed path in so called *general position*; that is, a path that does not contain the common point of α_i and each crossing point is a perpendicular crossing of the given path with α_i (see Fig. 8).

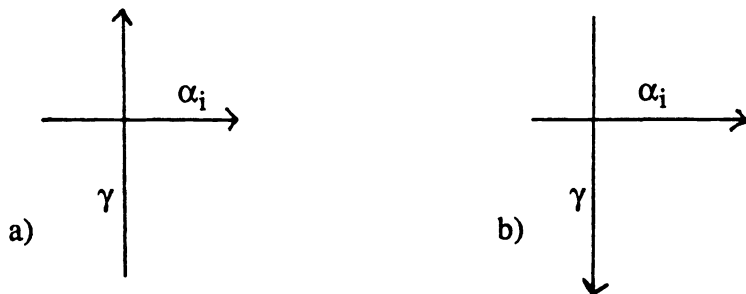


FIG. 8

Example 3. Consider the surface F_2 and the closed path corresponding to α_1 . This is a noncontractible path so its complexity is at least one. Note that α_1 is equivalent to α_1' (see Fig. 9) and therefore its complexity is exactly one.

LEMMA 3.2. *There are at most 9^s nonhomotopic paths on F_2 of complexity less than or equal to s .*

Proof. We can associate with each closed path a word on the alphabet $\{\alpha_1, \dots, \alpha_4, \alpha_1^{-1}, \dots, \alpha_4^{-1}\}$ in the following way.

Start from any point on the path and travel along the path in any direction. The word associated with the path corresponds to the order in which the given path cuts the edges $\alpha_1, \dots, \alpha_4$ (without loss of generality it is enough to consider only paths in a general position). If an edge α_i is cut as on Fig. 8(a) then the corresponding symbol in the word is α_i ; otherwise, (Fig. 8(b)) this symbol is α_i^{-1} .

Let $\beta = \alpha_i^{\pm 1}$. We consider as equal the following pairs of words: $w_1\beta\beta^{-1}w_2$ and w_1w_2 , βw and $w\beta$. It is easy to see that if two paths have associated equal words then they are equivalent.

There are at most $(8^{s+1} - 1)/7 \leq 9^s$ different words of length less than or equal to s . So there are at most 9^s nonequivalent closed paths which do not cut edges α_i more than s times. Therefore there are at most 9^s nonequivalent paths of complexity less than or equal to s . $\square$

In Definition 3.1, we introduced the notion of complexity for a closed path belonging to F_2 . We extend this definition to the complexity of a closed path on a regular covering of F_2 in the following definition.

DEFINITION 3.3. Let X be a regular covering of F_2 . The complexity of a closed path on X is equal to the complexity of its projection on the base surface F_2 .

Note that the complexity of a closed path has the following properties:

P1: Complexity of an r -developing generalized lift of γ is equal to r times complexity of γ . In particular, complexity of a generalized lift of γ is always greater than or equal to the complexity of γ .

P2: Complexity of two paths whose projections on F_2 are homotopic are equal. In particular, complexity of two homotopic paths are equal.

The main result of this section is stated by the following theorem.

THEOREM 3.4. *For any constant h there exists a covering surface X of F_2 such that the genus g satisfies $c \log^* g \leq h$ for some universal constant c (independent of the constant h) and all noncontractible closed paths on X have complexity greater than or equal to h .*

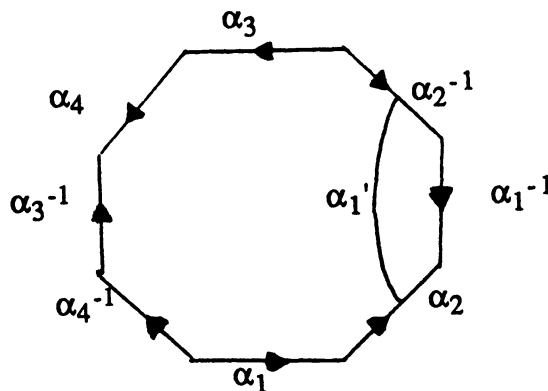


FIG. 9

To prove this theorem, we start with a lemma concerning simple closed paths.

LEMMA 3.5. *Let $\gamma_1, \dots, \gamma_k$ be a family of simple, closed, noncontractible and nonhomotopic paths on a surface X . There exists an m -fold regular covering of X such that each generalized lift of γ_i ($i = 1, \dots, k$) has developing number at least two and $m \leq 256^k$.*

Proof. For each γ_i , construct a covering space X_i such that at least one generalized lift of γ_i has developing number equal to two. For each nonseparating γ_i use the construction from Example 1. For each separating γ_i first use the construction from Example 2 to obtain two nonseparating generalized lifts of γ_i and then apply the construction from Example 1 to one of those lifts.

For each i , X_i is at most a 4-fold covering. By Theorem A.4 (see the Appendix) we can construct an m -fold regular covering X' of X such that X' covers also each of X_i and $m \leq (4^4)^k = 256^k$. $\square$

Consider a covering $p: X \rightarrow F_2$. Let Γ be the family of all closed noncontractible paths on X and let

$$r = \min_{\gamma \in \Gamma} \text{complexity}(\gamma).$$

Then the following lemma holds.

LEMMA 3.6. *If γ is a closed path from the family Γ , with self-crossings such that it is not homotopic to any path without self-crossings, then $\text{complexity}(\gamma) \geq 2r$.*

Proof. The lemma follows from the observation that if a closed path γ is composed of two closed noncontractible paths, then it cuts edges α_i in at least $2r$ points (we can again consider, without loss of generality, only paths which do not contain the common point of edges α_i). Furthermore, each path homotopic to γ is composed of two such closed paths. $\square$

Proof of Theorem 3.4. Construct the covering space in the following inductive way: Let $X_0 = F_2$. From Lemma 3.2, there exists at most 9^h nonhomotopic, noncontractible paths of complexity less than or equal to h . Consider those homotopy classes of closed paths on X_0 which possess a simple path as a representative. Denote those representatives by $\gamma_{1_0}, \dots, \gamma_{N_0}$. Note that $N_0 \leq 9^h$. By Lemma 3.5, we can construct a regular covering X_1 of X_0 in which generalized lifts of all these paths have complexity at least two. The complexity of the lifts of the rest of the paths cannot decrease (comparing complexity of those paths). Note that by Lemma 3.6, closed paths that do not have paths without self-crossing in their homotopy classes have complexity at least two. As well, their generalized lifts have complexity at least two. Therefore, complexity of generalized lifts of all noncontractible closed paths is at least two, so all closed noncontractible paths in X_1 have complexity at least two. Note that X_1 is a t_1 -fold covering where:

$$t_1 \leq 256^{(9^h)}.$$

The inductive step is as follows.

Let X_i be a regular t_i -fold covering of F_2 , and assume that all closed, noncontractible paths on X_i have complexity at least 2^i . First, construct a regular covering X'_{i+1} of X_i . To do this, consider those homotopy classes of noncontractible closed paths on F_2 which possess a representative with a simple generalised lift to X_i of complexity less than or equal to h . Let $\gamma_{1_i}, \dots, \gamma_{N_i}$ be the simple generalised lifts of those representatives. Observe again that $N_i \leq 9^h$. By Lemma 3.5, we can construct a regular at most $256^{(9^h)}$ -fold regular covering such that the complexity of all generalised lifts of $\gamma_{1_i}, \dots, \gamma_{N_i}$ is doubled. Note that by Theorem A.1 (see the Appendix) X'_{i+1} is at most a $t_i \cdot 256^{(9^h)}$ -fold covering of F_2 . Let X_{i+1} be a covering of X'_{i+1} which is also a regular

covering of F_2 . By Theorem A.4 of the Appendix, it is at most t_{i+1} -fold covering where

$$t_{i+1} \leq (t_i \cdot 256^{(9^h)})^{(t_i \cdot 256^{(9^h)})}.$$

By an argument similar to that one in the first step all closed noncontractible paths in X_{i+1} have complexity at least 2^{i+1} .

Repeat this construction until $2^i \geq h$. We need at most $\lceil \log h \rceil$ steps to do so. Denote by $\tilde{X}$ the obtained covering space of F_2 . By the construction, the complexity of all paths on $\tilde{X}$ is greater than or equal to h . $\tilde{X}$ is at most a $\tilde{t}$ -fold covering where $\tilde{t} \leq t_{\lceil \log_2 h \rceil}$. By Lemma 2.8 the genus of $\tilde{X}$ is less than or equal to $\tilde{t} + 1$. Let $b_1 = 256^{(9^h)}$ and inductively $b_{i+1} = (b_1 b_i)^{(b_1 b_i)}$. By the construction, $t_i \leq b_i$. Let $b = b_j$ where $j = \lceil \log h \rceil$. We will show that $h \geq \log^* b$. Note that:

$$(i) \log b_{i+1}^3 \leq b_i^3 \text{ for all } i$$

$$[\text{because } \log b_{i+1}^3 = 3 \log b_{i+1} = 3 \log (b_1 b_i)^{(b_1 b_i)} = 3 b_1 b_i \log (b_1 b_i) \leq 6 b_i^2 \log b_i \leq b_i^3].$$

Note as well that

$$(ii)$$

$$\begin{aligned} \log \log b_1^3 &= \log \log (256^{(9^h)})^3 = \log (3 \cdot 9^h \log 256) \\ &= \log 3 + h \log 9 + \log \log 16 \leq 4h + 4 \quad \text{for } h \geq 2. \end{aligned}$$

By the definition of j we have

$$(iii) \log (4h + 4) \leq j + 3,$$

and obviously

$$(iv) \log \cdots \log (j + 3) < 1$$

$j + 2$ times.

From (i)–(iv) we obtain

$$\begin{aligned} \log \cdots \log b_j &< 1; \\ 2j + 4 \text{ times} \end{aligned}$$

therefore,

$$\begin{aligned} \log \cdots \log \tilde{t} &< 1 \\ 2j + 4 \text{ times.} \end{aligned}$$

So $\log^* \tilde{t} < 2j + 4$ and for $h \geq 12$, $2j + 4 \leq h$ holds so $\log^* \tilde{t} < h$ and, therefore, $\log^* g \leq h$ for $h \geq 12$ and the theorem follows. $\square$

4. Construction of the triangulation. Let T be a triangulation of F_2 containing edges of the regular octagon from Fig. 7. Call such a triangulation a *base triangulation*. Let $n(T)$ denote the number of vertices in the triangulation T . Let X be a covering space of F_2 and p be the covering mapping. Triangulation T defines a triangulation T_X of the covering space such that vertices of T_X are preimages of vertices of T , and edges of T_X are defined by preimages of edges of T . Triangulation T_X is called a *covering triangulation* of the base triangulation T .

Note that the length of a closed path in T_X is greater than or equal to $\frac{1}{4}$ of its complexity (this follows from the fact that every edge of T_X projects on an edge of T and, by T 's definition, introduces an overall factor of at most four to the complexity). Therefore, we can use the ideas from the previous section to construct for any base triangulation T a covering triangulation T_X in which all noncontractible paths have length greater than any given constant. To disprove the conjecture, we use the fact that the ratios of the number of vertices to the genus in the covering and the base triangulations are almost equal.

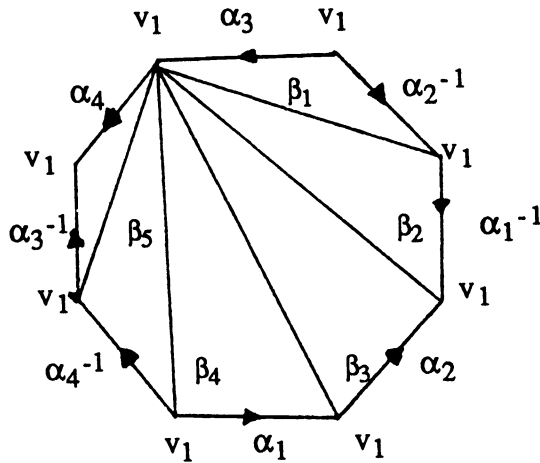


FIG. 10

LEMMA 4.1. Let T be a base triangulation and T_X its covering triangulation. Let $d = n(T)/2$. Then $d \leq (n(T_X)/g) < 2d$.

Proof. Assume that X is a k -fold covering of F_2 . Then $n(T_X) = k \cdot n(T)$ and by Theorem 2.8 the genus g of X is equal to $k + 1$. So

$$\frac{n(T_X)}{g} = \frac{n(T_X)}{k+1}$$

but

$$\frac{n(T_X)}{k+1} < \frac{k \cdot n(T)}{k} = 2d.$$

□

Using this lemma we can prove the following theorem.

THEOREM 4.3. For any $h > 0$, there exists a surface X_g of genus g such that for each $n \geq g - 1$ there exists a triangulation T_g of the surface X_g with n vertices such that $c \log^* g \leq h$ and T_g does not have noncontractible cycles shorter than $h/4$.

Proof. Let T be the triangulation with only one vertex v_1 and edges α_i ($i = 1, \dots, 4$) and β_i ($i = 1, \dots, 5$) as in Fig. 10.

Theorem 3.4 implies that there exists a regular $\tilde{l}$ -fold covering of F_2 with a covering space of genus $g = \tilde{l} + 1$ such that each noncontractible closed path has complexity at least h and $c \log^* g \leq h$ for some universal constant $c > 0$.

Consider the covering triangulation T_X . Denote by $|\gamma|$ the length of the path γ . By Definition 3.1, the complexity of a closed path γ in T_X is $\leq 4|\gamma|$. So $|\gamma| \geq$

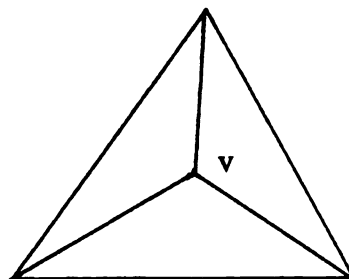


FIG. 11

complexity $(\gamma)/4$ and therefore $|\gamma| \geq h/4$. The triangulation T_X has $\tilde{t} = g - 1$ vertices. We can add to any triangulation a vertex, say ν , without decreasing the length of any noncontractible cycle. This can be done by the construction given in Fig. 11 so the theorem follows for each $n \geq g - 1$. $\square$

5. Conclusion. In this paper, we present a technique which allows us to construct a triangulation of an orientable surface with no short noncontractible cycles. In particular, Theorem 4.3 shows that $C(n, g)$ is not $O(\sqrt{n/g})$. Corollary 4.4 is the first step in a proof that $C(n, g)$ is $\Omega(\sqrt{n/g} \log^* g)$. To finish the proof, we should extend this corollary to all but finitely many pairs (n, g) . There are two main ideas behind such an extension:

(i) To extend it over all n , we can use the idea of construction of collars of triangulation along each of α_i . They should be constructed such that if the collars are of depth d , then if a path γ has complexity l then $|\gamma| \geq dl/4$.

(ii) To extend it over all g , we can repeat the whole construction for surfaces with a boundary and construct collars along the boundary as well. Then we can glue two surfaces along boundaries in such a way that the length of the shortest noncontractible cycle will not decrease.

Our construction to disprove Hutchinson's conjecture is elementary. Using the same technique more carefully (applying results of Hall [1]) we can improve Theorem 3.4 to $\sqrt{n/g} \log \log g$. To keep the paper self-contained, we do not present the improvement. On the other hand, we hope to prove our conjecture that $C(g, n) = \Theta(\sqrt{n/g} \log g)$.

Appendix. Here we prove a theorem concerning the existence of a covering space having properties needed in the proof of Theorem 3.4. For this proof, we need some results from group theory (for full proofs see [5]).

LEMMA A.1. *Let P, H be subgroups of a group G of finite index. If the index of P in G is k and the index of H in G is l then the index of $H \cap P$ in G is less than or equal to kl .*

LEMMA A.2. *Let H be a subgroup of G of a finite index k and let $g_1, \dots, g_k$ be representatives of the right cosets. Then $H^* = g_1 H g_1^{-1} \cap \dots \cap g_k H g_k^{-1}$ is a normal subgroup of G .*

Note that H^* is a subgroup of H .

LEMMA A.3. *Let $H_1, \dots, H_N$ be subgroups of G with indices $k_1, \dots, k_N$. There exists a normal subgroup H^* of G such that H^* is also a subgroup of H_i for each i and has index less than or equal to $k_1^{k_1} \dots k_N^{k_N}$.*

Let $p_1: X_1 \rightarrow X, \dots, p_N: X_N \rightarrow X$ be k_i -fold coverings. Let $x_1 \in X_1, \dots, x_N \in X_N$ and $x \in X$ be base points such that $p_i(x_i) = x$. Note that $p_1^*(\pi(X_1, x_1)), \dots, p_N^*(\pi(X_N, x_N))$ are subgroups of $\pi(X, x)$ and index of $p_i^*(\pi(X_i, x_i))$ in $\pi(X, x)$ is k_i .

THEOREM A.4. *There exists a k -fold regular covering X' of the base space X which also covers each of X_i ($i = 1, \dots, N$) such that $k \leq k_1^{k_1} \dots k_N^{k_N}$.*

Proof. By Lemma A.3, we can construct a normal subgroup G of $\pi(X, x)$ of index k where $k \leq k_1^{k_1} \dots k_N^{k_N}$ and G is a subgroup of every $p_i^*(\pi(X_i, x_i))$. But for any normal subgroup of $\pi(X, x)$ we can construct uniquely the corresponding regular covering (see [3] or [4]). If G is a subgroup of $p_i^*(\pi(X_i, x_i))$ for each i , then the constructed regular covering also covers each of X_i . $\square$

Note added in proof. Recently we have improved Theorem 4.3 to show that

$$C(g, n) = \Omega(\sqrt{n/g} \sqrt{\log g})$$

for n and g big enough. We use a result of Erdős and Sachs (*Reguläre Graphen gegebener Tailenweite mit minimaler Knotenzahl*, Wiss. Z. Univ. Halle-Wittenberg, Math.-Nat.

R., 12 (1963), pp. 251–258) and a method similar to that used by Buser (*Riemannsche Flächen mit grosser Kragenweite*, Comment. Math. Helvetici, 53 (1978), pp. 395–407).

REFERENCES

- [1] M. HALL, *Coset representation in free groups*, Trans. Amer. Math. Soc., 49 (1949), pp. 422–432.
- [2] J. HUTCHINSON, *On short noncontractible cycles in embedded graphs*, SIAM J. Discrete Math., 1 (1988), pp. 185–192.
- [3] C. KOSNIOWSKI, *A First Course in Algebraic Topology*, Cambridge University Press, Cambridge, 1980.
- [4] J. MASSEY, *Algebraic Topology; An Introduction*, Harcourt, Brace and World, New York, 1967.
- [5] T. PRZYTICKA AND J. H. PRZYTICKI, *On lower bound for short noncontractible cycles in embedded graphs*, Technical Report 88-2, Dept. of Computer Science, The University of British Columbia, Vancouver, 1988.

ON APPROXIMATE SOLUTIONS FOR COMBINATORIAL OPTIMIZATION PROBLEMS*

HANS ULRICH SIMON†

Abstract. The usefulness of a special kind of approximability-preserving transformations (called continuous reductions) among combinatorial optimization problems is demonstrated. One common measure for the approximability of an optimization problem is its best performance ratio. This parameter attains the same value for two problems (up to a bounded factor) whenever they are mutually related by continuous reductions. Therefore, lower and upper bounds or gap-theorems valid for a particular problem are transferred along reduction chains. In this paper, continuous reductions are used for the analysis of several basic combinatorial problems including graph coloring, consistent deterministic finite automaton, covering by cliques, covering by complete bipartite subgraphs, independent set, set packing, and others. The results obtained and the methods involved are a contribution towards a systematic classification of NP-complete problems with regard to their approximability.

Key words. combinatorial optimization, reductions, NP-completeness, approximation algorithms, graph coloring, consistent automata, partitioning, covering, independent set, set packing

AMS(MOS) subject classification. 68C25

1. Introduction. The theory of NP-completeness (see [9]) provides many tools for proving and understanding the intractability of combinatorial decision problems. A major tool is given by the notion of reductions. This notion is, however, not satisfactory for combinatorial optimization problems because it happens that nearly optimal solutions of one problem are mapped to poor solutions of another one.¹ We are interested in a careful design of reductions which makes them approximability-preserving. Although there already exist many particular examples for this, only few attempts (see [2], [3], [25], [20], [21], [26], [6], [4]) have been made to define an appropriate notion formally. This paper deals with the notion of continuous reductions from [26] and applies it to several basic combinatorial problems including graph coloring, consistent deterministic finite automaton, covering by cliques, covering by complete bipartite subgraphs, independent set, set packing, and others. The results obtained and the methods involved are a contribution toward a systematic classification of NP-complete problems with regard to their approximability.

The paper is organized as follows: § 2 contains the basic definitions concerning NP-complete problems and approximation algorithms. In particular, measures for the quality of an approximation algorithm (the performance ratios) and for the approximability of an optimization problem (the best performance ratios) are defined.

Section 3 begins with a discussion of the phenomenon of noncontinuous reductions. This discussion motivates the following formal definition of continuous reductions. Easy examples and counterexamples are described.

Section 4 investigates the problem of computing the smallest deterministic finite automaton that is consistent with a given finite sample of words. This problem plays an important role in the context of learning and prediction algorithms. It is shown that finding an automaton of less than twice the smallest size is NP-complete. The proof uses a continuous reduction from graph coloring to this problem.

Section 5 investigates the problem of covering the edges of a bipartite graph by a minimal number of complete bipartite subgraphs. This problem is closely related to the

* Received by the editors December 5, 1988; accepted for publication (in revised form) June 9, 1989.

† Fachbereich Informatik, University of Saarlandes, D-6600, Saarbrücken, Federal Republic of Germany.

¹ Although optimality is preserved.

computation of the nondeterministic communication complexity of discrete functions. It is shown that finding a cover of less than twice the smallest size is NP-complete. As in § 4, the proof uses a continuous reduction from graph coloring to this problem.

Algorithms for covering and partitioning problems that proceed round by round in a greedy fashion have iteratively to call a subprocedure for another optimization problem called the slave problem for the covering or partitioning problem at hand. The latter is called the master problem. Section 6 shows how the “master and slave”-relation can be combined with the technique of continuous reductions for getting approximability-preserving transformations.

Section 7 analyzes relaxations and restrictions of the independent set and the set packing problems. The gap-theorem of Garey and Johnson [9] (stating that the best performance ratio for an independent set is either 1 or ∞) is extended to other problems. This section also describes a useful combination of continuous reductions and the method of rounding and scaling.

A survey on the examined optimization problems and the continuous reductions among them is given in Fig. 3 of § 8. This section also discusses extensions of the notion of continuous reductions and several open problems.

2. Definitions and notations. This section briefly defines the basic notations concerning NP-complete problems and approximation algorithms. It can be skipped by readers who are familiar with the theory of NP-completeness.

Each combinatorial optimization problem Π consists of

- a set $\text{INPUT}(\Pi)$ of problem instances I of size $|I|$,
- for each instance I a finite set $S(I)$ of legal solutions,
- for each solution $\sigma \in S(I)$ its value $|\sigma|$ called profit in the case of maximization and cost in the case of minimization.

The sizes of problem instances² and the values of legal solutions are assumed as non-negative integers. The value of an optimal solution for I is denoted by $\text{opt}(I)$. An approximation algorithm A for Π is an algorithm that produces a legal solution $\sigma \in S(I)$ with a value $A(I) = |\sigma|$ for each given instance I and has a running time³ polynomially bounded in $|I|$. If the decision problem associated with Π is NP-complete, an approximation algorithm for Π cannot produce optimal solutions for all instances (unless $P = \text{NP}$). However, it may come “close” to the optimum. One common notion, formalizing what we mean by close, is the performance ratio of A on I defined as

$$R_A(I) = \begin{cases} \text{opt}(I)/A(I) & \text{in the case of maximization,} \\ A(I)/\text{opt}(I) & \text{in the case of minimization.} \end{cases}$$

Its value is 1 if the algorithm produces an optimal solution for I and is equal to the multiplicative deviation from the optimum in general. The function $R_A(n, m)$ is defined as follows:

$$R_A(n, m) = \max \{ R_A(I) \mid I \in \text{INPUT}(\Pi), |I| \leq n, \text{opt}(I) \geq m \}.$$

The performance ratio function is the function

$$R_A(n) = R_A(n, 0).$$

The (absolute) performance ratio is defined as

$$R_A = \lim_{n \rightarrow \infty} R_A(n),$$

² Corresponding basically to the lengths of binary encodings of the input.

³ This notion refers to the random access machine with the logarithmic cost measure.

and the asymptotic performance ratio as

$$R_A^\infty = \lim_{m \rightarrow \infty} \lim_{n \rightarrow \infty} R_A(n, m).$$

The best (absolute) performance ratio for Π is given by

$$R_{\text{MIN}}(\Pi) = \inf \{ R_A \mid A \text{ is an approximation algorithm for } \Pi \}.$$

The best asymptotic performance ratio for Π is defined analogously. $R_{\text{MIN}}(\Pi)$ has the value 1 if and only if there is an approximation algorithm A_ε for Π with $R_{A_\varepsilon} < 1 + \varepsilon$ for each $\varepsilon > 0$. The family A_ε is then called an *approximation scheme*. The scheme is called *full* if the family A_ε can be replaced by a uniform algorithm A with ε being an additional part of the input. The running time of A must be polynomial in $|I|$ and $\lceil 1/\varepsilon \rceil$. The problem Π is called *strong NP-complete* if it remains NP-complete for polynomially (in $|I|$) bounded values of the numerical input parameters occurring in $|I|$. In this case, no full approximation scheme exists (unless $P = \text{NP}$). For more details see [9].

3. Continuous reductions. This section contains the definition of continuous reductions and several easy examples and counterexamples. One might think that reductions among NP-complete problems can always be used to convert approximation algorithms for one problem into approximation algorithms for the other. This is, however, not generally the case. There are several reasons for noncontinuous relations between the respective approximate solutions. One reason, which is illustrated in [9], [21] for instance, is that the respective cost or profit functions may be reversely proportional. As an example from [9], consider the following optimization problems of independent set (IS) and vertex cover (VC):

IS Given an undirected graph $I = (V, E)$, compute a set $U \subseteq V$ of maximum size which is independent, i.e., $(x, y) \notin E$ for all $x, y \in U$.

VC Given an undirected graph $I = (V, E)$, compute a set $C \subseteq V$ of minimum size which is a vertex cover, i.e., at least one endpoint of each edge belongs to C .

Obviously, a subset of V is independent if and only if its complement is a vertex cover. Thus, there is an independent set of size B if and only if there is a vertex cover of size $|V| - B$. This relationship establishes mutual reductions between these problems which are far apart from being continuous. In fact, the vertices of a maximal matching always form a vertex cover of at most twice the optimal size whereas no approximation algorithm for IS with finite performance ratio is known to date (see [9], [10]).

Reductions among NP-complete problems usually map the pair (I, B) , where I denotes the input instance of one problem and B a bound on the values of the solutions, to a corresponding pair (I', B') for the other problem such that the following holds: there exists a solution for I which matches⁴ the bound B if and only if there exists a solution for I' which matches B' . A frequent source of noncontinuous relations between the approximate solutions is the fact that I' may depend on B . This is illustrated in more detail in the Appendix.

These discussions suggest the following definitions: let Π and Π' be optimization problems of the same type (either they are both maximization problems or they are both minimization problems). For $I \in \text{INPUT}(\Pi)$, $I' \in \text{INPUT}(\Pi')$ and rationals $a > 0$, $c \geq 0$, a mapping γ from $S(I)$ to $S(I')$ is called (a, c) -bounded if for all $\sigma \in S(I)$

$$\begin{cases} |\gamma(\sigma)| \geq \frac{1}{a} \cdot |\sigma| - c & \text{in the case of maximization,} \\ |\gamma(\sigma)| \leq a \cdot |\sigma| + c & \text{in the case of minimization.} \end{cases}$$

⁴ Matching means that the achieved cost (profit) is not greater (smaller) than B .

A reduction $\omega: \text{INPUT}(\Pi) \rightarrow \text{INPUT}(\Pi')$ from Π to Π' is called *asymptotic continuous* (with bounds $a > 0$, $b > 0$, $c \geq 0$, $d \geq 0$) if for all $I \in \text{INPUT}(\Pi)$ there exist mappings

$$\alpha_I: S(I) \rightarrow S(\omega(I)) \text{ and } \beta_I: S(\omega(I)) \rightarrow S(I)$$

such that:

- α_I is (a, c) -bounded and β_I is (b, d) -bounded.
- For all $I \in \text{INPUT}(\Pi)$ and all $\sigma' \in S(\omega(I))$, $\omega(I)$ and $\beta_I(\sigma')$ can be found in polynomial time.

The product $r = a \cdot b$ is called the *expansion* of the reduction. The reduction is called *absolute continuous* (or simply continuous) if $c = d = 0$ in addition. We use the short notation $\Pi \xrightarrow{r} \Pi'$ in this case.

A continuous reduction can be combined with an approximation algorithm A' for Π' to an approximation algorithm A for Π according to Fig. 1. With these notations, it is shown in [26] that the following hold.

LEMMA 3.1. *If $\Pi \xrightarrow{r} \Pi'$ then $R_A \leq r \cdot R_{A'}$.*

COROLLARY 3.2. *If $\Pi \xrightarrow{r} \Pi'$ then $R_{\min}(\Pi) \leq r \cdot R_{\min}(\Pi')$.*

The corresponding results hold for the asymptotic performance ratios if the reduction is merely asymptotic continuous. We say that there is a *weak* continuous reduction from Π to Π' with an expansion of r if each approximation algorithm A' for Π' can be converted into an approximation algorithm A for Π such that $R_A \leq r \cdot R_{A'}$. The corresponding short notation is $\Pi \xrightarrow{r} \Pi'$. Weak asymptotic continuous reductions are defined analogously. The following definitions refer to continuous reductions but are easily extended to asymptotic or weak continuous reductions. Two optimization problems are called *equivalent* if there exist mutual continuous reductions with an expansion of 1. The short notation $\Pi \xrightarrow{1} \Pi'$ is equivalent to: for all $\varepsilon > 0$ there is a continuous reduction from Π to Π' with an expansion of $1 + \varepsilon$. Two optimization problems are called *approximate equivalent* if this relationship is established in both directions.

The simplest example for a continuous reduction with an expansion of 1 occurs if Π is a subproblem of Π' . The mappings ω , α_I , β_I are the respective identity functions in this case.

Another easy example is obtained from the well-known mutual reductions (from [3]) between the independent set (IS) and the set packing (SP) problem. The latter is defined as follows:

SP Given a system $S = S_1, \dots, S_r$ of finite sets, compute a maximum *set packing*, i.e., a maximum number of pairwise disjoint sets from the system.

A continuous reduction from IS to SP with an expansion of 1 is obvious from the following construction: we associate with each vertex $v \in V$ the set of its incident edges. A subset $U \subseteq V$ is then independent if and only if the corresponding subsystem of sets is a set packing. The converse reduction from SP to IS with an expansion of 1 is obtained from

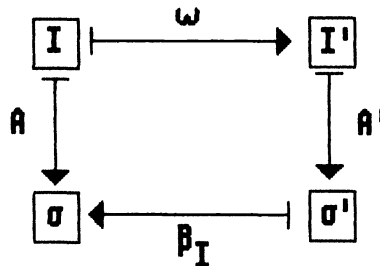


FIG. 1. The approximation algorithm A corresponding to A' and ω .

the following construction: we associate a vertex with each set of the system. Two vertices are connected by an edge if and only if the corresponding sets have a nonempty intersection. A subsystem of sets is then a set packing if and only if the corresponding vertices are independent. It follows from these discussions that IS and SP are equivalent.

Vertices in a graph form a *clique* if they are pairwise adjacent. The problem of computing a maximum clique is denoted by clique (C). The problems IS and C are obviously equivalent.

This section is closed by the following list of results which are stated without proof.

LEMMA 3.3. *Continuous reductions are transitive. More formally:*

$$\text{If } \Pi \xRightarrow{r} \Pi' \text{ and } \Pi' \xRightarrow{r'} \Pi'' \text{ then } \Pi \xRightarrow{r \cdot r'} \Pi''.$$

COROLLARY 3.4. *If $\Pi \xRightarrow{1+\epsilon} \Pi'$ and $\Pi' \xRightarrow{1+\epsilon} \Pi''$ then $\Pi \xRightarrow{1+\epsilon} \Pi''$.*

The analogous results hold for asymptotic and weak continuous reductions.

It is possible to decide for each constant natural B in polynomial time whether a given graph contains a clique (or an independent set) of size B . The following holds for problems of this kind.

LEMMA 3.5. *If for each constant bound B the decision problem corresponding to Π and B is solvable in polynomial time, then a weak asymptotic continuous reduction from Π to Π' with an expansion of r implies that for all $\epsilon > 0$ there is a weak (absolute) continuous reduction from Π to Π' with an expansion of $r + \epsilon$.*

The gap-theorem of Garey and Johnson [9] says that any approximation algorithm for IS with a bounded performance ratio can be converted into an approximation scheme. Gap-theorems have in general the following form for some bound $c \geq 1$:

$$R_{\text{MIN}}(\Pi) \in [1, c] \cup \{\infty\}.$$

LEMMA 3.6. *If there is a weak continuous reduction from Π to Π' with an expansion of r and a gap-theorem for Π with a bound c , then for all $\epsilon > 0$ there is a weak continuous reduction from Π to Π' with an expansion of $\min\{c + \epsilon, r\}$.*

4. Consistent deterministic finite automata. This section investigates a specific optimization problem: the problem of computing the smallest deterministic finite automaton which is consistent with a given sample of words (CDFA). CDFA is formally defined as follows:

CDFA Σ denotes a 2 letter alphabet and Σ^* the corresponding free monoid of words.

A *sample* is given by two distinguished finite sets $P \subset \Sigma^*$ of *positive* and $N \subset \Sigma^*$ of *negative examples*. The objective of CDFA is computing a deterministic finite automaton (DFA) with a minimum number of states which is *consistent* with the sample, i.e., it accepts all words from P and rejects all words from N .

CDFA plays an important role in the context of learning and prediction algorithms. A brief description of this relationship is in § 8.

It is known that CDFA is NP-complete (see [11]). In 1988, Li and Vazirani (see [17]) proved that finding an automaton of less than $\frac{9}{8}$ times the smallest size is also NP-complete. In the same year, we strengthened this result by replacing the factor $\frac{9}{8}$ with 2 (see [27]).⁵ The proof for the factor of 2 deals with continuous reductions and is described in the sequel. It turns out that, loosely speaking, CDFA is harder to approximate than graph coloring (GC) but easier than right congruent graph coloring (RCGC). The latter two problems are formally defined as follows:

⁵ This factor has been improved, recently, from 2 to $n^{(1/14)-\epsilon}$, where n denotes the size of the input and ϵ is an arbitrary positive constant (see [22]).

GC A *legal* coloring of an undirected graph $I = (V, E)$ is a mapping f which associates a color with each vertex such that adjacent vertices obtain different colors. The objective of GC is finding a legal coloring with a minimum number of colors.

RCGC An instance I of RCGC consists of an undirected graph $G = (V, E)$ and an injective mapping s which associates a string $s(v) \in \Sigma^*$ with each vertex v . In the following, we identify the vertices with their strings. We assume that the following conditions hold:

- V is closed with respect to prefixes, i.e., if $w \in V$ and v is a prefix of w then $v \in V$.
- For all $v, w \in V$ and $a \in \Sigma$, if $(va, wa) \in E$, then $(v, w) \in E$.

A *right congruent coloring* for I is a legal coloring f for G which also fulfills $f(va) = f(wa)$ if $f(v) = f(w)$ for all $a \in \Sigma, v, w, va, w, wa \in V$. The objective of RCGC is finding a right congruent coloring for I with a minimum number of colors.

THEOREM 4.1. $GC \xrightarrow{1+\epsilon} CDFA$ and $CDFA \xrightarrow{1} RCGC$.

Proof. 1) In order to show the first reduction, let us temporarily assume that the alphabet Σ has a variable size. There is a very simple continuous reduction with an expansion of 1 in this case. The reduction for the case of an alphabet with 2 letters is based on the same idea but involves more technical details.

We conceptionally give the edges in I an arbitrary, but fixed, orientation. We define $\Sigma = V \cup E$ and the sample of words as the set

$$\{ve \mid v \in V, e \in E \text{ and } v \text{ is an endpoint of } e\}.$$

A sample word ve is a positive example if and only if v is the tail of e .

If there is a legal coloring $f : V \rightarrow [0 : s - 1]$ of I , the following DFA A with s states $q_0, \dots, q_{s-1}$ is consistent with the sample: if I is an independent set, the sample is empty and A is chosen as a trivial automaton with 1 state. Otherwise, the following construction is applied: q_0 is the initial and unique accepting state. After reading v , A attains the state $q_{f(v)}$. The following letter e corresponds to an edge in I with exactly one vertex of color $f(v)$. After reading e , A attains the state q_0 if and only if the tail of e has the color $f(v)$.

Conversely, if there is a consistent DFA A for the sample of words, the states which are attained after reading the vertices v must induce a legal coloring in I because otherwise A could not differentiate between positive and negative examples. This construction leads to a continuous reduction of expansion 1.

We now perform an analogous construction for a 2-letter alphabet $\Sigma = \{INC, DEC\}$. Each input letter then carries exactly 1 bit of information (instead of logarithmically many as before). This loss of information must be compensated by the finite control. In order to maintain the relationship to legal colorings, we require the following notion: the *composition* $G \circ H$ of 2 graphs results if we replace each vertex x of G by a copy H_x of H and connect two copies H_x, H_y completely bipartitely if and only if (x, y) is an edge in G . If C_k denotes the clique of size k , our construction deals with the composition $J = C_k \circ I$. The parameter k has an adjustable value fixed appropriately below. We conceptionally

- number the vertices in I from 0 to $n - 1$,
- number the edges in I from 1 to m ,
- number the copies of I from 0 to $k - 1$,
- and orient an intra-edge, i.e., an edge in J which runs inside a copy of I , from small to big vertex numbers, and an inter-edge, i.e., an edge in J which runs between two copies of I , from small to big copy numbers.

The sample of words is defined as the following binary encoding of the aforementioned sample: if v is a vertex in J with vertex number i and copy number j , $e = \{v, w\}$ an inter-edge and h the copy number of w , then the string

$$INC^i DEC INC^j DEC^{\min\{j, h\}}$$

is inserted into the sample. In the case of an intra-edge e with edge number h , the string

$$INC^i DEC INC^j DEC^{j+h}$$

is inserted into the sample. A string for ve is a positive example if and only if v is the tail of e .

If $f : V \rightarrow [0 : s - 1]$ is a legal coloring of I with $s \geq 2$ colors, then the following DFA A with $n + s \cdot (m + k)$ states

$$a_i, b_{p,q} \text{ for } i = 0, \dots, n-1, p = 0, \dots, s-1 \text{ and } q = -m, \dots, 0, \dots, k-1$$

is consistent with the sample: A starts in the initial state a_0 . While reading the letter INC , the index of a is incremented. Thus, A is in the state a_i after reading INC^i . When reading the letter DEC in a state a_i , A transits to the state $b_{f(i),0}$. While reading the letter INC in a state of type b , the second index of b is incremented. Thus, A is in the state $b_{f(i),j}$ after reading $INC^i DEC INC^j$. While reading the letter DEC in a state of type b , the second index of b is decremented. If the input of A is a string w from the sample which corresponds to an inter-edge, A is finally in the state $b_{f(i),0}$ (accepting state) in the case of a positive example and in a state $b_{f(i),q}$ with $q > 0$ (rejecting state) in the case of a negative example. If w corresponds to an intra-edge, A is finally in the state $b_{f(i),-h}$ where h is the number of e . This state is defined as accepting if and only if the vertex of e with color $f(i)$ is the tail. It is evident from this discussion that A is consistent with the sample.

Conversely, we can construct a legal coloring for J with q colors from a consistent DFA A with q states. This holds because the states which are attained after reading the prefixes $INC^i DEC INC^j$ induce a legal coloring. Otherwise A could not differentiate between negative and positive examples. We then construct a coloring for I with q/k colors from the coloring for J in the obvious way.

The expansion of this continuous reduction is certainly bounded by $1 + (n + m)/k$. This expression is not greater than $1 + \varepsilon$ if we choose $k = \lceil (n + m)/\varepsilon \rceil$. This completes the proof for the first reduction.

2) In the second reduction, the instance $I' = (V', E')$ for RCGC is obtained from a sample $I = (P, N)$, $P, N \subset \Sigma^*$ as follows: V' is identified with the set of prefixes of $P \cup N$. A prefix x is connected with a prefix y if and only if the sample distinguishes them by a context on the right-hand side, i.e., there is a word v such that one of the words xv, yv is a positive and the other one is a negative example. I' certainly fulfills the two assumptions for instances of RCGC.

If f is a right congruent coloring for I' , a consistent DFA A for I is obtained as follows: the states of A are identified with the colors of f . The initial state is the color $f(\lambda)$ of the empty word. The transition function is defined by $\delta(f(v), a) = f(va)$ for all $a \in \Sigma$ and $v, va \in V'$. This transition function is well defined because f is right congruent. The accepting states are the colors of the positive examples.

Conversely, if A is a consistent DFA for the sample I , the states attained after reading the prefixes $x \in V'$ induce a right congruent coloring for I' . The expansion of this continuous reduction is 1. $\square$

An immediate consequence of this theorem and the fact (see [7]) that the best asymptotic performance ratio for GC is at least 2 is the following corollary.

COROLLARY 4.2. *The best asymptotic performance ratios for CDFA and RCGC are at least 2 (unless $P = NP$).*

5. Covering by complete bipartite subgraphs. This section investigates another specific optimization problem called covering by complete bipartite subgraphs (CCB):

CCB Given a bipartite graph $I = (V, E)$, compute a minimum system of complete bipartite subgraphs which is a *cover*, i.e., each edge is contained in at least one of the complete bipartite subgraphs.

CCB is closely related to the computation of the nondeterministic communication complexity of discrete functions. This relationship is briefly described in § 8. Here, we show that there are mutual continuous reductions between CCB and partitioning into cliques (PC). The latter problem is defined as follows:

PC Given an undirected graph $I = (V, E)$, compute a minimum partition of V into cliques.⁶

THEOREM 5.1. $PC \stackrel{1+\epsilon}{\Rightarrow} CCB$ and $CCB \stackrel{1}{\Rightarrow} PC$.

Proof. 1) In the first reduction, the bipartite graph $I' = (V', E')$ is constructed from the undirected graph $I = (V, E)$ in two stages. We first construct a bipartite version $I_B = (V_B, E_B)$ of I as follows: if V consists of n vertices $x_1, \dots, x_n$, then V_B consists of $x_1, \dots, x_n$ (the male vertices) and $x'_1, \dots, x'_n$ (the female vertices). For all $1 \leq i, j \leq n$, $(x_i, x'_j) \in E_B$ if and only if $i = j$ or $(x_i, x_j) \in E$. There are no edges between two male or between two female vertices. An edge $(x_i, x'_j) \in E_B$ is called *ascending* if $i < j$, *descending* if $i > j$ and *horizontal* if $i = j$.

I' consists of t^2 copies of I_B and several additional edges which connect male vertices of one copy with female vertices of another one. The parameter t has an adjustable value fixed appropriately below. We think of the t^2 copies as arranged in a $(t \times t)$ -array. For all $1 \leq i \leq n$ and $1 \leq p, q \leq t$, the i th male (female) vertex in copy (p, q) is denoted by (x_i, p, q) (by (x'_i, p, q)). Two vertices (x_i, p, q) and (x'_j, k, l) which belong to different copies of I_B (i.e., $(p, q) \neq (k, l)$) are connected by an edge if and only if

either they are in the same row (i.e., $p = k$) and $(x_i, x'_j) \in E_B$ is ascending
or they are in the same column (i.e., $q = l$) and $(x_i, x'_j) \in E_B$ is descending.

An edge in E' is called ascending (descending, horizontal) if and only if its projection onto E_B has this property.

Our construction has the following features: each ascending edge $e \in E_B$ has t copies in each row. The $2t$ endpoints of these copies form a complete bipartite subgraph of I' because all involved edges between males and females are ascending. The analogous assertion holds for descending edges and columns. Hence, all ascending and descending edges in E' can be covered by $2tm$ complete bipartite subgraphs (where $m = |E|$). The situation for horizontal edges is quite different. If $e, e' \in E_B$ are two (not necessarily different) horizontal edges and $(e, p, q), (e', k, l)$ belong to two different copies of I_B in I' , then the four corresponding endpoints do not form a complete bipartite subgraph because the two involved edges (from the male in (e, p, q) to the female (e', k, l) and from the male in (e', k, l) to the female in (e, p, q)) are neither both ascending nor both descending.

Each clique C in I corresponds to a complete bipartite subgraph of I_B in the obvious way. Hence, each partition of I into r cliques corresponds to a system of rt^2 complete

⁶ PC is obviously equivalent to GC but technically more convenient for the proof of the main result in this section.

bipartite subgraphs in I' . This system covers all horizontal edges. As already explained, the ascending and descending edges can be covered by $2tm$ complete bipartite subgraphs. The size of the whole cover is then $rt^2 + 2tm$.

We associate the subset H_B of horizontal edges with any complete bipartite subgraph B of I' . If H_B is not empty, it belongs to a unique copy of I_B . The male endpoints of the edges in H_B correspond to a clique in I . Thus, each cover of I' by s complete bipartite subgraphs induces t^2 partitions of I into cliques. There must be one partition whose size is bounded by s/t^2 .

It follows from these discussions that there is a continuous reduction from PC to CCB whose expansion is at most $1 + 2m/t$. This is smaller than $1 + \varepsilon$ if $t > 2m/\varepsilon$.

2) A continuous reduction from CCB to PC with an expansion of 1 is obvious from the following construction of $I' = (V', E')$ from $I = (V, E)$: we associate a vertex $v_e \in V'$ with each edge $e \in E$. Two different vertices $v_e, v_{e'} \in V'$ are connected by an edge in I' if and only if the endpoints of e and e' form a complete bipartite subgraph of I . $\square$

COROLLARY 5.2.

- *The best asymptotic performance ratio for CCB is at least 2 (unless $P = NP$).*
- *PC and CCB are approximate equivalent.*

6. Master and slave problems. Covering and partitioning problems are solvable by the following kind of greedy algorithm: proceed in rounds and in each round try to cover a maximum number of the remaining uncovered objects. The maximization problem which has to be solved during each round is called the *slave problem* of the covering or partitioning problem at hand. The latter is called the *master problem*. When we deal with graph problems, we call the substructure which is formed by the remaining uncovered vertices or edges the *surviving subgraph*. The surviving subgraph is initially identical to the input instance and has to be empty finally. During each round, it serves as an input for the slave problem. Johnson has pointed out [13] that optimal algorithms for the slave problem lead to approximation algorithms for the corresponding master problem whose performance ratio function is bounded by $\ln n$. More generally, if A' is an approximation algorithm for the slave problem, A the resulting approximation algorithm for the master problem and n the number of objects to be covered, then $R_A(n) \leq \ln n \cdot R_{A'}(n)$ (a proof for this is found in [24]). We denote this relationship by

$$\text{Master Problem} \xrightarrow{\ln(n)} \text{Slave Problem}.$$

The master problems, considered in this section, are partitioning into cliques (PC), covering by complete bipartite subgraphs (CCB) and covering by cliques (CC). The latter problem is defined as follows:

CC Given an undirected graph, compute a minimum system of cliques which is a cover, i.e., each edge is contained in at least one clique.

We know from § 5 that $\text{PC} \xrightarrow{1+\varepsilon} \text{CCB}$ and $\text{CCB} \Rightarrow \text{PC}$. In [15], the authors show a connection between PC and CC which has the following translation into our notation:

$$\text{PC} \xrightarrow{1+\varepsilon} \text{CC}.$$

The reduction

$$\text{CC} \xrightarrow{1} \text{PC}$$

follows from a construction similar to that in Part 2 of the proof for Theorem 5.1.

COROLLARY 6.1. *The problems PC, CC, and CCB are approximate equivalent.*

We want to show that the corresponding slave problems are equivalent. The slave problem for PC is obviously the problem clique (C) of computing a maximum clique

of a given undirected graph. The slave problem for CC is the following problem of edge weighted cliques (EWC): an instance I consists of an undirected graph and 0,1-weights of the edges. The edge-weight of a clique in I is the total weight of its edges. The objective is to find a clique with a maximum edge-weight. If EWC is used as a slave of CC, its initial instance is obtained from the input for CC by labelling each edge with weight 1. The surviving subgraph is obtained in general by relabelling all edges already covered by 0. Analogously, we define the slave problem EWCB (Edge Weighted Complete Bipartite Subgraph) for CCB.

THEOREM 6.2. *The problems C, EWC, and EWCB are equivalent.*

This theorem implies that for any of the three master problems and any of the three slave problems the relation Master Problem $\xrightarrow{\ln(n)}$ Slave Problem is valid (see also Fig. 3 in § 8 which gives a survey on all reductions).

Sketch of proof. A continuous reduction from C to EWCB with an expansion of 1 is obvious from the following construction: I_B denotes the bipartite version of I (see the proof of Theorem 5.1). The instance I' of EWCB is equal to I_B augmented with unit-weights on the horizontal edges and zero-weights on the remaining ones. The continuous reduction $C \Rightarrow \text{EWC}$ is based on a similar construction (replace the independent sets of male and female vertices in I_B by cliques of the same size, respectively; the edges of these cliques have weight 0). The reductions $\text{EWCB} \Rightarrow C$ and $\text{EWC} \Rightarrow C$ are similar to the corresponding reductions for the master problems. The details are omitted. $\square$

7. Problem hierarchies. A *problem hierarchy* of a given problem is obtained whenever we place additional restrictions or relaxations on the allowed instances. If we proceed to restrict a problem several times, we finally get a subproblem with a bounded best performance ratio. Iterative relaxation finally leads to lower bounds for the best performance ratio (unless $P = NP$). Between these extremes, we find versions of the problem for which the question of approximability is still unsolved. This section is concerned with the following problem hierarchies for IS and SP:

WIS Weighted independent set: each vertex has a nonnegative integer weight. The objective is finding an independent set with maximum total weight.

PWIS(k) Polynomially weighted independent set: this is like WIS except that the weights are not greater than n^k , where n denotes the number of vertices and $k > 1$ is a rational.

DWIS Degree weighted independent set: this is like WIS except that the weight of a vertex must be equal to its degree.

RIS Regular independent set: this is like IS except that the input graphs must be *regular*, i.e., all vertices have the same degree.

BDIS(k) Bounded degree independent set: this is like IS except that the degrees in the input graph are bounded by the natural k .

CB Complete bipartite subgraph in graphs: given an undirected graph $I = (V, E)$, compute a set $U \subseteq V$ of maximum size which forms a complete bipartite subgraph.

WSP, PWSP(k), SWSP, RSP, and BSSP are the corresponding problems in the SP-hierarchy, where the size of a set plays the role of the degree of a vertex. For instance, the objective of SWSP is to find a set packing which involves a maximum number of elements (instead of a maximum number of sets).

A survey on the continuous reductions among the problems of these hierarchies is given in Fig. 2. The reductions corresponding to the subproblem relation and $\text{RIS} \Rightarrow \text{DWIS}$ are obvious. Slight modifications of the mutual continuous reductions between IS and SP (described in § 3) lead to:

$$\text{RIS} \stackrel{1}{\Rightarrow} \text{RSP}, \text{DWIS} \stackrel{1}{\Rightarrow} \text{SWSP}, \text{PWIS}(k) \stackrel{1}{\Leftrightarrow} \text{PWSP}(k), \text{WIS} \stackrel{1}{\Leftrightarrow} \text{WSP}.$$

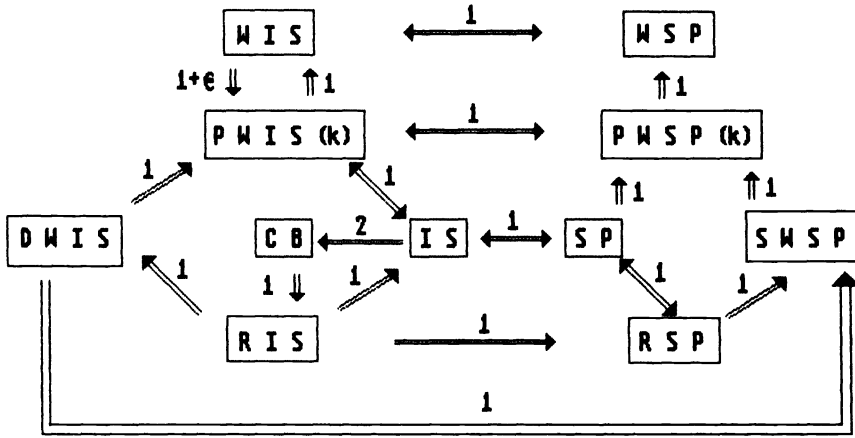


FIG. 2. Continuous reductions among the problems in the IS- and SP-hierarchy.

The remaining continuous reductions are established in Theorem 7.1.

THEOREM 7.1. 1. $WIS \xrightarrow{1+\epsilon} PWIS(k)$ for all k .

2. $PWIS(k) \xrightarrow{1} IS$ for all k .

3. $IS \xrightarrow{2} CB$.

4. $CB \xrightarrow{1} RIS$.

5. $SP \xrightarrow{1} RSP$.

Proof. The proof adopts the notions $\omega, I, I', \alpha_I, \beta_I$ from the definition of continuous reductions in § 3 (recall also the definition of (a, c) -bounded mappings from $S(I)$ to $S(I')$).

1. $WIS \xrightarrow{1+\epsilon} PWIS(k)$. We use the technique of *rounding and scaling* and show how it interacts with the idea of continuous reductions.⁷ The polynomial bound for the weights of the vertices is assumed as n^2 although an obvious modification of the following proof works for arbitrary bounds of the form n^k with $k > 1$. We fix an $\epsilon > 0$. For a constant number of vertices, an optimal independent set can be found by exhaustive search. Thus, we may assume that $n > 1 + 1/\epsilon$. In addition, we may assume that the maximum weight $W_{\max}$ in the instance I of WIS is greater than n^2 . Otherwise, a trivial reduction to $PWIS(2)$ is performed. We now set the instance $I' = \omega(I)$ identical to I except that the weights $w_1, \dots, w_n$ in I are scaled down to $\lfloor \lambda \cdot w_1 \rfloor, \dots, \lfloor \lambda \cdot w_n \rfloor$ with $\lambda = n^2/W_{\max}$. The largest weight in I' is then equal to n^2 . If we set β_I equal to the identity function, i.e., we choose for I the same independent set as for I' , then β_I is $(\lambda, 0)$ -bounded. Conversely, if an independent set U in I is given, we evaluate its scaled weight $W' = \sum_{i \in U} \lfloor \lambda \cdot w_i \rfloor$. If $W' > n^2$, we set $\alpha_I(U) = U$. Otherwise, we prefer the singleton solution in I' with maximum weight n^2 . This leads to the following estimation for the profit of $\alpha_I(U)$:

$$\begin{aligned} |\alpha_I(U)| &\geq \max \left\{ n^2, \sum_{i \in U} \lfloor \lambda \cdot w_i \rfloor \right\} \\ &\geq \max \left\{ n^2, \sum_{i \in U} \lambda \cdot w_i \right\} - n \\ &\geq \frac{n-1}{n} \cdot \sum_{i \in U} \lambda \cdot w_i. \end{aligned}$$

⁷ The technique of rounding and scaling has been usually combined with an optimal algorithm for the subproblem with polynomially bounded weights. Hence, it seemed to be useless for strong NP-complete problems (like WIS). In combination with continuous reductions, this restriction on the use of rounding and scaling is dropped.

The mapping α_I is $(n/\lambda \cdot (n-1), 0)$ -bounded. For $n > 1 + 1/\varepsilon$, the expansion of the reduction evaluates to $n/(n-1) < 1 + \varepsilon$.

2. $\text{PWIS}(k) \stackrel{1}{\Rightarrow} \text{IS}$. The polynomial weights in I can be made combinatorial by replacing each vertex i of weight w_i by an independent set U_i of size w_i . Two independent sets U_i and U_j are connected completely bipartitely if and only if (i, j) is an edge in I . An independent set in the resulting instance I' remains independent, if we replace non-empty subsets of the U_i by the whole set U_i for each i . The mappings $\alpha_{I'}$, $\beta_{I'}$ that are defined in the obvious way are then both $(1, 0)$ -bounded. This leads to a continuous reduction with an expansion of 1.

3. $\text{IS} \stackrel{2}{\Rightarrow} \text{CB}$. If we set $I' = \omega(I) = I$, each independent set in I can be considered as a complete bipartite subgraph in I' (with one empty vertex class). Conversely, if two vertex classes form a complete bipartite subgraph in I' , each single class, and in particular the larger one, forms an independent set in I . This leads to a continuous reduction with an expansion of 2.

4. $\text{CB} \Rightarrow \text{RIS}$. If $I = (V, E)$ is the instance of CB, the corresponding instance I' of RIS consists of two disjoint copies of I and the following edges between them: a vertex x_1 of copy 1 is connected with a vertex y_2 of copy 2 if and only if the corresponding pair $(x, y) \in V \times V$ does not belong to E . I' is a regular graph because each vertex has degree n . If U is an independent set in I' , U_i is the subset of U belonging to copy i for $i = 1, 2$ and V_i is the corresponding subset of V , then V_1 and V_2 form a complete bipartite subgraph of I . Conversely, if V_1 and V_2 form a complete bipartite subgraph of I and U_1 and U_2 are the corresponding sets in copy 1 and 2, then $U_1 \cup U_2$ is an independent set in I' . This leads to a continuous reduction with an expansion of 1.

5. $\text{SP} \Rightarrow \text{RSP}$. The sizes of the sets in an instance I of SP are easily equalized by padding the small sets with dummy elements. This leads to an instance of RSP and to a continuous reduction with an expansion of 1. $\square$

The reductions occurring in this theorem are composable. For instance, there is a reduction from SWSP to SP with an expansion of 1 running through SWSP, PWSP, PWIS, IS, SP. These composed reductions are not always the most elegant ones. For instance, Krentel [16] communicated to us two other nice mutual reductions between IS and SWSP. The idea of dummy elements, used in our reduction from SP to RSP, is borrowed from his construction.

We obtain the following corollary if we combine these continuous reductions with the gap-theorem for IS.

COROLLARY 7.2. *The best performance ratios of the problems*

RIS, IS, PWIS(k), DWIS, WIS, CB, RSP, SP, PWSP(k), SWSP, WSP

are simultaneously either 1 or ∞ .

In particular, all problems in this list are weak approximate equivalent (see also Fig. 3 in § 8).

We now switch over to the subproblems BDIS(k) and BSSP(k). These problems remain NP-complete for all $k \geq 3$ (see [8]). BSSP(2) is also known as the problem of computing a maximum matching in an undirected graph which is solvable in polynomial time. BDIS(2) is trivial because the input graphs are unions of disjoint paths and circles. A naive approximation algorithm for BDIS(k) and arbitrary k is the following:

begin

1. initialize G (the surviving subgraph) to the input graph;
2. initialize U (the independent set) to the empty set;
3. **while** G is nonempty **do**

- (a) insert the vertex x in G with minimum degree into U ;
- (b) remove x and all its neighbours from G **od**

end

We refer to this method as the *greedy algorithm* A_g . It is not hard to show that the performance ratio of this algorithm is bounded by $k - 1$ (the bound k is trivial).

Let $I = S_1, \dots, S_r$ denote an instance of BSSP(k) and ω, β_I the mappings of the continuous reduction from SP to IS from § 3. $I' = \omega(I)$ is an undirected graph with a vertex x_i for each set S_i and an edge between two vertices whenever the corresponding sets are not disjoint. The mapping $\beta_I \circ A_g$ applied to I' produces a solution for I . We have to show that the performance ratio of A_g on the instance I' is not greater than k . Although the degree of the vertices in I' is not necessarily bounded by k , the following observations are valid: when A_g inserts a vertex x_i into U , we conceptionally label each edge (x_i, x_j) with the smallest element in the intersection of S_i and S_j . There are at most k different labels because each label belongs to the set S_i whose size is bounded by k . Therefore, the subgraph induced by the neighbours of x_i can be covered by k cliques. The local profit of an optimal algorithm which is achieved on x_i and all its neighbours is therefore bounded by k . Summation over all steps of A_g shows that k is also a bound for the performance ratio. These discussions are summarized in the following theorem.

THEOREM 7.3.

$$R_{\text{MIN}}(\text{BDIS}(k)) \leq k - 1,$$

$$R_{\text{MIN}}(\text{BSSP}(k)) \leq k.$$

8. Final remarks and open problems. Figure 3 contains a list of the optimization problems examined in this paper and gives a survey on the (weak, approximate) continuous reductions among them. Note that Corollary 7.2 implies the weak approximate equivalence of all problems inside the right box. The notion of continuous reductions can be extended into several directions:

- The notion becomes appropriate for parallel (or realtime) approximation algorithms if (in the definition of continuous reductions) it is assumed that $\omega(I)$ and $\beta_I(\sigma')$ can be found in logspace (or in linear time).

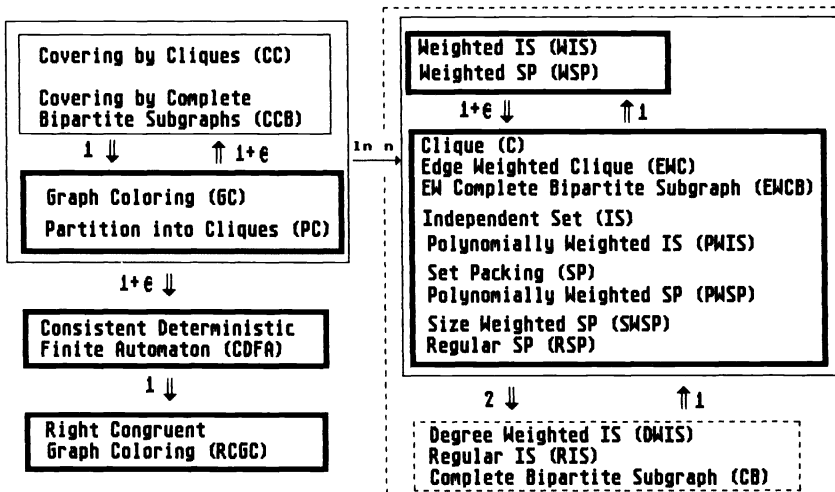


FIG. 3. A collection of continuous reductions among combinatorial optimization problems. The boxes with thick (thin, dashed) borderlines contain equivalent (approximate equivalent, weak approximate equivalent) problems.

- Continuous reductions ω from Π to Π' with a bounded expansion r are not only meaningful for bounded performance ratios but also for performance ratio functions. The *blow-up* of ω is defined as

$$b_\omega(n) = \max_{I \in \text{INPUT}(\Pi), |I| \leq n} \{|\omega(I)|\}.$$

If A' is an approximation algorithm for Π' and A is the corresponding approximation algorithm for Π (compare with Fig. 1 in § 3), then

$$R_A(n) \leq r \cdot R_{A'}(b_\omega(n)).$$

For instance, $R_A(n) = O(R_{A'}(n))$ if $R_{A'}(n)$ is polylogarithmic (or alternatively if $R_{A'}(n)$ is polynomial and the blow-up $b_\omega(n)$ is a linear function). We may also allow expansion functions $r(n)$ instead of bounded expansions r .

The examination of these extensions as well as the following open problems are left as objects of future research:

- The sample of words used in the continuous reduction from GC to CDFA is extremely sparse: it contains $O(k \cdot (n^2 \cdot m))$ words and the longest word has a length of $n + m + 2k - 2$. This does not rule out the possibility of polynomial time algorithms for dense samples of words. For instance, if all words up to a given maximal length are included in the sample, an algorithm of Trakhtenbrot and Barzdin [28] computes a minimum consistent deterministic finite automaton in polynomial running time. Does this result generalize to arbitrary dense samples if we define the notion of “density” properly?
- We describe a simple noncontinuous reduction from RCGC to GC in the appendix. Can we make this reduction continuous? If the answer is “yes,” we may apply this reduction together with $\text{CDFA} \Rightarrow \text{RCGC}$ and the recent result about the nonapproximability of CDFA (see [22]), and conclude that the best asymptotic performance ratio for GC is infinite (unless $P = \text{NP}$). This would explain why the problem of graph coloring has resisted all hitherto existing attempts to find approximate solutions efficiently.
- Can we rule out an approximation algorithm for CDFA with a sublinear performance ratio function (unless $P = \text{NP}$)? There is some evidence against such algorithms because their existence would have the following implications on learning algorithms:
 - Following [5] and [23], DFA’s would be *polynomially predictable*.⁸ A recent result of Kearns and Valiant [14] shows, however, that DFA’s are not polynomially predictable if any of several well established cryptographic assumptions are valid (for instance, hardness of factoring or quadratic residuosity).
 - Following [23] again, a polynomial prediction algorithm for DFA’s is convertible into a polynomial algorithm for any prediction problem whose corresponding *evaluation problem*⁹ is solvable in logspace. In particular, Boolean formulae would be polynomially predictable. For details, see [5], [23], [22].

⁸ A prediction algorithm for DFA’s has as input a sequence of positive and negative examples (drawn according to an unknown probability distribution) of an unknown regular language and outputs a prediction for each new word (drawn according to the same probability distribution) such that the probability of an error becomes arbitrarily small.

⁹ The evaluation problem corresponding to the prediction of DFA’s, for instance, is the problem of deciding whether a given DFA accepts a given word.

- Are there approximation algorithms for the problem covering by edge-disjoint complete bipartite subgraphs (CEDCB) which is defined like CCB except that the complete bipartite subgraphs in the cover must not have any edge in common? The motivation of this and the CCB problem comes from the following relation to distributed computations of Boolean functions¹⁰

$$f: \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}.$$

If the computation of $z = f(x, y)$ is performed by two processors P (with input $x \in \{0, 1\}^n$) and Q (with input $y \in \{0, 1\}^n$), one may ask how many bits of information must be exchanged for determining $z \in \{0, 1\}$ properly. The answer to this question also depends on the communication protocol which may be deterministic, nondeterministic, or probabilistic. The minimum number of bits which must be exchanged in the worstcase is called the (deterministic, nondeterministic, probabilistic) communication complexity of f . It turns out that the nondeterministic communication complexity (modulo a bounded factor) of a given Boolean function is computable in polynomial time if and only if CBB has an approximation algorithm with a bounded performance ratio. A similar (but slightly weaker) relationship is valid for CEDCB and the deterministic communication complexity. For a more detailed description see [1], [29]. Readers interested in the relationship between communication complexity and lower bounds for VLSI-computations are referred to [18], [19], [12].

- The greedy algorithm of § 7 for BDIS(k) and BSSP(k) is quite simple. Are there approximation algorithms with smaller performance ratios? A result in [21] gives, at least, evidence against approximation schemes. The authors show that a scheme for BDIS or BSSP implies schemes for a broad class (called MaxSNP) of maximization problems. No member of MaxSNP is known to have a scheme as yet.

Appendix. The phenomenon of noncontinuous reductions. We now continue in the discussion from § 3 about the phenomenon of noncontinuous reductions. A frequent and instructive violation of the axioms for continuous reductions has the following outline: transform the pair (I, B) , consisting of an input instance of Π and a bound on the values of the solutions, into a corresponding pair (I', B') for Π' such that I' depends on B .

A concrete example for this is the following transformation from RCGC into GC. An input instance $I = (V, E)$ of RCGC is an undirected graph with $V \subset \Sigma^*$ for an alphabet Σ . B is a bound on the number of colors in right congruent colorings of I . The transformation outputs a corresponding pair (I', B') for GC. The graph I' is constructed from (I, B) by the technique of local replacement. For all $a \in \Sigma$, $x, y, xa, ya \in V$, several additional vertices and edges are inserted according to Fig. 4. Part (a) shows the replacement for $B = 4$ and part (b) indicates the general rule: the additional vertices are x', y' and the vertices of the two cliques $I_1 = I_1(x, y, a, B)$ and $I_2 = I_2(x, y, a, B)$ each of size $B - 2$. All the vertices of I_1 are connected with x' and y' , (roughly) half of them are connected with x and the remaining ones with y . I_2 has the analogous connections to xa, ya, x', y' .

The following result clears up the purpose of this construction.

LEMMA A.1. *For all $B \geq 6$:*

1. *Each coloring of I' with B colors induces a right congruent coloring for the subgraph I .*
2. *Each right congruent coloring of I , using at most B colors, can be extended to a coloring of I' using B colors.*

¹⁰ Slight modifications of the following remarks are valid for arbitrary discrete functions.

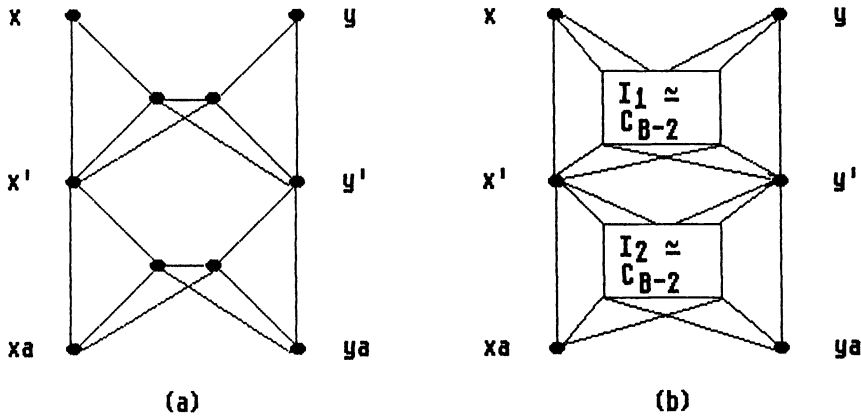


FIG. 4. (a) the local replacement for $B = 4$, (b) the local replacement for arbitrary values of B .

Proof. 1) If x and y have the same color F , this color is forbidden for any vertex of the clique $I_1 \cong C_{B-2}$. Therefore, a unique color for x' and y' is left over. Thus, x' and y' , and with the same argument, xa and ya , must have the same color, respectively.

2) For any right congruent coloring of I using at most B colors, if x and y have the same color F then xa and ya have the same color F'' for some F'' . We choose $F' \notin \{F, F''\}$, color x' and y' with F' and the 2 cliques I_1 and I_2 with the respectively remaining $B - 2$ colors. If x and y have different colors, these colors can be reused in I_1 such that two colors different from $f(xa)$ and $f(ya)$ are left to x' and y' , respectively. If we color x' and y' with these colors and reuse them in I_2 , it is possible to leave $f(xa)$ and $f(ya)$ to xa and ya , respectively. This completes the proof of the lemma. $\square$

This lemma establishes an ordinary reduction from the decision problem for RCGC to the decision problem for GC. An optimal algorithm for GC is convertible into an optimal algorithm for RCGC by applying an obvious binary search on the optimal value for B .

It is worthwhile understanding why this procedure fails for approximation algorithms A' . The reason is the apparent dependence of $I' = I'(B)$ from B . In particular, $\text{opt}(I'(B)) \geq B$. Thus, A' may fail to produce a coloring with B colors for $I'(B)$ even if B is much greater than $\text{opt}(I)$. It is therefore possible that A' produces colorings for $I'(B)$ whose restrictions to I are not right congruent at all. These and other difficulties suggested the definition of continuous reductions chosen in § 3.

Acknowledgments. We thank Mark Krentel [16] who communicated to us two nice mutual reductions between IS and SWSP. We adopted one of his ideas in our reduction from SP to RSP. We are grateful to Leonard Pitt and Manfred Warmuth for directing our attention to the problem of consistent deterministic finite automata, and to Joachim Hartmann for helpful discussions.

REFERENCES

- [1] H. ABELSON, *Lower bounds on information transfer in distributed computations*, in Proc. of the 10th ACM Symp. on Theory of Computing, Association for Computing Machinery, May 1978, pp. 151–158.
- [2] G. AUSIELLO, A. D'ATRI, AND M. PROTASI, *On the structure of combinatorial problems and structure preserving reductions*, in Proceedings of the 4th International Colloquium on Automata, Languages and Programming, 1977, pp. 45–57.
- [3] ———, *Structure preserving reductions among convex optimization problems*, J. Comput. System Sci., 21 (1980), pp. 136–153.

- [4] P. BERMAN AND G. SCHNITTGER, *On the complexity of approximating the independent set problem*, in Proc. of the Annual Symposium on Theoretical Aspects of Computer Science, B. Monien and R. Cori, eds., Springer Verlag, Berlin, February 1989, pp. 256–268.
- [5] A. BLUMER, A. EHRENFUCHT, D. HAUSSLER, AND M. WARMUTH, *Occam's razor*, Inform. Process. Lett., 24 (1987), pp. 377–380.
- [6] P. CRESCENZI AND A. PANCONESI, *Completeness in approximation classes*, Research Report, Department of Mathematics, University of Rome, 1988; Proc. of the 17th Conf. on Fundamentals of Computation Theory, August 1989, to appear.
- [7] M. R. GAREY AND D. S. JOHNSON, *The complexity of near-optimal graph coloring*, J. Assoc. Comput. Mach., 23 (1976), pp. 43–49.
- [8] M. R. GAREY, D. S. JOHNSON, AND L. STOCKMEYER, *Some simplified NP-complete graph problems*, Theoret. Comput. Sci., 1 (1976), pp. 237–267.
- [9] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, 1979.
- [10] F. GAVRIL, *unpublished manuscript*, 1974.
- [11] E. M. GOLD, *Complexity of automaton identification from given data*, Inform. and Control, 37 (1978), pp. 302–320.
- [12] J. JAJA AND P. KUMAR, *Information transfer in distributed computing with application to VLSI*, J. Assoc. Comput. Mach., 31 (1984), pp. 150–162.
- [13] D. S. JOHNSON, *Approximation algorithms for combinatorial problems*, J. Comput. System Sci., 9 (1974), pp. 256–278.
- [14] M. KEARNS AND L. G. VALIANT, *Learning Boolean formulae or finite automata is as hard as factoring*, Technical Report TR-14-88, Aiken Computation Laboratory, Harvard University, November 1988.
- [15] L. T. KOU, L. J. STOCKMEYER, AND C. K. WONG, *Covering edges by cliques with regard to keyword conflicts and intersection graphs*, Comm. ACM, 21 (1978), pp. 135–139.
- [16] M. KRENTEL, *personal communication*, 1989.
- [17] M. LI AND U. VAZIRANI, *On the learnability of finite automata*, in Proceedings of the 1st Workshop on Computational Learning Theory, M. Kaufmann, ed., San Mateo, California, August 1988.
- [18] R. LIPTON AND R. SEDGEWICK, *Lower bounds for VLSI computations*, in Proc. of the 13th ACM Symp. on Theory of Computing, Association for Computing Machinery, May 1981, pp. 300–307.
- [19] K. MEHLHORN AND F. PREPARATA, *AT²-optimal VLSI integer multiplication with minimum computation time*, Inform. and Control, 58 (1983), pp. 137–156.
- [20] P. ORPONEN AND H. MANNILA, *On approximation preserving reductions: complete problems and robust measures*, Technical Report, University of Helsinki, 1987.
- [21] C. H. PAPADIMITRIOU AND M. YANNAKAKIS, *Optimization, approximation, and complexity classes*, in Proc. of the 20th ACM Symp. on Theory of Computing, Association for Computing Machinery, May 1988, pp. 229–234.
- [22] L. PITT AND M. K. WARMUTH, *The minimum consistent DFA problem cannot be approximated within any polynomial*, Research Report UIUCDCS-R-89-1499, Department of Computer Science, University of Illinois at Urbana–Champaign, February 1989; Proc. of 4th Conf. on Structure in Complexity Theory, June, 1989, to appear.
- [23] ———, *Reductions among prediction problems: on the difficulty of predicting automata*, in Proc. of the 3rd IEEE Conf. on Structure in Complexity Theory, IEEE, Washington DC, June, 1988, pp. 60–69.
- [24] H. U. SIMON, *The analysis of dynamic and hybrid channel assignment*, Research Report SFB 124-B1 10, Fachbereich Informatik, Universität des Saarlandes, D-6600 Saarbrücken, FRG, 1988; Proc. of the 17th Conf. on Fundamentals of Computation Theory, August 1989, to appear.
- [25] ———, *Continuous reductions among combinatorial optimization problems*, Research Report, Siemens AG, Zentralbereich Technik, Zentrale Aufgaben Informationstechnik, 1986.
- [26] ———, *Continuous reductions among combinatorial optimization problems*, Research Report SFB 124-B1 5, Fachbereich Informatik, Universität des Saarlandes, D-6600 Saarbrücken, FRG, 1988; Acta Inform., to appear.
- [27] H. U. SIMON, *On approximate solutions for combinatorial optimization problems*, Research Report SFB 124-B1 15, Fachbereich Informatik, Universität des Saarlandes, D-6600 Saarbrücken, FRG, 1988. A preliminary version has been presented at the 4th SIAM Conf. on Discrete Mathematics in San Francisco.
- [28] B. A. TRAKHTENBROT AND Y. M. BARZDIN, *Finite Automata*, North-Holland, Amsterdam, 1973, pp. 98–99.
- [29] A. YAO, *Some complexity questions related to distributive computing*, in Proc. of the 11th ACM Symp. on Theory of Computing, Association for Computing Machinery, May 1979, pp. 209–213.

RANDOM PATHS AND CUTS, ELECTRICAL NETWORKS, AND REVERSIBLE MARKOV CHAINS*

KENNETH A. BERMAN† AND MOKHTAR H. KONSOWA‡

Abstract. Let G be a graph representing an electrical network having source a and sink b , where edge e has conductivity $C(e)$ and resistance $R(e)$ ($R(e) = 1/C(e)$). Let C_{eff} and R_{eff} denote the effective conductivity and effective resistance of G , respectively, ($R_{\text{eff}} = 1/C_{\text{eff}}$). Let $\mathcal{P}$ denote the set of all paths joining a and b , and $\mathcal{K}$ denote the set of all cuts separating a and b . Let P be a random path from $\mathcal{P}$ having probability function p , and K be a random cut from $\mathcal{K}$ having probability function q . Denote by $\pi(e)$ ($\kappa(e)$) the expected number of paths (cuts) containing edge e . For w a weighting defined on E and H a subset of E (or subgraph), let $|H|_w = \sum_{e \in H} w(e)$. In this paper it is shown that C_{eff} equals the maximum expected value of $1/|P|_{R_{\pi}}$ over all probability functions p . Further, the maximum is achieved when p is chosen so that π is the current weighting. An immediate corollary of this result is Thomson's Principle which characterizes the effective resistance as the minimum energy dissipation among all unit flows through the network. It is also shown that R_{eff} equals the maximum expected value of $1/|K|_{C_{\kappa}}$ over all probability functions q . Furthermore, the maximum is achieved when q is chosen so that κ is the voltage weighting. This result yields Nash–Williams' condition [5] for recurrence of reversible Markov chains.

Key words. random paths, random cuts, electrical networks, reversible Markov chains

AMS(MOS) subject classifications. 05C, 05C38

1. Introduction. Let G be a graph with vertex set V and edge set E representing an electrical network (where all edges are resistors). In this paper we allow G to have multiple edges. Let a and b be the source and sink, respectively, (i.e., current flow enters the network at a and leaves at b). Let ι_a denote the current entering at a , and let ν_a denote the voltage at a . We will assume that the voltage at b is zero, i.e., $\nu_b = 0$. Consider an edge e joining two vertices u and v , where the current flows from u to v . Let $R(e)$, $C(e)$ denote the resistance, and conductivity of edge e ($R(e) = 1/C(e)$). Let $\iota(e)$ denote the current flowing through e , and let $\nu(e)$ denote the potential difference between u and v , i.e., $\nu(e) = \nu_u - \nu_v$. The *effective resistance* is given by

$$(1.1) \quad R_{\text{eff}} = \frac{\nu_a}{\iota_a}.$$

The *effective conductivity* is given by $C_{\text{eff}} = \iota_a/\nu_a = 1/R_{\text{eff}}$.

A *weighting* w on E is a mapping from E to the real numbers. For H a subset of E (or subgraph) let

$$(1.2) \quad |H|_w = \sum_{e \in H} w(e).$$

Note that $|H|_w$ is the cardinality of H when $w \equiv 1$. The *characteristic* edge weighting of H is given by

$$(1.3) \quad \chi_H(e) = \begin{cases} 1 & \text{if } e \text{ is an edge of } H \\ 0 & \text{otherwise.} \end{cases}$$

An *ab-path* is a path joining a and b . Let $\mathcal{P}$ denote the set of all *ab*-paths. Let p be a probability function on the set $\mathcal{P}$, and let P be a random path from $\mathcal{P}$ with probability

* Received by the editors January 25, 1989; accepted for publication August 15, 1989.

† Department of Computer Science, University of Cincinnati, Cincinnati, Ohio 45221.

‡ Department of Mathematics, University of Cincinnati, Cincinnati, Ohio 45221.

function p . For ϕ a mapping from $\mathcal{P}$ to the set of real numbers (set of edge weightings), we denote the expected value (expected edge weighting) of $\phi(P)$ by $E[\phi(P)]$, i.e.,

$$(1.4) \quad E[\phi(P)] = \sum_{P \in \mathcal{P}} p(P) \phi(P).$$

Let $\pi(e)$ denote the expected number of paths containing edge e , i.e.,

$$(1.5) \quad \pi = E[\chi_P].$$

An *ab-cut* is a cut which separates a and b , i.e., a set K of edges for which there exists a bipartition of the vertex set V into subsets U and W , $a \in U$ and $b \in W$, such that K consists of all the edges having one end in U and the other in W . Let $\mathcal{K}$ denote the set of all *ab-cuts*. Let q be probability function on $\mathcal{K}$, and let K be a random cut from $\mathcal{K}$ with probability function q . Let $\kappa(e)$ denote the expected number of cuts containing edge e , i.e.,

$$(1.6) \quad \kappa = E[\chi_K].$$

In § 2, we show that there exists a probability function p on $\mathcal{P}$ such that $\pi = \iota/\iota_a$, and a probability function q on $\mathcal{K}$ such that $\kappa = \nu/\nu_a$.

In § 3, we show that C_{eff} equals the maximum value of $E[1/|P|_{R\pi}]$ over all probability functions p on $\mathcal{P}$. Furthermore, the maximum is achieved when p is chosen such that $\pi = \iota/\iota_a$. We also show that R_{eff} equals the maximum value of $1/|K|_{C\kappa}$ over all probability functions q on $\mathcal{K}$. Furthermore, the maximum is achieved when q is chosen such that $\kappa = \nu/\nu_a$.

In § 4, we show how the above formula for C_{eff} yields Thomson's Principle which characterizes the effective resistance as the minimum energy dissipation among all unit flows through the network.

There is a natural connection between electrical networks and reversible Markov chains (see [3], [5]). It is well known that a reversible Markov chain is recurrent if and only if the corresponding electrical network has infinite effective resistance. In § 5, we show that the formula (given in § 3) for R_{eff} implies Nash-Williams' condition ([3], [5]) for recurrence of reversible Markov chains.

2. Flows and coboundaries. Consider an orientation of the edges of G . An *ab-flow* or simply a *flow* is an edge weighting f such that the sum of the f -weights over all the edges having head v equals the sum of the f -weights over all the edges having tail v for all vertices v different from a and b . Let f_a denote the sum of the f -weights over all the edges having tail a . If $f_a = 1$ then f is a *unit flow*. If the f -weight of every edge is nonnegative then f is a *nonnegative flow*.

With each vertex v associate a nonnegative real number g_v such that $g_b = 0$. A *coboundary* is an edge weighting g for which there exists a weighting g_v of the vertices such that $g(e) = g_t - g_h$ where t is the tail of edge e and h is the head of edge e . If $g_a = 1$ then g is a *unit coboundary*. If the g -weight of every edge is nonnegative then g is a *nonnegative coboundary*.

The following are the two classical laws of electricity due to Kirchhoff ([2], [3]). Orient the edges of G in the direction of the current flow.

Current Law. The current ι is an *ab-flow*.

Voltage Law. The voltage ν is a coboundary.

A third well known law is due to Ohm.

Ohm's Law. $R = \nu/\iota$.

We remind the reader that (for a given current ι_a entering at the source a) the current and voltage are the unique solutions to the three laws given above (see [2]).

A *directed ab-path* is an *ab-path* such that every edge of the path is directed toward b and away from a . A *directed ab-cut* is an *ab-cut* such that every edge of the cut has tail in U and head in W , where U, W bipartition V , and $a \in U, b \in W$. Let $\vec{\mathcal{P}}$ denote the set of all *directed ab-paths*, and $\vec{\mathcal{K}}$ the set of *directed ab-cuts*. The following two propositions are well known and easily proved.

PROPOSITION 2.1. *Let f be any ab -flow and let K be any directed ab -cut. Then*

$$(2.1) \quad |K|_f = f_a.$$

PROPOSITION 2.2. *Let g be any coboundary and let P be any directed ab -path. Then*

$$(2.2) \quad |P|_g = g_a.$$

PROPOSITION 2.3. *Let f be any nonnegative ab -flow. Then there exists a flow ϕ and nonnegative scalars λ_P such that $\phi_a = 0$ and*

$$(2.3) \quad f = \phi + \sum_{P \in \vec{\mathcal{P}}} \lambda_P \chi_P.$$

Proof. Let $\theta(f)$ denote the number of edges whose f -weights are nonzero. We will prove Proposition 2.3 by induction on $\theta(f)$. If $\theta(f) = 0$ then $f \equiv 0$ and the proposition is true. Assume the proposition is true for all f such that $\theta(f) \leq k$. Now consider any nonnegative flow f such that $\theta(f) = k + 1$. If $f_a = 0$ then the proposition is trivially true. Thus we may assume that $f_a > 0$. Let Q be the longest directed path having initial vertex a such that f -weight of every edge of Q is nonzero. We will show that the terminal vertex t of Q must be vertex b . Assume the contrary. Let e be the edge of Q having head t . Since the f -weight of e is nonzero and since f is an ab -flow it follows that there exists an edge e' having tail t whose f -weight is nonzero. But by adding edge e' to path Q we obtain a longer path. This is a contradiction. Hence the terminal vertex of Q is b , i.e., Q is a directed ab -path. Let μ denote the minimum f -weight among all the edges of Q . Let $f' = f - \mu \chi_Q$. Then f' is a nonnegative flow such that $\theta(f') \leq k$. Thus by induction assumption there exist a nonnegative flow ϕ and scalars λ'_P such that

$$f' = \phi + \sum_{P \in \vec{\mathcal{P}}} \lambda'_P \chi_P.$$

Let $\lambda_P = \lambda'_P$ for every directed ab -path P different from Q and let $\lambda_Q = \lambda'_Q + \mu$. Then we have that

$$f = f' + \mu \chi_Q = \phi + \sum_{P \in \vec{\mathcal{P}}} \lambda_P \chi_P. \quad \square$$

COROLLARY 2.4. *There exists a probability function p on the set $\mathcal{P}$ of ab -paths such that for every edge e , $\iota(e)/\iota_a$ is the expected number of paths which pass through edge e , i.e., $\pi = \iota/\iota_a$.*

Proof. The orientation of G determined by the current flow is *acyclic*, i.e., there are no directed circuits. It is immediate that the only flow ϕ such that $\phi_a = 0$ is the zero flow, i.e., $\phi(e) = 0$ for all $e \in E$. Thus by Proposition 2.3 we have that

$$(2.4) \quad \iota/\iota_a = \sum_{P \in \vec{\mathcal{P}}} \lambda_P \chi_P.$$

Let p be the probability function on the set $\mathcal{P}$ of all ab -paths given by

$$(2.5) \quad p(P) = \begin{cases} \lambda_P & \text{for } P \in \vec{\mathcal{P}} \\ 0 & \text{otherwise.} \end{cases}$$

For this choice of probability function it is immediate that $\iota(e)/\iota_a$ is the expected number of ab -paths that pass through edge e , $e \in E$. $\square$

PROPOSITION 2.5. *Let g be any nonnegative coboundary. Then there exists a coboundary γ and nonnegative scalars λ_K such that $\gamma_a = 0$ and*

$$(2.6) \quad g = \gamma + \sum_{K \in \vec{\mathcal{K}}} \lambda_K \chi_K.$$

Proof. Let $\theta(g)$ denote the number of edges whose g -weight is nonzero. We will prove Proposition 2.5 by induction on $\theta(g)$. If $\theta(g) = 0$ then $g \equiv 0$ and the proposition is true. Assume the proposition is true for all g such that $\theta(g) \leq k$. Now consider any nonnegative coboundary g such that $\theta(g) = k + 1$. If $g_a = 0$ then the proposition is trivially true. Thus we may assume that $g_a > 0$. Let $U = \{v \in V \mid g_v \geq g_a\}$, and let $W = V - U$. Since $g_b = 0$, $b \in W$. Thus the set L of edges having one end in U and the other in W is an ab -cut. Since g is nonnegative it follows that the tail of every edge of L lies in U , i.e., L is a directed ab -cut. Further, since the g -weight of any vertex in W is strictly less than the g -weight of any vertex in U , the g -weights of all the edges in L are nonzero. Let μ denote the minimum g -weight among all the edges in L . Let $g' = g - \mu \chi_L$. Then g' is a nonnegative coboundary such that $\theta(g') \leq k$. Thus by induction assumption there exist a nonnegative coboundary γ and nonnegative scalars λ'_K such that

$$g' = \gamma + \sum_{K \in \vec{\mathcal{K}}} \lambda'_K \chi_K.$$

Let $\lambda_K = \lambda'_K$ for every directed ab -cut K different from L and let $\lambda_L = \lambda'_L + \mu$. Then we have that

$$g = g' + \mu \chi_L = \gamma + \sum_{K \in \vec{\mathcal{K}}} \lambda_K \chi_K. \quad \square$$

COROLLARY 2.6. *There exists a probability function q on the set $\mathcal{K}$ of ab -cuts such that for every edge e , $\nu(e)/\nu_a$ is the expected number of ab -cuts which contain edge e , i.e., $\kappa = \nu/\nu_a$.*

Proof. Consider the orientation of G determined by the current flow. Then every vertex v different from a and b is incident with at least one edge having tail v and at least one edge having head v . Thus the only coboundary γ such that $\gamma_a = 0$ is the zero coboundary, i.e., $\gamma(e) = 0$ for all $e \in E$. Thus by Proposition 2.5 we have that

$$(2.7) \quad \nu/\nu_a = \sum_{K \in \vec{\mathcal{K}}} \lambda_K \chi_K.$$

Let q be the probability function on the set $\mathcal{K}$ of all ab -cuts given by

$$(2.8) \quad q(K) = \begin{cases} \lambda_K & \text{for } K \in \vec{\mathcal{K}} \\ 0 & \text{otherwise.} \end{cases}$$

For this choice of probability function it is immediate that $\nu(e)/\nu_a$ is the expected number of ab -cuts that contain edge e , $e \in E$. $\square$

3. Main Theorem.

THEOREM 3.1. *Let G be a connected graph representing an electrical network having source a and sink b . Let P be a random ab -path having probability function p , and let K be a random ab -cut having probability function q . For e an edge of G let $\pi(e)$ denote the expected number of paths containing edge e , and $\kappa(e)$ the expected number of cuts containing edge e . Then the effective conductivity equals the maximum expected value of $1/$*

$|P|_{R\pi}$ over all probability functions p , and the effective resistance equals the maximum expected value of $1/|K|_{C\kappa}$ over all probability functions q , i.e.,

$$(3.1) \quad C_{\text{eff}} = \max_p \{E[1/|P|_{R\pi}]\},$$

$$(3.2) \quad R_{\text{eff}} = \max_q \{E[1/|K|_{C\kappa}]\}.$$

Furthermore, the maximum is achieved in (3.1) if p is chosen so that $\pi = \iota/\iota_a$, and the maximum is achieved in (3.2) if q is chosen so that $\kappa = \nu/\nu_a$.

Proof. We prove Theorem 3.1 with the aid of the Cauchy–Schwartz inequality and the following well-known inequality. For any nonnegative real numbers $x_1, x_2, \dots, x_n$ and $y_1, y_2, \dots, y_n$,

$$(3.3) \quad \left(\sum_{i=1}^n x_i y_i\right)^2 \leq \left(\sum_{i=1}^n x_i\right) \left(\sum_{i=1}^n x_i y_i^2\right).$$

Define the dot product of two edge weightings α and β by

$$(3.4) \quad \alpha \cdot \beta = \sum_{e \in E} \alpha(e) \beta(e).$$

Let

$$(3.5) \quad w = \sum_{K \in \mathcal{K}} q(K) \chi_K / |K|_{C\kappa}.$$

By the Cauchy–Schwartz inequality [1] we have

$$(3.6) \quad (\sqrt{R}\pi \cdot \sqrt{R}\pi)(\sqrt{C}w \cdot \sqrt{C}w) \geq (\sqrt{R}\pi \cdot \sqrt{C}w)^2.$$

It follows from the definition of π that

$$\sqrt{R}\pi \cdot \sqrt{R}\pi = E[\chi_P] \cdot R\pi = \sum_{P \in \mathcal{P}} p(P)(\chi_P \cdot R\pi) = \sum_{P \in \mathcal{P}} p(P)|P|_{R\pi}.$$

Hence

$$(3.7) \quad E[|P|_{R\pi}] = \sqrt{R}\pi \cdot \sqrt{R}\pi.$$

It follows from (3.5) that

$$\sqrt{R}\pi \cdot \sqrt{C}w = \left(\sum_{P \in \mathcal{P}} p(P)\sqrt{R}\chi_P\right) \cdot \left(\sum_{K \in \mathcal{K}} q(K)\sqrt{C}\chi_K / |K|_{C\kappa}\right).$$

Now every ab -path has at least one edge in common with every ab -cut. Also $R = 1/C$. It follows that $\sqrt{R}\chi_P \cdot \sqrt{C}\chi_K \geq 1$ for every $P \in \mathcal{P}$ and $K \in \mathcal{K}$. Hence

$$(3.8) \quad E[1/|K|_{C\kappa}] \leq \sqrt{R}\pi \cdot \sqrt{C}w.$$

It follows from the definition of the dot product that

$$\sqrt{C}w \cdot \sqrt{C}w = \sum_{e \in E} C(e) \left(\sum_{K: e \in K} q(K) / |K|_{C\kappa}\right)^2.$$

Employing inequality (3.3) we have that

$$\begin{aligned}\sqrt{C_W} \cdot \sqrt{C_W} &\leq \sum_{e \in E} C(e) \left(\sum_{K: e \in K} q(K) \right) \left(\sum_{K: e \in K} q(K) / |K|_{C_K}^2 \right) \\ &= \sum_{e \in E} C(e) \kappa(e) \left(\sum_{K: e \in K} q(K) / |K|_{C_K}^2 \right) = \sum_{K \in \mathcal{K}} |K|_{C_K} (q(K) / |K|_{C_K}^2).\end{aligned}$$

Hence

$$(3.9) \quad \sqrt{C_W} \cdot \sqrt{C_W} \leq E[1 / |K|_{C_K}].$$

Combining (3.6), (3.7), (3.8), and (3.9) we obtain

$$(3.10) \quad E[1 / |K|_{C_K}] \leq E[|P|_{R\pi}].$$

Let $\pi^* = \iota / \iota_a$, and $\kappa^* = \nu / \nu_a$. Let p be the probability function on $\mathcal{P}$ given by (2.5), i.e., $\pi = \pi^*$, and q be the probability function on $\mathcal{K}$ given by (2.8), i.e., $\kappa = \kappa^*$. Let $f = \iota / \nu_a$ and let $g = \nu / \iota_a$. It follows from the definitions of C_{eff} and R_{eff} that

$$f_a = \iota_a / \nu_a = C_{\text{eff}},$$

$$g_a = \nu_a / \iota_a = R_{\text{eff}}.$$

It follows from Ohm's Law that $f = C_{\kappa^*}$ and $g = R_{\pi^*}$. Since f is a flow, Proposition 2.1 implies that $|K|_f = f_a$ for every directed ab -cut K , i.e., $|K|_{C_{\kappa^*}} = C_{\text{eff}}$ for each $K \in \mathcal{K}$. Since (by definition) $q(K) = 0$ for every ab -cut K which is not a directed ab -cut we have that

$$(3.11) \quad E[1 / |K|_{C_{\kappa^*}}] = 1 / C_{\text{eff}} = R_{\text{eff}}.$$

Since g is a coboundary, Proposition 2.2 implies that $|P|_g = g_a$ for every directed ab -path P , i.e., $|P|_{R_{\pi^*}} = R_{\text{eff}}$ for each $P \in \mathcal{P}$. Since (by definition) $p(P) = 0$ for every ab -path P which is not a directed ab -path we have that

$$(3.12) \quad E[|P|_{R_{\pi^*}}] = R_{\text{eff}}.$$

It follows from (3.10), (3.11), and (3.12) that

$$(3.13) \quad E[1 / |K|_{C_K}] \leq R_{\text{eff}} \leq E[|P|_{R\pi}]$$

and equality holds throughout (3.13) if $\pi = \iota / \iota_a$ and $\kappa = \nu / \nu_a$.

By analogous arguments to the ones given above we have that

$$(3.14) \quad E[1 / |P|_{R\pi}] \leq C_{\text{eff}} \leq E[|K|_{C_K}]$$

and equality holds throughout (3.14) if $\pi = \iota / \iota_a$ and $\kappa = \nu / \nu_a$.

Theorem 3.1 follows immediately from (3.13) and (3.14). $\square$

4. Path and cut intersection numbers. Let p be a probability function on $\mathcal{P}$, and let P and Q be independent $\mathcal{P}$ -valued random variables, each having probability function p . The *expected path intersection number* $\eta_R(p)$ is the expected value of $|P \cap Q|_R$, i.e.,

$$(4.1) \quad \eta_R(p) = \sum_{P \in \mathcal{P}} \sum_{Q \in \mathcal{P}} p(P)p(Q) |P \cap Q|_R.$$

The *expected cut intersection number* $\gamma_c(q)$ is defined analogously.

THEOREM 4.1. *The effective resistance equals the minimum expected path intersection number over all probability functions p on the set $\mathcal{P}$ of ab -paths, and the effective*

conductivity equals the minimum expected cut intersection number over all probability functions q on the set $\mathcal{H}$ of ab-cuts, i.e.,

$$(4.2) \quad R_{\text{eff}} = \min_p \{ \eta_R(p) \},$$

$$(4.3) \quad C_{\text{eff}} = \min_q \{ \gamma_c(q) \}.$$

Furthermore, the minimum is achieved in (4.2) if p is chosen so that $\pi = \iota/\iota_a$, and the minimum is achieved in (4.3) if q is chosen so that $\kappa = \nu/\nu_a$.

Proof. It is easily verified that

$$(4.4) \quad \eta_R = E[|P|_{R\pi}],$$

$$(4.5) \quad \gamma_C = E[|K|_{C\kappa}].$$

It follows from Markov's inequality [1] (where the convex function is $1/x$) that

$$(4.6) \quad 1/\eta_R \leq E[1/|P|_{R\pi}],$$

$$(4.7) \quad 1/\gamma_C \leq E[1/|K|_{C\kappa}].$$

Theorem 4.1 now follows immediately from Theorem 3.1. $\square$

The (Euclidean) norm of an edge weighting w is defined by

$$\|w\| = \left(\sum_{e \in E(G)} w^2(e) \right)^{1/2}.$$

The energy dissipation of a unit flow f of G is given by $\|\sqrt{R}f\|^2$. The following corollary of Theorem 4.1 is known as Thomson's Principle (see [3]).

COROLLARY 4.2 (Thomson's Principle). *The effective resistance is the minimum energy dissipation among all unit flows f , i.e.,*

$$(4.8) \quad R_{\text{eff}} = \min_f \{ \|\sqrt{R}f\|^2 \}.$$

Proof. Let f be any unit flow (negative weights allowed). If some edge e has negative f -weight, then replace the weight of e with $-f(e)$ and reverse the orientation of edge e . The resulting edge weighting is a flow with respect to the new orientation. Let f^+ be the nonnegative flow obtained from f by reversing the orientation and changing the sign of all edges whose f -weight is negative. By Proposition 4.1 $f^+ = \phi + \hat{f}$ where ϕ is a nonnegative flow such that $\phi_a = 0$ and $\hat{f}$ is a unit flow for which there exists nonnegative scalars λ_P such that

$$\hat{f} = \sum_{P \in \mathcal{P}} \lambda_P \chi_P.$$

Since $\hat{f}$ is a unit flow, the function $p(P) = \lambda_P$ is a probability function on $\mathcal{P}$. Furthermore,

$$(4.9) \quad \|\sqrt{R}\hat{f}\|^2 = \left(\sum_{P \in \mathcal{P}} p(P) \sqrt{R} \chi_P \right) \cdot \left(\sum_{Q \in \mathcal{P}} p(Q) \sqrt{R} \chi_Q \right) = \eta_R(p).$$

Now $\|\sqrt{R}f\| = \|\sqrt{R}f^+\| \geq \|\sqrt{R}\hat{f}\|$. Thus

$$(4.10) \quad \|\sqrt{R}f\|^2 \geq \eta_R(p).$$

If f is chosen to be ι/ι_a and p is chosen such that $\iota(e)$ equals the expected number of paths through edge e , then it follows from Theorem 4.1 and definition (4.1) that $\|\sqrt{R}f\| = \eta_R(p) = R_{\text{eff}}$. Corollary 4.2 follows.

Analogously, we have the “dual” of Thomson’s Principle for the effective conductivity C_{eff} .

COROLLARY 4.3. *The effective conductivity is the minimum of $\|\sqrt{C}g\|^2$ among all unit coboundaries g , i.e.,*

$$(4.11) \quad C_{\text{eff}} = \min_g \{ \|\sqrt{C}g\|^2 \}.$$

5. Reversible Markov chains. A Markov chain X_n ($n = 0, 1, 2, \dots$) with state space V is said to be *reversible* if there exists a positive weighting w on V such that

$$(5.1) \quad w(u)p_{uv} = w(v)p_{vu},$$

where p_{uv} is the transition probability from u to v , i.e., $p_{uv} = \Pr[X_{n+1} = v | X_n = u]$ ($u, v \in V$).

With an electrical network G we may associate a reversible Markov chain X_n as follows (see [3], [5]). Let C_v denote the sum of the C -weights over all the edges incident with v . Let C_{uv} denote the sum of the C -weights over all the edges joining u and v . By convention if u and v are not adjacent then $C_{uv} = 0$. The reversible Markov chain associated with G is defined by

$$(5.2) \quad p_{uv} = C_{uv}/C_u.$$

Every reversible Markov chain can be obtained from some electrical network in this way. If the conductivity of each edge is taken to be 1 then the associated reversible Markov chain is the *simple random walk* on G .

Let $M = M(G)$ denote the mean number of visits to a before visiting b , having started at a . It is well known (see [3]) that

$$(5.3) \quad M = C_a R_{\text{eff}}.$$

The following theorem is an immediate consequence of (5.3) and Theorem 3.1.

THEOREM 5.1. *Let X_n be a reversible Markov chain corresponding to an electrical network G with source a and sink b . Let P be any random ab -path, and let K be any random ab -cut. For e an edge of G let $\pi(e)$ denote the expected number of paths containing edge e , and $\kappa(e)$ the expected number of cuts containing edge e . Then the mean number M of visits to a before visiting b , having started at a , satisfies*

$$(5.4) \quad C_a E[1/|K|_{C_K}] \leq M \leq C_a/E[1/|P|_{R_P}].$$

Furthermore, $M = C_a/E[1/|P|_{R_P}]$ if $\pi = \iota/\iota_a$, and $M = C_a E[1/|K|_{C_K}]$ if $\kappa = \nu/\nu_a$.

COROLLARY 5.2. *Let $K_1, K_2, \dots, K_s$ be any collection of pairwise edge disjoint ab -cuts, and let $P_1, P_2, \dots, P_t$ be any collection of pairwise edge disjoint ab -paths. Then*

$$(5.5) \quad C_a \sum_{i=1}^s 1/|K_i|_C \leq M \leq C_a / \left(\sum_{j=1}^t 1/|P_j|_R \right).$$

Proof. Choose a probability function p on $\mathcal{P}$ such that $p(P) = 1/t$,

$$P \in \{P_1, P_2, \dots, P_t\},$$

and $p(P) = 0$ otherwise. Choose a probability function q on $\mathcal{K}$ such that $q(K) = 1/s$, $K \in \{K_1, K_2, \dots, K_s\}$, and $q(K) = 0$, otherwise. Corollary 5.2 follows immediately from Theorem 5.1 with the observation that $C(e) \geq C(e)\kappa(e)$ and $R(e) \geq R(e)\pi(e)$ for every edge e .

Now consider an *infinite* electrical network G (for a discussion of infinite electrical networks see [3]). We assume that V and E are countably infinite and that G is *locally*

finite, i.e., every vertex is incident with a finite number of edges. In an infinite electrical network the sink b is at ∞ . We may associate with G an *infinite* reversible Markov chain X_n as defined by (5.2). For an infinite Markov chain we define M to be the expected number of visits to a having started at a (since $b = \infty$). We say that X_n is *recurrent* if $M = \infty$. Otherwise X_n is *transient*.

Let $V^{(m)}$ denote the set of vertices whose distance from a is m and let $G^{(m)}$ denote the finite electrical network obtained by deleting all vertices in $\bigcup_{i=m+2}^{\infty} V^{(i)}$ and identifying all the vertices in $V^{(i+1)}$ to form a single vertex b . Then

$$(5.6) \quad R_{\text{eff}}(G) = \lim_{m \rightarrow \infty} R_{\text{eff}}(G_m).$$

Also

$$(5.7) \quad M(G) = \lim_{m \rightarrow \infty} M(G_m).$$

Let $V = X \cup Y$ be a bi-partition of the vertex set V and let K be the cut consisting of all edges with one end in X and the other in Y . Let $\mathcal{K}$ denote the set of all cuts such that $a \in X$ and X is finite. Clearly $\mathcal{K}$ is countable.

The following result of Nash-Williams ([3], [5]) follows immediately from (5.6), (5.7), and Corollary 5.2.

COROLLARY 5.3 (Nash-Williams). *Let $K_1, K_2, \dots$ be any collection of pairwise disjoint cuts from $\mathcal{K}$. If*

$$\sum_{i=1}^{\infty} 1 / |K_i|_C = \infty$$

then X_n is recurrent.

REFERENCES

- [1] P. BILLINGSLEY, *Probability and Measure*, Second edition, John Wiley, New York, 1986.
- [2] W. K. CHEN, *Applied Graph Theory*, North-Holland, New York, 1971.
- [3] P. DOYLE AND J. SNELL, *Random Walks and Electrical Networks*, Carus Mathematical Monographs, Mathematical Society of America, 1984.
- [4] T. LYONS, *A simple criterion for transience of a reversible Markov chain*, Ann. Probability, 11 (1983), pp. 393–402.
- [5] C. ST. J. A. NASH-WILLIAMS, *Random walks and electric currents in networks*, Proc. Cambridge Philos. Soc., 55 (1959), pp. 181–194.

ON THE STRUCTURE OF MINIMUM-WEIGHT k -CONNECTED SPANNING NETWORKS*

DANIEL BIENSTOCK†, ERNEST F. BRICKELL†, AND CLYDE L. MONMA†

Abstract. The problem of finding a minimum-weight k -connected spanning subgraph of a complete graph, assuming that the edge weights satisfy the triangle inequality, is studied. It is shown that the class of minimum-weight k -edge connected spanning subgraphs can be restricted to those subgraphs which, in addition to the connectivity requirements, satisfy the following two conditions:

- (I) Every vertex has degree k or $k + 1$;
- (II) Removing any $1, 2, \dots$, or k edges does not leave the resulting connected components all k -edge connected.

For the k -vertex connected case, the parallel result is obtained with “ k -edge” replaced by “ k -vertex,” with the added technical restriction that $|V| \geq 2k$ for condition (I) to hold. This generalizes recent work of Monma, Munson, and Pulleyblank for the case $k = 2$.

Key words. survivability, graph theory, lifting, connectivity

AMS(MOS) subject classifications. 05C40, 94C15, 90C35

1. Introduction. In the design of communication or transportation networks, it is frequently important to produce networks of low “cost” which are also “survivable.” In many cases the cost arises, to a good degree of approximation, in the form of edge weights that satisfy the “triangle inequality” (defined in precise form below). The overall cost, or weight, of a network is the sum of the individual edge weights. For survivability reasons, the network must satisfy certain connectivity requirements (see [CW], [GM], [MS], [SWK] for more motivation). A typical survivability requirement is that the removal of any $(k - 1)$ or fewer edges (or vertices) leaves the remaining network connected. The following standard definitions are required to make the above statements precise.

A graph or network $G = (V, E)$ is called k -edge connected if the removal of any $(k - 1)$ or fewer edges leaves G connected. If, in addition, the removal of any $(k - 1)$ or fewer vertices leaves the remaining vertices of G connected, then G is called k -vertex connected. We note that the degenerate graph consisting of a single vertex is k -edge and k -vertex connected for all values of k . A variation of Menger’s Theorem states that a nondegenerate graph G is k -edge (respectively, k -vertex) connected if and only if there are k edge (respectively, vertex) disjoint paths between every pair of vertices in G .

Hence we obtain the following problem, k -connected network design with triangle inequality: given a complete graph with edge weights that satisfy the triangle inequality, and an integer k , find a minimum-weight k -edge (or k -vertex) connected spanning subgraph. We remark that for any $k \geq 2$ this problem is NP-Hard, as the Hamiltonian Cycle problem can be reduced to a 2-connected network design problem with triangle inequality. Further, in general there will be a difference between the “edge-connected” and “vertex-connected” versions of this problem.

In the following, the word “spanning” will be omitted, for convenience. A solution will be a k -connected subgraph. An optimal subgraph or solution will be a solution of least total weight.

This paper presents some strong structural properties that optimal subgraphs can be assumed to satisfy. In particular, our results show that there are optimal subgraphs

* Received by the editors March 8, 1988; accepted for publication (in revised form) September 18, 1989.

† Bell Communications Research, Morristown, New Jersey 07960.

that are essentially edge-minimal (least number of edges) from among all k -connected subgraphs, without regard to weight (the number of edges will be within a small constant factor from being best possible). In order to describe the results in detail, we need some terminology and background.

We let V denote a set of *vertices*. A nonnegative symmetric *weight function* $d(\cdot, \cdot)$ is defined on all pairs of vertices so that the triangle inequality holds, i.e., $d(u, v) \geq 0$, $d(u, u) = 0$, $d(u, v) = d(v, u)$, and $d(u, v) + d(v, w) \leq d(u, w)$ for all u, v , and w in V .

Given a connected graph $G = (V, E)$, the *canonical weight function* defined by $G = (V, E)$, is given by $d(u, v) =$ minimum number of edges in a shortest path from u to v in G for all u and v in V . It is clear that this choice of weights satisfies the triangle inequality.

A recent paper [MMP] studies this network design problem in the particular case where $k = 2$. They show that for $k = 2$ the class of minimum-weight k -edge (respectively, k -vertex) connected subgraphs can be restricted to the class of k -edge (respectively, k -vertex) connected subgraphs $G = (V, E)$ satisfying the following conditions:

- (I) Every vertex of G has degree k or $k + 1$;
- (II) Removing any 1, 2, $\dots$, or k edges in G does not leave all the resultant connected components all k -edge (respectively, k -vertex) connected.

They also show that any solution $G = (V, E)$ satisfying (I) and (II) for $k = 2$, is the *unique* optimal subgraph for the canonical weight function defined by G . They conjecture that these results would extend to any $k > 2$.

We note that for $k = 2$, conditions (I) and (II) ensure that the graph will be two-vertex connected. Hence, the cases for two-edge and two-vertex connectivity are in fact just one case. For $k \geq 3$, a minimum-weight k -edge connected subgraph can have a value strictly less than a minimum-weight k -vertex connected subgraph. For example, consider the 3-edge connected graph G in Fig. 1 with the canonical weight function defined by G .

In § 2, we consider the k -edge connected problem and show that some optimal solution satisfies conditions (I) and (II). We also show that conditions (I) and (II) do not characterize the class of minimum-weight k -edge connected subgraphs for $k \geq 3$, contrary to the case for $k = 2$.

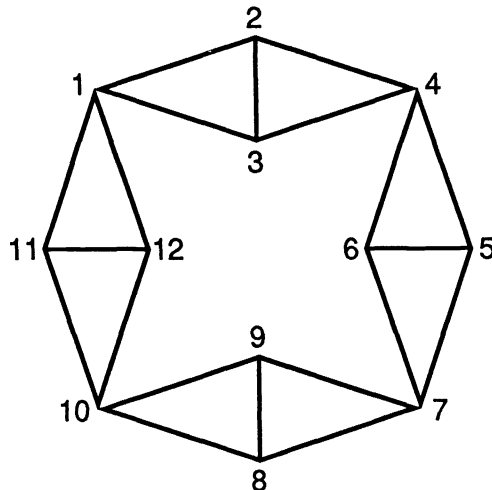


FIG. 1

In § 3, we consider the k -vertex connected case and show that some minimum-weight solution satisfies condition (II), and that it also satisfies condition (I) when $|V| \geq 2k$. This restriction is tight, since for every $k \geq 4$, we provide a k -vertex connected graph $G = (V, E)$ with $|V| = 2k - 1$, and maximum vertex degree of $(2k - 2)$, which is the unique optimal solution for the canonical weight function defined on G .

Our structural results are of use towards obtaining heuristics. In particular, the proof of the degree condition (I) yields a polynomial-time algorithm that, given a solution, will produce a new one that satisfies (I) without increasing cost. This is useful, as in practice it is very desirable to produce networks with vertices of small degree, and in a k -connected graph every vertex has degree at least k . Another way of restating these facts is the following: condition (I) shows that (except for small cases in the vertex-connected case) there are optimal solutions whose total number of edges is within a factor of $1 + 1/k$ of the least possible number of edges in any k -connected subgraph, independent of edge-weights. We remark that the structural properties derived in [MMP] for $k = 2$ have been used to obtain heuristics for producing low-cost networks for “real-world” problems in [MS].

In § 4, we present a polynomial-time heuristic that for each fixed $k \geq 2$ produces a solution at most a constant factor (depending on k) larger than the optimal value.

2. Case of edge-connectivity. In this section, we show that some optimal k -edge connected subgraph satisfies conditions (I) and (II). We also show that these conditions do *not* characterize the class of minimum-weight k -edge connected subgraphs for $k \geq 3$, contrary to the case for $k = 2$. The following definitions are due to Mader [Ma]. Let $G = (V, E)$ be a graph, with $x_i \in V$ for $0 \leq i \leq 2$, and with x_1 and x_2 adjacent to x_0 . Then the graph G' obtained from G by deleting (x_0, x_j) $j = 1, 2$, and adding (x_1, x_2) is called a *lifting at x_0* of G . If the edge-connectedness of G' , between any two vertices of $G' - x_0$, is not smaller than that of G , then the lifting is called *admissible*.

THEOREM 1. *For any set of vertices V with nonnegative symmetric weight function $d(\cdot, \cdot)$ satisfying the triangle inequality, and any $k \geq 2$, there exists a minimum-weight k -edge connected subgraph $G = (V, E)$ satisfying conditions (I) and (II).*

Proof of (I). Mader [Ma] proved the following results for a graph $H = (V, E)$ with $w \in V$:

(1) If the degree of w is at least 4, and w is not cutvertex, then H has an admissible lifting at w .

(2) If w is a cutvertex, but no single edge incident to w is a cutset, then H has an admissible lifting at w . Hence, suppose that x is a vertex of G with degree at least $k + 2$. Since $k \geq 2$, if x is not a cutvertex, then by (1) G has an admissible lifting. If x is a cutvertex, then there are at least $k \geq 2$ edges connecting x to each connected component of $G - x$, and all edges incident to x are accounted for in this way. Hence, no such edge can be a cutset of G , and (2) applies. Thus, let G' be an admissible lifting at x .

Now G' is k -edge connected between any two vertices y and z of $G' - x$. Since, in G' , the degree of x is at least k , we conclude that G' itself is k -edge connected. Further, by the triangle inequality, the weight of G' is no greater than that of G . Consequently, after repeating the above lifting procedure a finite number of times, we will obtain a minimum-weight k -edge connected graph with all degrees equal to k or $k + 1$. $\square$

Proof of (II). Let $G = (V, E)$ be a minimum-weight k -edge connected subgraph, where G has the minimum number of edges among all minimum-weight solutions. We suppose, in order to obtain a contradiction, that (II) does not hold; that is, removing 1, 2, $\dots$, or k edges leaves the resultant connected components k -edge connected.

Since G is k -edge connected, removing fewer than k edges from G leaves the resultant graph connected; if this resultant graph is still k -edge connected then we get a contradiction to the minimality of G . The same contradiction occurs if removing k edges from G leaves the graph connected. So, assume that the removal of the k edges (u_i, w_i) $1 \leq i \leq k$ results in two k -edge connected components on vertex sets U and W , with u_i in U , and w_i in W for $1 \leq i \leq k$. Let x_i be a neighbor of u_i in U , and y_i be a neighbor of w_i in W . The vertices u_i and x_i for $1 \leq i \leq k$ need not be distinct, and the vertices w_i and y_i for $1 \leq i \leq k$ need not be distinct. However, at least one of U , W must be nondegenerate, and consequently it must contain at least $k + 1$ vertices. Suppose W is nondegenerate. Then we may assume that $w_1 \neq w_2$, $w_1 \neq y_1$, and $w_2 \neq y_2$, and that y_i , $i = 1, 2$, is not an endpoint of any of the deleted edges.

By the triangle inequality, $d(x_1, u_1) + d(u_1, w_1) + d(w_1, y_1) \geq d(x_1, y_1)$ and $d(x_2, u_2) + d(u_2, w_2) + d(w_2, y_2) \geq d(x_2, y_2)$. Without loss of generality, $d(u_2, w_2) \geq d(u_1, w_1)$, and thus replacing edges (x_1, u_1) , (w_1, y_1) , and (u_2, w_2) by edge (x_1, y_1) does not increase the cost of the graph, but decreases the number of edges. To obtain the desired contradiction, it remains to be shown that the resulting graph is k -edge connected.

Consider two distinct vertices s and t in U . Since the subgraph of G induced by U is assumed to be k -edge connected, there were k edge-disjoint paths between s and t before the transformation cited above was performed. If none of these paths use the deleted edge (x_1, u_1) , then these same paths suffice. If one path uses the edge (x_1, u_1) , replace (x_1, u_1) in that path by the added edge (x_1, y_1) , a path from y_1 to w_1 in W , and the edge (w_1, u_1) to obtain k edge-disjoint paths in the transformed graph. A similar argument holds if both s and t are in W . Finally, suppose that s is in U and t is in W . Since the subgraph of G induced by U is k -edge connected even after removing the edge (x_1, u_1) , there must be k edge-disjoint paths from s to $u_1, x_1, u_3, u_4, \dots, u_k$ in U . Similarly, there are k edge-disjoint paths from t to $w_1, y_1, w_3, w_4, \dots, w_k$ in W even after removing the edge (w_1, y_1) . Together with the added edge (x_1, y_1) , and existing edges $(u_1, w_1), (u_3, w_3), \dots, (u_k, w_k)$, these paths provide k edge-disjoint paths from s to t . Hence, the transformed graph is a minimum-weight k -edge connected graph with fewer edges than G' , a contradiction. $\square$

We note that the graph G in Fig. 1 is 3-edge connected and satisfies conditions (I) and (II). Now, G is not the unique minimum-weight solution for any set of weights satisfying the triangle inequality. This follows since we may assume, without loss of generality, that $d(1, 3) \geq d(3, 4)$ by symmetry and that $d(3, 4) + d(4, 6) \geq d(3, 6)$ by the triangle inequality. Hence, removing the edges $(1, 3)$ and $(4, 6)$ and adding the edge $(3, 6)$ leaves the graph 3-edge connected but does not increase the cost. Therefore, conditions (I) and (II) do *not* characterize the class of minimum-weight solutions for $k = 3$ as they do in the case $k = 2$.

3. Case of vertex connectivity. In this section, we analyze the design problem in the vertex-connectivity case. This section will be divided into two parts. In the first part, we give a structural theorem that is an analogue of Theorem 1. This theorem uses a technical theorem, which is proved in the second part.

3.1. The structural theorem. We will prove the following result below.

THEOREM 2. *For any set of vertices V with nonnegative symmetric weight function $d(\cdot, \cdot)$ satisfying the triangle inequality, and any $k \geq 2$, there exists a minimum-weight k -vertex connected subgraph satisfying (II), and also satisfying (I) if $|V| \geq 2k$.*

Before proving this theorem, let us review the implications of its statement. At first glance, the additional boundary condition imposed in (I) might seem odd. We note, however, that this requirement is needed (and tight) since for any $k \geq 4$, there is a graph which is the unique minimum-weight k -vertex connected solution for the canonical weight function defined by G . We demonstrate this fact next. Consider a complete bipartite graph with vertex set $V_1 \cup V_2$ where $|V_1| = |V_2| = k - 1$. Add a vertex x that is adjacent to every other vertex. Call this graph $G = (V, E)$. G is k -vertex connected with $|V| = 2k - 1$ and the degree of x equal to $(2k - 2)$. To see that G is the unique minimum-weight solution for the canonical weight function on G , note that all edges in G have weight one and all edges not in G have weight two. Now, G has $(k^2 - 1)$ edges and any k -vertex connected graph has at least $(2k^2 - k/2)$ edges. Therefore, at most $p \leq (k - 2)/2$ edges of weight one can be removed from G and at most $p/2$ edges of weight two can be added to G to obtain another minimum-weight k -vertex connected solution. Since all the vertices other than x have degree equal to k , the removed edges must all be adjacent to vertex x ; if not, then some vertex in the resultant graph will have degree less than k . Since $V_1 \cup V_2$ form a complete bipartite graph, the edges of weight two added to G must be entirely contained in V_1 or V_2 . Therefore, the new graph is not k -vertex connected if any $p \geq 1$ edges of weight one are removed. Therefore, G is the unique minimum-weight k -vertex connected solution, as desired.

Informally, we can explain why the additional condition in (I) arises, as follows. In order to prove condition (I) for the vertex connected case, it would be useful to have available a result on liftings similar to the theorem of Mader, used in § 2 for the edge-connectivity case, but instead for vertex connectivity. We would only need to show that if G is k -vertex connected for $k \geq 2$ and x a vertex of G with $d(x) \geq k + 2$, then there is a lifting at x that is k -vertex connected. However, this result is not true. Consider the graph consisting of $k + 2$ copies, $D_1, \dots, D_{k+2}$, of the complete graph on k vertices, together with k additional vertices $x_1, \dots, x_k$. Add $k(k + 2)$ edges so that there is a perfect matching between $x_1, \dots, x_k$ and each of $D_1, \dots, D_{k+2}$, i.e., each x_i is adjacent to exactly one vertex in each D_j and each vertex $y \in D_j$ is adjacent to exactly one of $x_1, \dots, x_k$. Then for any lifting at x_1 , the vertices $x_2, \dots, x_k$ form a cut set.

The next technical theorem is the result we need for proving the revised condition (I). Essentially, it shows that we can always find either a single lifting, or a pair of liftings that will keep the graph k -vertex connected.

THEOREM 3. *Let $G = (V, E)$ be a minimal k -vertex connected graph with $|V| \geq 2k$ and $k \geq 2$. If $x \in V$ has degree at least $k + 2$ then either:*

- (i) *there exists a lifting of x that is k -vertex connected, or*
- (ii) *there exists a vertex $y \in V$ such that for any lifting G' at x , there exists a lifting at y of G' that is k -vertex connected.*

The requirement on the number of vertices arises naturally in the proof of Theorem 3, which shall be given in § 3.2.

Now we pass to the proof of Theorem 2.

Proof of (I). As in the proof of Theorem 1, this follows easily from Theorem 3 since, by the triangle inequality, the weight of G does not increase as a result of a lifting. $\square$

Proof of (II). Let $G = (V, E)$ be a minimum-weight k -vertex connected subgraph V , where G has the minimum number of edges among all minimum-weight solutions. We suppose, in order to obtain a contradiction, that (II) does not hold. That is, removing 1, 2, $\dots$, or k edges leaves the resultant connected components k -vertex connected.

Since G is k -vertex connected, removing fewer than k edges from G leaves the resultant graph connected; if this resultant graph is still k -vertex connected then we get

a contradiction to the minimality of G . The same contradiction occurs if removing k edges leaves the graph connected. So we assume that the removal of the k edges (u_i, w_i) $1 \leq i \leq k$ results in two k -vertex connected components on vertex set U and W with u_i in U and w_i in W for $1 \leq i \leq k$. We claim that either $|U| = 1$, $|W| = 1$ or all of the u_i and w_i are distinct. This follows since if $|U| > 1$, $|W| > 1$ and $u_1 = u_2$, then $|U| \geq k + 1$ and $|W| \geq k + 1$ since the subgraphs of G induced by U and W , respectively, are k -vertex connected, and so $\{u_1, w_3, w_4, \dots, w_k\}$ is a $(k - 1)$ -vertex separator of G .

Consider the case where $|U| = 1$; the proof for $|W| = 1$ is similar. Let u be the single vertex in U which is the endpoint of all of the k removed edges. Since G has no parallel edges, each w_i is distinct for $1 \leq i \leq k$. Also, since the subgraph of G induced by W is k -vertex connected, $|W| \geq k + 1$. Let y_i be a neighbor of w_i in W . The y_i 's need not be distinct. The triangle inequality implies that $d(u, w_i) + d(w_i, y_i) \geq d(u, y_i)$ for $1 \leq i \leq k$. So, without loss of generality, we may replace edges (w_1, y_1) and (u, w_2) by the edge (u, y_1) without increasing the cost of the graph. We claim that the resultant graph is still k -vertex connected, which would provide the desired contradiction. Consider u in U and any t in W . Since the subgraph of G induced by W is k -vertex connected, there are k vertex-disjoint paths from t to vertices $w_1, y_1, w_3, w_4, \dots, w_k$ even excluding the removed edge (w_1, y_1) . Continuing these paths with the added edge (u, y_1) and existing edges $(u, w_1), (u, w_3), (u, w_4), \dots, (u, w_k)$ yields the desired result.

Now, consider the case where $|U| \geq k + 1$, $|W| \geq k + 1$ and the vertices u_i and w_i are all distinct for $1 \leq i \leq k$. Let x_i be a neighbor of u_i in U , and let y_i be a neighbor of w_i in W . The x_i 's need not be distinct, and the y_i 's need not be distinct. By the triangle inequality, $d(x_i, u_i) + d(u_i, w_i) + d(w_i, y_i) \geq d(x_i, y_i)$ for $1 \leq i \leq k$. So, without loss of generality, replacing edges $(x_1, u_1), (w_1, y_1)$ and (u_2, w_2) by the edge (x_1, y_1) does not increase the total cost. We need only show that the resultant graph is k -vertex connected to obtain the desired contradiction.

First, consider distinct vertices s and t both in U ; the case where both s and t are in W is similar. Since the subgraph of G induced by U is k -vertex connected, there were k vertex-disjoint paths from s to t in U . If none of these paths used the deleted edge (x_1, u_1) then we are done. If one path used this edge then replace the removed edge (x_1, u_1) by the added edge (x_1, y_1) , a path in W from y_1 to w_1 , and the edge (u_1, w_1) to complete the path.

Finally, consider a vertex s in U , and a vertex t in W . Since the subgraph of G induced by U is k -vertex connected, there exists k vertex-disjoint paths from s to $u_1, x_1, u_3, \dots, u_k$ not using edge (x_1, u_1) . Similarly, there are k vertex-disjoint paths from t to $w_1, y_1, w_3, \dots, w_k$ not using edge (y_1, w_1) . Combining these paths with the added edge (x_1, y_1) and existing edges $(u_1, w_1), (u_3, w_3), \dots, (u_k, w_k)$ yields the desired result. Hence, the resultant graph is a minimum-weight k -vertex connected subgraph with fewer edges than G , a contradiction. $\square$

3.2. Proof of Theorem 3. Below we will prove the theorem on liftings which was used in the proof of Theorem 2. First we need some auxiliary definitions and results.

For a graph $G = (V, E)$ and a set of vertices $W \subseteq V$, let $\delta_G(W)$ be the set of vertices contained in $V \setminus W$ that is adjacent to at least one vertex in W . It is not hard to prove for $W, U \subseteq V$, that the following submodular inequality holds:

$$|\delta(W \cap U)| + |\delta(W \cup U)| \leq |\delta(W)| + |\delta(U)|.$$

Below we will require the following slightly stronger result that involves two graphs defined on the same vertex set.

PROPOSITION 1. Let $G = (V, E)$ and $G' = (V, E')$ be graphs. Let $W, U \subseteq V$. Suppose that $\delta_G(W \cap U) \subseteq \delta_{G'}(W \cap U)$ and $\delta_{G'}(W \cup U) \subseteq \delta_G(W \cup U)$. Then:

- (i) $|\delta_G(W \cap U)| + |\delta_{G'}(W \cup U)| \leq |\delta_G(W)| + |\delta_{G'}(U)|$, and
- (ii) if, in addition, G and G' are k -vertex connected and $|\delta_G(W)| = |\delta_{G'}(U)| = k$, $W \cap U \neq \emptyset$, and $|W \cup U| \leq |V| - k$ then $|\delta_G(W \cup U)| = |\delta_{G'}(W \cup U)| = k$.

Proof. Since $|X| + |Y| = |X \cup Y| + |X \cap Y|$, we can prove (i) by proving that $\delta_G(W \cap U) \cap \delta_{G'}(W \cup U) \subseteq \delta_G(W) \cap \delta_{G'}(U)$, and $\delta_G(W \cap U) \cup \delta_{G'}(W \cup U) \subseteq \delta_G(W) \cup \delta_{G'}(U)$. To prove the first condition suppose $y \in \delta_G(W \cap U) \cap \delta_{G'}(W \cup U)$. Then $y \notin W \cup U$ and is in $\delta_G(W \cap U)$. So y is adjacent in G to a vertex in $W \cap U$. Thus $y \in \delta_G(W)$. Also $\delta_G(W \cap U) \subseteq \delta_{G'}(W \cap U)$ so $y \in \delta_{G'}(U)$. The proof of the latter condition is similar.

To prove (ii), observe that either $|W \cap U| = |V| - k$ or $\delta_G(W \cap U)$ is a cutset in G . So $|\delta_G(W \cap U)| \geq k$. Also, either $\delta_{G'}(W \cup U) = V \setminus (W \cup U)$ or $\delta_{G'}(W \cup U)$ is a cutset in G' . So $|\delta_{G'}(W \cup U)| \geq k$. So (ii) follows from (i). $\square$

We will say that a set $W \subseteq V$ is a (k, T, x) separator in G if $\emptyset \neq T \subseteq W$, $x \notin W \cup \delta(W)$ and $|\delta(W)| = k$.

PROPOSITION 2. Let G be a k -vertex connected graph. Let $x \in V$ and $T \subseteq V$. If there exists a (k, T, x) separator in G , then there exists a unique maximal (k, T, x) separator in G .

Proof. If W and W' are both (k, T, x) separators then $T \subseteq W \cap W'$ and $x \notin W \cup W' \cup \delta(W) \cup \delta(W')$. So by Proposition 1, $W \cup W'$ is a (k, T, x) -separator. Thus $W = \bigcup \{Z \mid Z \text{ is a } (k, T, x)\text{-separator}\}$ is a maximal (k, T, x) -separator. $\square$

Next we prove Theorem 3, which we restate here.

THEOREM 3. Let $G = (V, E)$ be a minimal k -vertex connected graph with $|V| \geq 2k$ and $k \geq 2$. If $x \in V$ has degree at least $k + 2$ then either:

- (i) there exists a lifting of x that is k -vertex connected, or
- (ii) there exists a vertex $y \in V$ such that for any lifting G' at x , there exists a lifting at y of G' that is k -vertex connected.

Proof. Let $x \in V$ have degree at least $k + 2$. Suppose there is no lifting at x that is k -vertex connected. We will show that condition (ii) holds.

Let d be the degree of x and let $x_1, \dots, x_d$ be the neighbors of x . For each pair of neighbors x_i and x_j of x , let G_{ij} be the lifting at x with respect to vertices x_i and x_j . By assumption, G_{ij} is not k -vertex connected. Let S be a cutset in G_{ij} with $|S| \leq k - 1$. Let W be a connected component of the graph $G_{ij} \setminus S$. Since S is not a cutset in G , W contains at least one of the vertices x, x_i or x_j . So $G_{ij} \setminus S$ contains exactly two connected components W and U . Without loss of generality, $x \in U$, $\{x_i, x_j\} \subseteq W \cup S$, and $x_i \in W$. Since x has at least k neighbors in G_{ij} , $|U| \geq 2$. $S \cup \{x\}$ is a cutset in G , so $|S| = k - 1$. If there exists a $(k - 1, x_i, x)$ separator in G_{ij} , let W_{ij} be a maximal $(k - 1, x_i, x)$ separator in G_{ij} . If there is no $(k - 1, x_i, x)$ separator in G_{ij} , then define $W_{ij} = \emptyset$. Note that if $x_j \in W_{ij}$, then $W_{ij} = W_{ji}$. Let $S_{ij} = \delta_{G_{ij}}(W_{ij})$. Let $U_{ij} = V \setminus (W_{ij} \cup S_{ij})$.

Similarly, for each neighbor x_i of x , let G_i be the graph G with the edge (x, x_i) deleted. Since G is minimal k -vertex connected, G_i is not k -vertex connected, but it is $(k - 1)$ vertex connected, and there exists a $(k - 1, x_i, x)$ separator in G_i . Let W_i be the maximal $(k - 1, x_i, x)$ separator in G_i . Let $S_i = \delta_{G_i}(W_i)$ and let $U_i = V \setminus (W_i \cup S_i)$.

The remainder of the proof will be broken into several steps.

Step 3.1. For any two neighbors x_i and x_j of x , $W_i \cap W_j = \emptyset$. If $W_{ij} \neq \emptyset$ then $W_i \subseteq W_{ij}$ and if, in addition, $x_j \notin W_{ij}$, then $W_i = W_{ij}$.

Proof. The only edge that G_i contains that is not an edge of G_j is (x, x_j) , and the only edge that G_j contains that might not be an edge of G_i is the edge (x, x_i) . Thus, it is easy to verify that $\delta_{G_i}(W_i \cap W_j) \subseteq \delta_{G_j}(W_i \cap W_j)$, and $\delta_{G_j}(W_i \cup W_j) \subseteq \delta_{G_i}(W_i \cup W_j)$.

Now x and all neighbors of x except for x_i and x_j are not in $W_i \cup W_j$. So $|W_i \cup W_j| \leq |V| - (k + 1)$. If $W_i \cap W_j \neq \emptyset$, then by Proposition 1, $|\delta_{G_i}(W_i \cap W_j)| = k - 1$. But $x_i \notin W_i \cap W_j$, so $\delta_G(W_i \cap W_j) = \delta_{G_i}(W_i \cap W_j)$. This contradicts the fact that G is k -vertex connected and establishes the first condition.

Similarly, it is also easy to verify that if $W_{ij} \neq \emptyset$ then $\delta_{G_i}(W_i \cap W_{ij}) \subseteq \delta_{G_{ij}}(W_i \cap W_{ij})$, $\delta_{G_{ij}}(W_i \cup W_{ij}) \subseteq \delta_{G_i}(W_i \cup W_{ij})$, and $|W_i \cup W_{ij}| \leq |V| - (k + 1)$. Also, $W_i \cap W_{ij} \neq \emptyset$ since $x_i \in W_i \cap W_{ij}$. So by Proposition 1, $|\delta_{G_{ij}}(W_i \cup W_{ij})| = k - 1$. By the maximality of W_{ij} , $W_i \subseteq W_{ij}$. If $x_j \notin W_{ij}$, then W_{ij} is a $(k - 1, x_i, x)$ separator in G_i . By the maximality of W_i , $W_{ij} \subseteq W_i$, which establishes the second condition. $\square$

Step 3.2. If $W_{ij} \neq W_{ji}$ then $W_j \cap S_i \neq \emptyset$.

Proof. If $W_{ij} \neq \emptyset$ and $W_{ij} \neq W_{ji}$, then $x_j \notin W_{ij}$ and $W_i = W_{ij}$. Now $\delta_{G_i}(W_i) \subseteq \delta_{G_{ij}}(W_i)$ and $|\delta_{G_i}(W_i)| = k - 1 = |\delta_{G_{ij}}(W_i)|$ imply that $\delta_{G_i}(W_i) = \delta_{G_{ij}}(W_i)$. Consequently, since $x_j \in \delta_{G_{ij}}(W_i)$, we have $x_j \in \delta_{G_i}(W_i)$.

If $W_{ij} \neq \emptyset$, then $W_{ji} \neq \emptyset$ and $x_i \in \delta_{G_j}(W_j)$. There exists a path in G_j from x_j to x_i with all vertices of the path except x_i contained in W_j . Since $x_j \notin W_i$, there must be a vertex on this path contained in S_i . $\square$

In reading the following, it will be helpful to consider the diagram in Fig. 2. This is a partition of the vertices of G into nine disjoint sets. Two of the sets in Fig. 2 are said to be adjacent if their boundaries in Fig. 2 share a point or a line segment in the diagram. (Note that a set is adjacent to itself.) Then two vertices in G can be adjacent only if the two sets that contain them are adjacent in Fig. 2. $\square$

Step 3.3. If $W_{ij} \neq \emptyset$ and $W_i \neq \emptyset$, then $W_i = W_{ij} \cap W_i$, and $S_i = (S_{ij} \cap S_i) \cup (S_i \cap W_{ij}) \cup (S_{ij} \cap W_i)$.

Proof. Clearly, $x_i \in W_{ij} \cap W_i$, and x together with at least $(k - 1)$ neighbors of x are not in $W_{ij} \cup W_i$. Hence by Proposition 1, $|\delta_G(W_{ij} \cap W_i)| = k = |\delta_G(W_{ij} \cup W_i)|$.

Now $x \in \delta_G(W_{ij} \cap W_i)$ but $x \notin \delta_{G_i}(W_{ij} \cap W_i)$. So $W_{ij} \cap W_i$ is a $(k - 1, x_i, x)$ separator in G_i . By Step 3.1 and the maximality of W_i , $W_i = W_{ij} \cap W_i$.

Let $T = (S_{ij} \cap S_i) \cup (S_{ij} \cap U_i) \cup (S_i \cap U_{ij})$, and let $T' = (S_{ij} \cap S_i) \cup (S_{ij} \cap W_i) \cup (S_i \cap W_{ij})$. Now, $|T| + |T'| = |S_{ij}| + |S_i|$. Also, $\delta_G(W_{ij} \cap W_i) \subseteq \{x\} \cup T'$ and $\delta_G(W_{ij} \cup W_i) \subseteq \{x\} \cup T$. So $|T| = |T'| = k - 1$, and $\delta_G(W_{ij} \cap W_i) = \{x\} \cup T'$. Hence, $S_i = (S_{ij} \cap S_i) \cup (S_{ij} \cap W_i) \cup (S_i \cap W_{ij})$. $\square$

Step 3.4. If $W_j \cap S_i = \emptyset$ then $S_i = S_j = S_{ij}$.

Proof. From Steps 3.2 and 3.1, and since $|S_i| = k - 1$, we see that there is another neighbor x of x such that $W \cap S_i = \emptyset$. By Step 3.2, we get $x_j \in W_{ij}$. But $x_j \notin W_{ij} \cap$

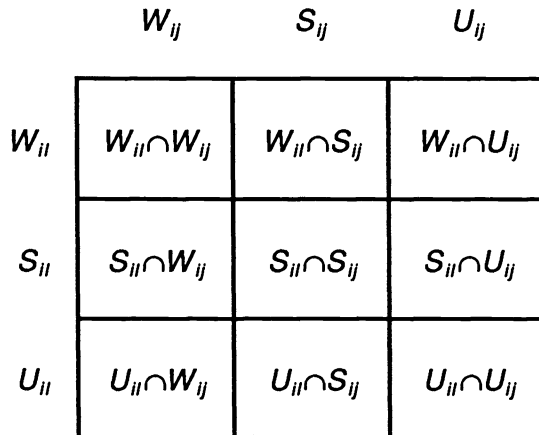


FIG. 2

W_i by Step 3.3, and $x_j \notin W_{ij} \cap S_i$. Since $x_j \notin S_i$, we have $x_j \in W_{ij} \cap U_i$. Similarly, $x \in W_i \cap U_{ij}$.

Let $R = (S_{ij} \cap S_i) \cup (S_{ij} \cap U_i) \cup (S_i \cap W_{ij})$, and let $R' = (S_{ij} \cap S_i) \cup (S_i \cap U_{ij}) \cup (S_{ij} \cap W_i)$. Now $|R| + |R'| = |S_{ij}| + |S_i|$. Also, $\delta_G(W_{ij} \cap U_i) \subseteq \{x\} \cup R$, and $\delta_G(W_i \cap U_{ij}) \subseteq \{x\} \cup R'$. Hence, $|R| = |R'| = k - 1$. So $W_{ij} \cap U_i$ is a $(k - 1, x_j, x)$ separator in G_j . Since $W_j \subseteq W_{ij}$, $W_j \cap W_{ij} \cap W_i = \emptyset$ and $W_j \cap W_{ij} \cap S_i = \emptyset$, we have $W_j \subseteq W_{ij} \cap U_i$. By the maximality of W_j , we get $W_j = W_{ij} \cap U_i$ and $S_j = R$. Similarly, $W = W_i \cap U_{ij}$ and $S = R'$. Thus, $S_j \cap S = S_i \cap S_j \cap S$. We also get that $W_a \cap S_b = \emptyset$ for $\{a, b\} \subseteq \{i, j, k\}$. So from $S_i \cap S_j = S_i \cap S_j \cap S$, we get $S_i \cap W_{ij} = \emptyset$. Similarly, $S_{ij} \cap W_{ie} = \emptyset$. Thus, $S_i = S_{ij} \cap S_i$. So $|S_{ij} \cap S_i| = k - 1$ and $S_j = S_{ij} \cap S_i = S_i = S_{ij}$. $\square$

Step 3.5. $W_{ij} = W_{ji}$ if and only if $x_j \notin S_i$.

Proof. Suppose $W_{ij} = W_{ji}$. Let x be a vertex such that $W \cap S_i = \emptyset$. By Steps 3.3 and 3.4, we have $S_i = (S_i \cap W_{ij}) \cup (S_{ij} \cap S_i) \cup (S_{ij} \cap W_i) = S_i$. So $S_{ij} \cap W_i = \emptyset = S_i \cap U_{ij}$. Thus, $\delta_G(W_i \cap U_{ij}) = (S_{ij} \cap S_i) \cup \{x\}$, and $|S_{ij} \cap S_i| \geq k - 1$. So $S_{ij} = S_{ij} \cap S_i = S_i = S_i$ and $x_j \notin S_i$.

Suppose $x_j \notin S_i$. Let X be the set of neighbors of x such that $W \cap S_i = \emptyset$ for each $x \in X$. So $S_i = S$ for each $x \in X$. Now if $W \cap S_j = \emptyset$, then $S_j = S = S_i$ and $W_{ij} = W_{ji}$ by Step 3.2. Hence we may assume that $W \cap S_j \neq \emptyset$ for each $x \in X$. There exist disjoint paths from x_j to the vertices in S_j with each path contained in $W_j \cup S_j$. For each $x \in X$ and for each vertex y in $W \cap S_j$, the path from x_j to y must contain a vertex of $S = S_i$. So $|S_i \cap W_j| \geq |X|$. But for each neighbor $x_g \notin X$, $|S_i \cap W_g| \geq |X|$. So $|S_i| \geq d(x) - 1$. $\square$

Step 3.6. Let x_{i_0} be a neighbor of x . Let $x_1, \dots, x_t$ be the neighbors of x that are not in S_{i_0} . Then $V = \{x\} \cup S_{i_0} \cup_{i=1}^t W_i$.

Proof. Let G' be the graph G induced on $V \setminus S_{i_0}$ with edges (x, x_i) deleted for $1 \leq i \leq t$. By Step 3.5, x is an isolated vertex in G' . So if $y \in V$ but $y \notin \{x\} \cup S_{i_0} \cup_{i=1}^t W_i$, then S_{i_0} separates y from x in G . $\square$

Step 3.7. Let x_i be a neighbor of x . Then S_i contains no neighbors of x .

Proof. Suppose $x_j \in S_i$. Let $x_{i_1}, \dots, x_{i_t}$ be the neighbors of x that are not in S_i . Let $x_{j_1}, \dots, x_{j_s}$ be the neighbors of x that are not in S_j . Then $V = \{x\} \cup S_i \cup_{i=1}^t W_i = \{x\} \cup S_j \cup_{j=1}^s W_j$.

By Step 3.5, $\{x_{j_1}, \dots, x_{j_s}\} \cap \{x_{i_1}, \dots, x_{i_t}\} = \emptyset$. So $V \subseteq \{x\} \cup S_i \cup S_j$. Hence $|V| \leq 2k - 1$. $\square$

We are now ready to complete the proof of Theorem 3. By Steps 3.5 and 3.7, we have $S_i = S_j$ for any two neighbors x_i , and x_j of x . So let $S = S_i$, and let $y \in S$.

Let y_i be a vertex W_i that is adjacent to y . Given two neighbors, x_1 and x_2 of x , let G' be the lifting at y of G_{12} that is obtained by deleting edges (y, y_2) and (y, y_3) and adding edge (y_2, y_3) . We can now finish the proof of Theorem 3.1.

Step 3.8. G' is k -connected.

Proof. Since G is k -vertex connected, we need only show that there are k vertex disjoint paths in G' between each of the pairs (x, x_1) , (x, x_2) , (y, y_2) , and (y, y_3) . In the remainder of the proof, we will say that a path P is contained in a vertex set $Y \subseteq V$ if all vertices of P except for the endpoints are in Y . We will say that a set of paths $P_1, \dots, P_t$ are paths from u to $u_1, \dots, u_t$ if path P_i is a path from u to u_i . Let $z_1, \dots, z_{k-1}$ be the vertices in S with $y = z_1$.

For (x, x_1) let $P_1, \dots, P_{k-1}$ be vertex disjoint paths in G' contained in W_1 from x_1 to $z_1, \dots, z_{k-1}$. Extend P_i along a path in W_{i+3} to x . Let P_k be a path x_1 to x_2 extended along a path to y_2 contained in W_2 , then to y_3 and then along a path in W_3 to x .

For (x, x_2) let $P_1, \dots, P_k$ be vertex disjoint paths in G from x_2 to $z_1, \dots, z_{k-1}, y_2$. If one of these paths contains the vertex x , call that path P . P cannot be the path from x_2 to y_2 . All paths except P are paths in G' . Extend P_k in G' to y_3 and then along a path in W_3 to x . For $P_i \neq P$, extend P_i along a path in W_{i+3} to x . If z_j is an endpoint of P , then truncate P at x_2 and extend it in G' to x and then to z_j in W_1 and then along a path in W_{j+3} to x .

For (y, y_2) , let $P_1, \dots, P_k$ be vertex disjoint paths in G contained in W_2 from y_2 to $z_1, \dots, z_{k-1}, x_2$. All paths except P_1 are paths in G' . Replace P_1 with a path in G' starting at y_2 , going to y_3 , then to x along a path in W_3 , and then to y along a path in W_4 . For $2 \leq i \leq k-1$, extend P_i to y_{i+3} along a path in W_{i+3} and then to y . Extend P_k to x , and then to y along a path in W_1 .

For (y, y_3) let $P_1, \dots, P_k$ be vertex disjoint paths in G contained in W_3 from y_3 to $z_1, \dots, z_{k-1}, x$. All paths except P_1 are paths in G' . Replace P_1 with a path in G' starting at y_3 , going to y_2 , then to x_2 along a path in W_2 , then to x_1 , and then to y along a path in W_1 . For $2 \leq i \leq k-1$, extend P_i to y along a path in W_{i+2} .

This completes the proof of Theorem 3. $\square$

4. Concluding remarks. We have derived conditions on the class of minimum-weight k -edge (or k -vertex) connected networks, where the distances between the points satisfy the triangle inequality. This generalizes recent results [MMP] for the case $k = 2$. We also showed that these conditions do not characterize the optimal solutions for $k \geq 3$, contrary to the case $k = 2$. We leave as an open question the problem of determining additional properties which characterize these classes.

For completeness, we mention that for each fixed $k \geq 2$, there is a polynomial-time algorithm that produces a solution of cost at most a constant factor (depending on k) of the optimum. This is obtained as follows: start with a Hamiltonian cycle C produced by the Christofides heuristic. It is well known that the value of C is at most $\frac{3}{2}$ that of an optimal 2-connected spanning subgraph [FJ]. Next, let C^k denote the graph obtained from C by adding an edge to join any two vertices that are within distance k in C (distance in the graph-theoretic sense). It is not difficult to see that C^k is $2k$ -connected, and that its value is at most $3k(k+1)/4$ times that of an optimal 2-connected spanning subgraph. Since the cost of any k -connected spanning subgraph is at least that of an optimal 2-connected spanning subgraph, we obtain the desired result.

REFERENCES

- [CW] N. CHRISTOFIDES AND C. A. WHITLOCK, *Network synthesis with connectivity constraints: a survey*, in Operational Research '81, J. P. Brans, ed., North-Holland, Amsterdam, pp. 706–723.
- [FJ] G. N. FREDERIKSON AND J. JÁ JÁ, *On the relationship between the biconnectivity augmentation and the traveling salesman problem*, Theoret. Comput. Sci., 13 (1982), pp. 189–202.
- [GM] M. GRÖTSCHEL AND C. L. MONMA, *Integer polyhedra arising from certain network design problems with connectivity constraints*, SIAM J. Discrete Math., to appear.
- [Ma] W. MADER, *A reduction method for edge-connectivity in graphs*, Ann. Discrete Math., 3 (1978), pp. 145–164.
- [MMP] C. L. MONMA, B. S. MUNSON, AND W. R. PULLEYBLANK, *Minimum-weight two-connected spanning networks*, Math. Programming, to appear.
- [MS] C. L. MONMA AND D. F. SHALLCROSS, *Methods for designing communications networks with certain two-connected survivability constraints*, Oper. Res., 37 (1989), pp. 531–541.
- [SWK] K. STEIGLITZ, P. WEINER, AND D. J. KLEITMAN, *The design of minimum-cost survivable networks*, IEEE Trans. Circuit Theory, 16 (1969), pp. 455–460.

GENERALIZATIONS OF THE NORMAL BASIS THEOREM OF FINITE FIELDS*

NADER H. BSHOUTY† AND GADIEL SEROUSSI‡

Abstract. A combinatorial characterization of sets of integers $\{r_0, r_1, \dots, r_{n-1}\}$, with $0 \leq r_i \leq q^n - 2$, such that $\alpha^{r_0}, \alpha^{r_1}, \dots, \alpha^{r_{n-1}}$ form a basis of $GF(q^n)$ over $GF(q)$ for some $\alpha \in GF(q^n)$ is presented. This characterization is used to prove the following generalization of the normal basis theorem for finite fields of characteristic two: Let $\lambda_0, \lambda_1, \dots, \lambda_{n-1}$ be integers in the range $0 \leq \lambda_i < q$, with at most one λ_i equal to zero. Then, there exists an element $\alpha \in GF(q^n)$ such that $\alpha^{\lambda_0}, \alpha^{\lambda_1 q}, \alpha^{\lambda_2 q^2}, \dots, \alpha^{\lambda_{n-1} q^{n-1}}$ form a basis of $GF(q^n)$ over $GF(q)$. This result, which includes the normal basis theorem as a particular case when $\lambda_0 = \lambda_1 = \dots = \lambda_{n-1} = 1$, is proved for all choices of $\lambda_0, \lambda_1, \dots, \lambda_{n-1}$ satisfying the above conditions when n is odd, and for more restricted sets of values $\{\lambda_i\}$ when n is even.

Key words. finite fields, normal bases

AMS(MOS) subject classification. 12

1. Introduction. Let $q = p^t$ for some prime integer p and integer $t \geq 1$, let $F = GF(q)$ denote the finite field of q elements, and let $K = GF(q^n)$, $n \geq 1$, denote the n th degree extension of F . The classical *normal basis theorem* for finite fields can be stated as follows (see, for instance, [1, p. 60], [2, p. 122]):

THEOREM 1. *There exists an element $\alpha \in K$ such that $\{\alpha, \alpha^q, \alpha^{q^2}, \dots, \alpha^{q^{n-1}}\}$ is a basis of K over F .*

The basis referred to in Theorem 1 is called a *normal basis*. Normal bases have recently received special attention in the literature due to their application to the design of efficient multipliers for finite fields ([3]–[6]).

In this paper we conjecture and partially prove the following generalization of the normal basis theorem: Let $\lambda_0, \lambda_1, \lambda_2, \dots, \lambda_{n-1}$ be integers in the range $0 \leq \lambda_i < q$, with at most one of the λ_i equal to zero. Then, there exists an element $\alpha \in K$ such that $\alpha^{\lambda_0}, \alpha^{\lambda_1 q}, \alpha^{\lambda_2 q^2}, \dots, \alpha^{\lambda_{n-1} q^{n-1}}$ form a basis of K over F . We call this a *generalized normal basis theorem*. Notice that the normal basis theorem is a particular case of the generalized normal basis theorem, with $\lambda_0 = \lambda_1 = \lambda_2 = \dots = \lambda_{n-1} = 1$.

Our generalization of the normal basis theorem is based on a combinatorial characterization of sets of integers $\{r_0, r_1, \dots, r_{n-1}\}$ such that $\alpha^{r_0}, \alpha^{r_1}, \dots, \alpha^{r_{n-1}}$ form a basis of K over F for some $\alpha \in K$. This characterization is presented in § 2, while its application to the generalized normal basis theorem is presented in § 3, where we prove the theorem in full generality for q even and n odd, and in a more restricted form for q and n even. Finally, in the concluding § 4 we present other potential applications of the results of § 2, conjecture that the generalized normal basis theorem is true for all finite fields, and present evidence to the validity of the conjecture.

The following notation is used throughout the paper: for integers or polynomials a and b , $a \nmid b$ denotes “ a does not divide b ,” $a \bmod b$ is the least positive (or least degree) residue of a modulo b , $\det(M)$ denotes the determinant of a matrix M , and $|A|$ denotes the cardinality of a set A .

* Received by the editors April 10, 1989; accepted for publication (in revised form) August 18, 1989.

† Department of Computer Science, Technion-Israel Institute of Technology, Haifa 32000, Israel.

‡ Hewlett-Packard Laboratories, 1501 Page Mill Road, Palo Alto, California 94304. Part of this work was done while the author was with the Department of Computer Science, Technion-Israel Institute of Technology and with Cyclotomics Incorporated, Berkeley, California.

2. A combinatorial characterization of bases. Let $x_0, x_1, x_2, \dots, x_{n-1}$ be indeterminates, and define the polynomial

$$(1) \quad \Delta(x_0, x_1, \dots, x_{n-1}) = \det \begin{pmatrix} x_0 & x_1 & \cdots & x_{n-1} \\ x_0^q & x_1^q & \cdots & x_{n-1}^q \\ \vdots & \vdots & \ddots & \vdots \\ x_0^{q^{n-1}} & x_1^{q^{n-1}} & \cdots & x_{n-1}^{q^{n-1}} \end{pmatrix}.$$

The following is a well-known characterization of bases of K over F (see, for instance, [1, p. 62]).

LEMMA 1. *A set of n elements $\beta_0, \beta_1, \beta_2, \dots, \beta_{n-1} \in K$ form a basis of K over F if and only if*

$$\Delta(\beta_0, \beta_1, \dots, \beta_{n-1}) \neq 0.$$

Let $R = \{r_0, r_1, \dots, r_{n-1}\}$ be a set of integers in the range $0 \leq r_i \leq q^n - 2$, $0 \leq i \leq n-1$. We say that $\alpha \in K$ satisfies R if $\alpha^{r_0}, \alpha^{r_1}, \dots, \alpha^{r_{n-1}}$ form a basis of K over F . We say that R is *satisfiable* if there exists an element $\alpha \in K$ such that α satisfies R . (Of course, satisfiability is relative to the particular choice of field K and subfield F . In the sequel, we assume that these fields are well defined in the context.)

LEMMA 2. *Let $D_R(x) = \Delta(x^{r_0}, x^{r_1}, \dots, x^{r_{n-1}})$. Then, R is satisfiable if and only if $(x^{q^n-1} - 1) \nmid D_R(x)$.*

Proof. Assume that $\alpha \in K$ satisfies R . Obviously, α must be nonzero. By the definition of $D_R(x)$, and by Lemma 1,

$$D_R(\alpha) = \Delta(\alpha^{r_0}, \alpha^{r_1}, \dots, \alpha^{r_{n-1}}) \neq 0.$$

Hence, α is not a root of $D_R(x)$. Since α is a root of $(x^{q^n-1} - 1)$ (all nonzero elements of K are), it follows that $(x^{q^n-1} - 1) \nmid D_R(x)$. The proof of the "if" part is a straightforward reversion of the above argument. $\square$

The main results of this paper are based on the following theorem.

THEOREM 2. *Let $R = \{r_0, r_1, \dots, r_{n-1}\}$ be a set of integers in the range $0 \leq r_i \leq q^n - 2$, and let s be an arbitrary integer. Consider the congruence*

$$(2) \quad \sum_{i=0}^{n-1} r_i q^{\phi(i)} \equiv s \pmod{(q^n - 1)},$$

where ϕ denotes a permutation on $\{0, 1, \dots, n-1\}$. Let $N_e(s)$ (respectively, $N_o(s)$) denote the number of even (respectively, odd) permutations ϕ satisfying (2). Then, R is satisfiable if and only if there exists an integer s such that

$$(3) \quad N_e(s) \not\equiv N_o(s) \pmod{p}.$$

Proof. Consider the polynomial $D_R(x)$ defined in Lemma 2. By straightforward application of the determinant formula, we obtain

$$(4) \quad D_R(x) = \Delta(x^{r_0}, x^{r_1}, \dots, x^{r_{n-1}}) = \sum_{\phi \in S_n} \left(\prod_{i=0}^{n-1} x^{r_i q^{\phi(i)}} \right) \text{sg}(\phi) = \sum_{\phi \in S_n} x^{\sum_i r_i q^{\phi(i)}} \text{sg}(\phi).$$

Here, S_n denotes the symmetric group on $\{0, 1, \dots, n-1\}$, and $\text{sg}(\phi)$ denotes the sign of $\phi \in S_n$. Let

$$(5) \quad \bar{D}_R(x) = \sum_{s=0}^{q^n-2} D_s x^s$$

denote the least degree residue of $D_R(x)$ modulo $(x^{q^n-1} - 1)$. By Lemma 2, R is satisfiable if and only if $\bar{D}_R(x) \neq 0$, that is, if and only if $D_s \neq 0$ for some $0 \leq s \leq q^n - 2$. An explicit expression for the coefficients D_s can be obtained by grouping together the contributions of terms at the right-hand side of (4) whose exponents are congruent to s modulo $(q^n - 1)$. Hence, we obtain

$$(6) \quad D_s = \sum_{\phi \in \Phi(s)} \text{sg}(\phi),$$

where

$$(7) \quad \Phi(s) = \left\{ \phi \in S_n \left| \sum_{i=0}^{n-1} r_i q^{\phi(i)} \equiv s \pmod{(q^n - 1)} \right. \right\}.$$

Since $\Phi(s)$ is the set of solutions ϕ to (2), it follows from (6) and from the definitions of $N_e(s)$ and $N_o(s)$ that $D_s = N_e(s) - N_o(s)$. The latter is an operation on field integers, and thus it is carried out modulo p . Therefore, $D_s \neq 0$ if and only if $N_e(s) \neq N_o(s) \pmod{p}$. $\square$

3. Generalization of the normal basis theorem for fields of characteristic two. In this section, we consider finite fields of characteristic 2, i.e., $F = GF(q)$, with $q = 2^t$. We start with a series of lemmas leading to the main results of the section.

The following lemma is a generalization of the uniqueness of radix- q representation of numbers. Its proof is deferred to the Appendix.

LEMMA 3. *Let n and m be integers with $1 \leq m \leq n$, and let $\lambda_0, \lambda_1, \dots, \lambda_{m-1}, k_0, k_1, \dots, k_{m-1}, j_0, j_1, \dots, j_{m-1}$, be integers satisfying the following conditions:*

$$(8) \quad 0 < \lambda_0, \lambda_1, \dots, \lambda_{m-1} < q,$$

$$(9) \quad 0 \leq k_0, k_1, \dots, k_{m-1} \leq n-1,$$

$$(10) \quad 0 \leq j_0 < j_1 < \dots < j_{m-1} \leq n-1,$$

$$(11) \quad \sum_{i=0}^{m-1} \lambda_i q^{k_i} \equiv \sum_{i=0}^{m-1} \lambda_i q^{j_i} \pmod{(q^n - 1)}.$$

Then,

$$(12) \quad \{k_0, k_1, \dots, k_{m-1}\} = \{j_0, j_1, \dots, j_{m-1}\}.$$

In the following, all arithmetic operations on indices from the set $\{0, 1, \dots, n-1\}$ are carried out modulo n , unless noted otherwise. Also, we decompose n (the extension degree of K over F) as $n = 2^k m$, with $k \geq 0$ and m odd.

Let B_n denote the set of permutations $\Psi \in S_n$ satisfying the following properties:

$$(P1) \quad \Psi(i) = \Psi(i+m) + m, \quad 0 \leq i \leq n-1,$$

$$(P2) \quad \{i + \Psi(i) \mid 0 \leq i \leq m-1\} = \{0, 2, \dots, 2(m-1)\}.$$

Note that if property (P1) is satisfied, then $\Psi(i) + i = \Psi(i+m) + (i+m)$, $0 \leq i \leq n-1$. It follows that when (P1) is true, (P2) is equivalent to

$$(P2') \quad \{i + \Psi(i) \mid 0 \leq i \leq n-1\} = \{0, 2, \dots, 2(m-1)\}.$$

Note also that the set at the right-hand side of (P2) and (P2') contains exactly m elements, even though its members are computed modulo n .

LEMMA 4. (i) If $\Psi \in B_n$, then $\Psi^{-1} \in B_n$.

(ii) There is one and only one permutation Ψ_0 in B_n such that $\Psi_0 = \Psi_0^{-1}$. The permutation Ψ_0 is defined by

$$(13) \quad \Psi_0(i) = \begin{cases} i & 0 \leq i \leq m-1, \\ \Psi_0(i-m) - m & m \leq i \leq n-1. \end{cases}$$

(iii) $|B_n|$ is odd.

Proof. (i) Assume $\Psi \in B_n$. Let $i \in \{0, 1, \dots, n-1\}$, and let $j = \Psi^{-1}(i+m)$. By property (P1) of Ψ , we have $\Psi(j+m) = \Psi(j) - m$. Substituting $\Psi^{-1}(i+m)$ for j , we obtain $\Psi(\Psi^{-1}(i+m) + m) = i$, which implies that $\Psi^{-1}(i+m) + m = \Psi^{-1}(i)$. Hence, Ψ^{-1} satisfies (P1). To show that Ψ^{-1} satisfies (P2), notice that if Ψ satisfies (P2') then so does Ψ^{-1} . Since (P2') is equivalent to (P2) once (P1) is established, Ψ^{-1} satisfies (P2).

(ii) It can be readily verified that $\Psi_0 \in B_n$, and that $\Psi_0 = \Psi_0^{-1}$. Assume that $\Psi \in B_n$, and $\Psi = \Psi^{-1}$. Furthermore, assume that $\Psi \neq \Psi_0$. By property (P1), Ψ is uniquely determined by its values at $0, 1, \dots, m-1$. Therefore, we must have $\Psi(h) \neq \Psi_0(h) = h$ for some $h, 0 \leq h \leq m-1$, and we can write

$$(14) \quad \Psi(h) = j + wm, \quad 0 \leq j \leq m-1, \quad 0 \leq w < 2^k,$$

where either $j \neq h$ or $w > 0$. If $j \neq h$, then, since $\Psi = \Psi^{-1}$, (14) implies

$$(15) \quad h = \Psi(j + wm).$$

By property (P1) of Ψ , it follows from (15) that $h = \Psi(j) - wm$, which, combined with (14) implies

$$(16) \quad \Psi(h) + h = \Psi(j) + j.$$

This contradicts (P2), since we have $0 \leq h, j \leq m-1$, and the set

$$\{i + \Psi(i) | 0 \leq i \leq m-1\}$$

must contain exactly m elements. Hence, we cannot have $j \neq h$. Assume $j = h$ and $w > 0$. Since $w < 2^k = n/m$, we must have $m < n$, and n must be even. Equation (14) can now be rewritten as

$$(17) \quad \Psi(h) + h = 2h + wm, \quad 0 < w < 2^k.$$

From (17) and property (P2), it follows that $2h + wm = 2i \pmod n$ for some $0 \leq i \leq m-1$. Since n is even and m is odd, w must be even, and, therefore, we must have $2 \leq w \leq 2^k - 2$. But then, since $0 \leq h \leq m-1$, it follows from (17) that $2m \leq \Psi(h) + h < n$, which, again, contradicts property (P2). Therefore, we must have $\Psi = \Psi_0$.

(iii) This part follows directly from parts (i) and (ii), since B_n consists of Ψ_0 and possibly other permutations in pairs of distinct inverses. $\square$

Let $\Lambda = (\lambda_0, \lambda_1, \dots, \lambda_{n-1})$ be a vector of integers in the range $0 \leq \lambda_i < q$, with at most one λ_i equal to zero. We say that Λ is m -periodic if $\lambda_{i+m} = \lambda_i$, $0 \leq i \leq n-1$. In the following discussion, we assume that such an m -periodic vector Λ is given. For an arbitrary permutation $\sigma \in S_m$, let

$$(18) \quad s(\sigma) = \sum_{i=0}^{n-1} \lambda_i q^{2\sigma(i \bmod m)}.$$

Consider the set of congruences

$$(19) \quad \sum_{i=0}^{n-1} \lambda_i q^{i+\phi(i)} \equiv s(\sigma) \pmod{(q^n-1)},$$

for all $\sigma \in S_m$, with $\phi \in S_n$. Let Φ_σ denote the set of solutions ϕ of (19) for a given σ , and let $\Phi = \bigcup_{\sigma \in S_m} \Phi_\sigma$.

LEMMA 5. Consider the transformation $T: S_n \rightarrow S_n$ defined by

$$(20) \quad (\phi T)(i) = \phi(i+m) + m, \quad \phi \in S_n, \quad 0 \leq i \leq n-1.$$

If $\phi \in \Phi_\sigma$ for some σ , then $\phi T \in \Phi_\sigma$.

Proof. Since $\phi \in \Phi_\sigma$, ϕ satisfies (19). Substituting $i+m$ for i in the sum at the left-hand side of (19), and recalling that indices are taken modulo n , we obtain

$$(21) \quad \sum_{i=0}^{n-1} \lambda_{i+m} q^{i+m+\phi(i+m)} \equiv s(\sigma) \pmod{(q^n-1)}.$$

Using the definition of T , and recalling that $\lambda_{i+m} = \lambda_i$, $0 \leq i \leq n-1$, it follows from (21) that

$$(22) \quad \sum_{i=0}^{n-1} \lambda_i q^{i+(\phi T)(i)} \equiv s(\sigma) \pmod{(q^n-1)}.$$

Hence, $\phi T \in \Phi_\sigma$. $\square$

LEMMA 6. For any $\phi \in S_n$, let $G_\phi = \{\phi, \phi T, \phi T^2, \dots\}$. Then, $|G_\phi| = 2^h$ for some integer h , $0 \leq h \leq k$.

Proof. From the definition of T in (20), and from the fact that $n = 2^k m$, it follows that $\phi T^{2^k} = \phi$ for all $\phi \in S_n$. Therefore, by a standard group-theoretic argument, the least positive integer g_ϕ satisfying $\phi T^{g_\phi} = \phi$ must divide 2^k . Hence, $|G_\phi| = g_\phi = 2^h$ for some $0 \leq h \leq k$. $\square$

LEMMA 7. $|\Phi|$ is odd.

Proof. It follows from the result of Lemma 5 and from the arguments in the proof of Lemma 6 that the transformation T induces a partition of Φ into disjoint orbits of the form $G_\phi = \{\phi, \phi T, \phi T^2, \dots, \phi T^{g_\phi-1}\}$. If $\phi T \neq \phi$, then by Lemma 6 g_ϕ is a nontrivial power of two, and G_ϕ contributes an even number of permutations to Φ . Hence, it remains to show that there is an odd number of permutations ϕ in Φ that satisfy $\phi T = \phi$ (i.e., $g_\phi = 1$). We claim that the set of all such ϕ is precisely B_n . Since by Lemma 4(iii) $|B_n|$ is odd, this would suffice to prove Lemma 7. To prove the claim, we first notice that if $\phi T = \phi$ then ϕ satisfies property (P1). Hence, $\phi(i) + i = \phi(i+m) + (i+m)$, and, since $\lambda_i = \lambda_{i+m}$, $0 \leq i \leq n-1$, we have

$$(23) \quad \sum_{i=0}^{n-1} \lambda_i q^{i+\phi(i)} = 2^k \sum_{i=0}^{m-1} \lambda_i q^{i+\phi(i)},$$

and

$$(24) \quad s(\sigma) = \sum_{i=0}^{n-1} \lambda_i q^{2\sigma(i \bmod m)} = 2^k \sum_{i=0}^{m-1} \lambda_i q^{2\sigma(i)},$$

for all $\sigma \in S_m$. Now, since $\phi \in \Phi$, ϕ satisfies (19) for some σ . Using (23) and (24), and noting that 2^k has an inverse modulo $q^n - 1$, we can rewrite (19) as

$$(25) \quad \sum_{i=0}^{m-1} \lambda_i q^{i+\phi(i)} = \sum_{i=0}^{m-1} \lambda_i q^{2\sigma(i)} \pmod{(q^n-1)}.$$

If $\lambda_i \neq 0$ for all $0 \leq i \leq m-1$, then by Lemma 3, (25) implies that

$$(26) \quad \{i + \phi(i) | 0 \leq i \leq m-1\} = \{0, 2, \dots, 2(m-1)\}.$$

Thus, ϕ satisfies (P2), and, hence, $\phi \in B_n$. If $\lambda_a = 0$ for some $0 \leq a \leq m-1$ then we must have $m = n$, since Λ is m -periodic and at most one element of Λ can be zero. Therefore, n must be odd. Let $b = 2\sigma(a) \bmod n$. Then, it follows from (25) and Lemma 3 that

$$(27) \quad \{i + \phi(i) | 0 \leq i \leq n-1, i \neq a\} = \{0, 1, \dots, n-1\} - \{b\}.$$

We claim that $a + \phi(a) = b$. This follows immediately from (27) and from the fact that, for odd n , we have $\sum_{i=0}^{n-1} i = \sum_{i=0}^{n-1} \phi(i) = \sum_{i=0}^{n-1} (i + \phi(i)) = 0 \bmod n$. Hence, (26) also holds in this case, and $\phi \in B_n$. On the other hand, if $\Psi \in B_n$, then by (P1), $\Psi T = \Psi$, and by (P2), $\Psi \in \Phi_\sigma$, with σ defined by

$$(28) \quad \sigma(i) = \frac{i + \Psi(i)}{2}, \quad 0 \leq i \leq m-1.$$

Hence, $\Psi \in \Phi$, and the claim is established. $\square$

The following theorem presents the main result of this section.

THEOREM 3. *Let $q = 2^t$, and let $n = 2^k m$ with $2 \nmid m$. Let $(\lambda_0, \lambda_1, \dots, \lambda_{n-1})$ be an m -periodic vector of integers in the range $0 \leq \lambda_i < q$, with at most one λ_i equal to zero. Then, there exists an element $\alpha \in K$ such that $\alpha^{\lambda_0}, \alpha^{\lambda_1 q}, \alpha^{\lambda_2 q^2}, \dots, \alpha^{\lambda_{n-1} q^{n-1}}$ form a basis of $GF(q^n)$ over $GF(q)$.*

Proof. Consider the set of congruences (19), for all $\sigma \in S_m$. By Lemma 7, the total number of solutions ϕ to (19) is odd. Hence, for some $\sigma_0 \in S_m$, $|\Phi_{\sigma_0}|$ must be odd. Let $r_i = \lambda_i q^i$, $0 \leq i \leq n-1$. Then, the number of solutions ϕ to

$$(29) \quad \sum_{i=0}^{n-1} r_i q^{\phi(i)} \equiv s(\sigma_0) \pmod{q^n - 1}$$

is odd. Therefore,

$$(30) \quad N_e(s(\sigma_0)) + N_o(s(\sigma_0)) \equiv 1 \pmod{2},$$

and

$$(31) \quad N_e(s(\sigma_0)) \not\equiv N_o(s(\sigma_0)) \pmod{2}.$$

By Theorem 2, this implies that $\{\lambda_0, \lambda_1 q, \dots, \lambda_{n-1} q^{n-1}\}$ is satisfiable. $\square$

Theorem 3 is strongest when n is odd (i.e., $m = n$). In this case, the m -periodicity restriction on $(\lambda_0, \lambda_1, \dots, \lambda_{n-1})$ is removed, and we obtain the following full-fledged generalization of the normal basis theorem.

COROLLARY. *Let $q = 2^t$, and let n be an odd positive integer. Let $\lambda_0, \lambda_1, \dots, \lambda_{n-1}$ be any integers in the range $0 \leq \lambda_i < q$, with at most one λ_i equal to zero. Then, there exists an element $\alpha \in K$ such that $\alpha^{\lambda_0}, \alpha^{\lambda_1 q}, \alpha^{\lambda_2 q^2}, \dots, \alpha^{\lambda_{n-1} q^{n-1}}$ form a basis of $GF(q^n)$ over $GF(q)$.*

On the other hand, when $n = 2^k$, the m -periodicity requirement implies that $1 \leq \lambda_0 = \lambda_1 = \dots = \lambda_{n-1} < q$. In this case, Theorem 3 is weakest, but it still generalizes the normal basis theorem.

4. Conclusion. We have presented a combinatorial characterization of sets of integers $\{r_0, r_1, \dots, r_{n-1}\}$ such that $\alpha^{r_0}, \alpha^{r_1}, \dots, \alpha^{r_{n-1}}$ form a basis of $GF(q^n)$ over $GF(q)$ for some $\alpha \in GF(q^n)$. Although the characterization of such sets was applied in this paper specifically to the generalization of the normal basis theorem, it has also arisen in other

contexts, namely in the study of the use of maximum-length shift register sequences (*M-sequences*) for testing of logic circuits [7]. In that context, it is interesting to determine whether there exist sets $\{r_0, r_1, \dots, r_{n-1}\}$ that are not satisfiable. For example, $\{0, 2, 5, 7, 9\}$ is such a set for $q = 2$ and $n = 5$ [7]. These sets are not “testable” by *M-sequences* with characteristic polynomials of degree n . The existence of such sets for arbitrary q and n is an open question when n is prime (the answer is known to be positive when n is composite [7]).

We have generalized the normal basis theorem for finite fields of characteristic two. Our proof is valid for all choices $\lambda_0, \lambda_1, \dots, \lambda_{n-1}$ in the range $0 \leq \lambda_i < q$ (with at most one λ_i equal to zero) when n is odd, and for more restricted choices of the λ_i when n is even. We conjecture that the generalized normal basis theorem holds, without additional restrictions on the values λ_i , for all values of q and n . For odd values of n , we can establish that the conjecture is “almost always” true in the following sense: given n , the conjecture is true for all but a finite number of values p of the field characteristic. This follows by observing that Theorem 2, and the lemmas leading to Theorem 3 and its corollary, are valid for fields of any characteristic. In the proof of Theorem 3, we establish the existence of an integer $s = s(\sigma_0)$ such that $N_e(s) - N_o(s)$ is odd. This suffices to prove the theorem when $p = 2$. For $p \neq 2$, we observe that $N_e(s) - N_o(s)$ being odd, it must be nonzero, and upperbounded in absolute value by $n!$. Hence, there is only a finite number of potential prime divisors of the difference, and the theorem holds for all other primes.

Appendix.

Proof of Lemma 3. Let $A = \sum_{i=0}^{m-1} \lambda_i q^{k_i}$, and $B = \sum_{i=0}^{m-1} \lambda_i q^{j_i}$. By the conditions at (8) and (10), we have $m < B \leq q^n - 1$. If all the k_i are distinct, then we also have $m < A \leq q^n - 1$, and congruence (11) implies $A = B$. By the uniqueness of radix- q representation of integers, we must have $\{k_0, k_1, \dots, k_{m-1}\} = \{j_0, j_1, \dots, j_{m-1}\}$.

Now assume that the k_i are not all distinct. Apply the following symbolic simplification procedure to the sum $\sum_{i=0}^{m-1} \lambda_i q^{k_i}$.

Step 1. Let a and b be indices such that $k_a = k_b$, $0 \leq a < b \leq m-1$. Do the following substitution: if $\lambda_a + \lambda_b < q$, replace $\lambda_a q^{k_a} + \lambda_b q^{k_b}$ in the sum by the single summand $\lambda'_a q^{k'_a}$, where $\lambda'_a = (\lambda_a + \lambda_b)$ and $k'_a = k_a$ (this decreases the number of summands by one). If $\lambda_a + \lambda_b \geq q$, replace $\lambda_a q^{k_a} + \lambda_b q^{k_b}$ in the sum by $\lambda'_a q^{k'_a} + \lambda'_b q^{k'_b}$, where

$$(A1) \quad \lambda'_a = (\lambda_a + \lambda_b) \mod q,$$

$$(A2) \quad \lambda'_b = \left\lceil \frac{\lambda_a + \lambda_b}{q} \right\rceil,$$

$$(A3) \quad k'_a = k_a,$$

$$(A4) \quad k'_b = (k_a + 1) \mod n.$$

(Notice that $\lambda'_a + \lambda'_b < \lambda_a + \lambda_b$).

Step 2. Step 1 produces a new sum $\sum_{i=0}^{m'-1} \lambda'_i q^{k'_i}$. If all k'_i are distinct, **stop**. Else, **go to Step 1**.

After executing Step 1, we have

$$(A5) \quad m' + \sum_{i=0}^{m'-1} \lambda'_i < m + \sum_{i=0}^{m-1} \lambda_i.$$

Hence, Step 1 can only be executed a finite number of times, and the procedure eventually stops with all k'_i distinct, and the new sum satisfying

$$(A6) \quad 0 < \lambda'_0, \lambda'_1, \dots, \lambda'_{m'-1} < q,$$

$$(A7) \quad 0 \leq k'_0, k'_1, \dots, k'_{m'-1} \leq n-1,$$

and

$$(A8) \quad \sum_{i=0}^{m'-1} \lambda'_i q^{k'_i} \equiv \sum_{i=0}^{m-1} \lambda_i q^{k_i} \pmod{(q^n-1)}.$$

Together with congruence (11), this implies

$$(A9) \quad \sum_{i=0}^{m'-1} \lambda'_i q^{k'_i} \equiv \sum_{i=0}^{m-1} \lambda_i q^{j_i} \pmod{(q^n-1)}.$$

Since all k'_i are distinct, uniqueness of radix- q representation applies again, and we must have $m' = m$, $\{k'_0, k'_1, \dots, k'_{m'-1}\} = \{j_0, j_1, \dots, j_{m-1}\}$, and $\{\lambda'_0, \lambda'_1, \dots, \lambda'_{m'-1}\} = \{\lambda_0, \lambda_1, \dots, \lambda_{m-1}\}$ (by an abuse of notation, the latter set equality allows for element multiplicity). This contradicts (A5), implying that Step 1 of the simplification procedure could not have been applied and that the original k_i were all distinct. $\square$

REFERENCES

- [1] R. LIDL AND H. NIEDERREITER, *Finite Fields*, Addison-Wesley, Reading, MA, 1983.
- [2] F. J. MACWILLIAMS AND N. J. A. SLOANE, *The Theory of Error-Correcting Codes*, North-Holland, New York, 1977.
- [3] J. L. MASSEY AND J. K. OMURA, *Computational method and apparatus for finite field arithmetic*, U.S. patent application, 1981.
- [4] R. C. MULLIN, I. M. ONYSZCHUK, S. A. VANSTONE, AND R. M. WILSON, *Optimal normal bases in $GF(p^n)$* , Discrete Applied Math., 22 (1988/89), pp. 149–161.
- [5] D. W. ASH, I. F. BLAKE, AND S. A. VANSTONE, *Low complexity normal bases*, Research report CORR 86-21, Faculty of Mathematics, University of Waterloo, November 1986.
- [6] A. LEMPEL AND M. J. WEINBERGER, *Self-complementary normal bases in finite fields*, SIAM J. Discrete Math., 1 (1988), pp. 193–198.
- [7] A. LEMPEL AND M. COHN, *Design of universal test sequences for VLSI*, IEEE Trans. Inform. Theory, IT-31 (1985), pp. 10–17.

SHRINKING LATTICE POLYHEDRA*

JOHN CREMONA† AND SUSAN LANDAU‡

Abstract. This paper treats the following geometric problem: Given vertices $x_1, \dots, x_n$ of a polyhedron in the integer lattice in k dimensions, can that polyhedron be “shrunk” to a similar one (i.e., one whose sides remain in the same ratio as the original polyhedron) while still remaining on the integer lattice? A necessary condition is given for this to be possible, which depends on the parity of k ; for $k \leq 4$ it is shown that the condition is sufficient, and algorithms are also given to do the shrinking. In two dimensions, the algorithm only involves computing greatest common divisors over the Gaussian integers and is polynomial time. For $k = 3$ and 4 the algorithms involve computing g.c.d.s in the algebra of Hurwitz quaternions. This gives a polynomial time algorithm for $k = 4$, but because the algorithm in three dimensions relies on determining the square factors of an integer, it is at present exponential. The proofs are remarkably simple and are quite computational in nature.

Key words. shrinking lattice polyhedra

AMS(MOS) subject classifications. 51

1. Introduction. This paper treats the following geometric problem: Given vertices $x_1, \dots, x_n$ of a polyhedron in the integer lattice in k dimensions, can that polyhedron be “shrunk” to a similar one (i.e., one whose sides remain in the same ratio as the original polyhedron) while still remaining on the integer lattice?

We give a necessary condition for this to be possible, which depends on the parity of k ; for $k \leq 4$ we show that the condition is sufficient, and we also give algorithms to do the shrinking. For $k \geq 7$ the condition is not sufficient; we give counterexamples in 7 and 8 dimensions. For $k = 5$ and 6 the sufficiency of the condition is still open.

In two dimensions, the algorithm only involves computing greatest common divisors (g.c.d.s) over the Gaussian integers and is polynomial time. For $k = 3$ and 4 the algorithms involve computing g.c.d.s in the algebra of Hurwitz quaternions. This gives a polynomial-time algorithm for $k = 4$, but because the algorithm in three dimensions relies on determining the square factors of an integer, it is at present exponential. The proofs are remarkably simple and are quite computational in nature.

The idea behind the algorithms is that there is a natural correspondence between the points of the integer lattice and elements of a certain algebraic structure: the Gaussian integers in the two-dimensional case, Hurwitz quaternions in the three- and four-dimensional cases. Then the shrinking problem can be viewed algebraically. The solutions given rely on the existence of Euclidean algorithms in the rings of Gaussian integers and Hurwitz quaternions, respectively.

In § 2 we make some preliminary geometrical observations common to all dimensions and derive a necessary condition for shrinking to be possible. This depends, somewhat surprisingly, on the parity of the dimension k . Then in the last three sections we present the solution for the two-, three-, and four-dimensional cases, respectively.

2. Geometrical preliminaries. A general reference for this section is [8, §§ 65, 70].

A *similarity* of $\mathbf{R}^k$ is a linear map $T: \mathbf{R}^k \rightarrow \mathbf{R}^k$ that preserves angles. If T has matrix M (with respect to the standard basis of $\mathbf{R}^k$), this is equivalent to $M^t M = \lambda I$ for some nonzero scalar λ . Moreover, λ is necessarily positive, since for any $x \in \mathbf{R}^k$ with $Mx \neq 0$,

* Received by the editors November 30, 1987; accepted for publication (in revised form) July 20, 1989.

† Department of Mathematics, University of Exeter, Exeter EX4 4QE, United Kingdom.

‡ Department of Mathematics, Wesleyan University, Middletown, Connecticut 06457. This research was supported in part by an MSRI fellowship and National Science Foundation grant DCR-8402175.

we have $|Mx|^2 = \lambda|x|^2$, so that $\lambda = |Mx|^2/|x|^2$. The (positive) quantity $\sqrt{\lambda}$ is the *scale factor* of T (or of M), and T is the product of the orthogonal transformation $(1/\sqrt{\lambda})M$ and a scaling by $\sqrt{\lambda}$. A similarity is *direct* or orientation-preserving if it has positive determinant.

As with any linear map, a similarity T is determined by the images of any k linearly independent points in $\mathbf{R}^k$. If these points and their images all have rational coordinates, then clearly the matrix M of T has rational entries. When k is odd, this implies that $\sqrt{\lambda}$ is also rational.

LEMMA 1. *Let k be odd, and let M in $M_k(\mathbf{Q})$ satisfy $M'M = \lambda I$, with $\lambda \neq 0$. Then $\lambda = \mu^2$ for some μ in $\mathbf{Q}$ with $\mu^k = \det M$. Hence $\mu^{-1}M$ is in $\mathbf{SO}(k, \mathbf{Q})$, the group of $k \times k$ orthogonal matrices over $\mathbf{Q}$ with determinant 1.*

Proof. We have $(\det M)^2 = \lambda^k$, so $|\det M| = \lambda^{k/2}$. Since $\det M$ is rational and k is odd, this implies that $\mu = \sqrt{\lambda}$ is rational, and $\mu^k = |\det M|$. Replacing μ by $-\mu$ if necessary we can ensure that $\mu^k = \det M$ again, since k is odd. $\square$

A general angle-preserving map $\mathbf{R}^k \rightarrow \mathbf{R}^k$ consists of a translation followed by a similarity, and is determined by the images of any $k+1$ affinely independent points (that is, of any $k+1$ points which do not all lie on a hyperplane of dimension $k-1$).

Our strategy will be as follows: given n points $x_1, \dots, x_n$ with integer coordinates in $\mathbf{R}^k$, we must find an angle-preserving transformation R of $\mathbf{R}^k$ with minimal scale factor such that each Rx_i again has integer coordinates. First assume that zero (the origin) is one of the given points. Then without loss of generality we may assume that $R(0) = 0$, so that R is a similarity; again without loss, we may presume R to be direct. Having found the solution to this problem of finding the best direct similarity, we must then determine the translation with which to precede such a similarity in order to achieve the best shrinkage. For this it is always optimal to translate so that one of the given points is at the origin: since similarities preserve 0, for any other choice we are effectively including zero as an extra $n+1$ st point in the given configuration, which could not give a better shrinkage. Hence, we will assume from now on that the origin is one of the given points, and that the transformation sought is a direct similarity.

Since the transformations we seek are linear (after the initial translation), we could replace the given points in $\mathbf{Z}^k$ with a set of (at most) k points that generate the same subgroup of the lattice $\mathbf{Z}^k$, thus reducing the problem to one involving at most k points in k dimensions. This shows that the smallest scale factor is not a function of the particular set of points given, but of the subgroup of $\mathbf{Z}^k$ which they generate (assuming that zero is in the set). So properties of the particular geometrical configuration, such as convexity, are irrelevant in the solution.

For vectors x and y in $\mathbf{R}^k$ we denote the usual scalar product of x and y by $x \cdot y$, and (when $k=3$) their vector or cross product, by $x \times y$. Also, for a positive integer g the *square part* of g is the largest integer whose square divides g (for example, the square part of 12 is 2). A number is said to be *square-free* if its square part is 1.

From the defining equation of a similarity we see that $(Rx) \cdot (Ry) = \lambda x \cdot y$ for all vectors x and y in $\mathbf{R}^k$. If for $1 \leq i \leq n$ we have both $x_i \in \mathbf{Z}^k$ and $Rx_i \in \mathbf{Z}^k$, it follows that λ is a rational number whose denominator divides each of the scalar products $x_i \cdot x_j$. Moreover, when k is odd this denominator is a square, by Lemma 1. This gives the following necessary condition for a scaling by a factor λ to be possible, and a lower bound on the possible shrinkage of a given configuration.

LEMMA 2. *Let $X = \{x_1, x_2, \dots, x_n\} \subset \mathbf{Z}^k$. Set $g = \gcd_{\mathbf{Z}}(\{x \cdot y \mid x, y \in X\})$. Let R be a similarity with scale factor $\sqrt{\lambda}$ such that $R(X) \subset \mathbf{Z}^k$; write $\lambda = a/b$ with $\gcd(a, b) = 1$. Then b divides g , and, moreover, if k is odd then b is a square divisor of g . Hence the smallest such λ satisfies $\sqrt{\lambda} \geq 1/d$, where $d = \sqrt{g}$ when k is even, and d is*

the square part of g when k is odd. In particular, no shrinkage of X is possible if $g = 1$, or if k is odd and g is square-free.

A question similar to that of shrinking a lattice is the “integrability” of lattices, as discussed in [1]. Given independent vectors $x_1, x_2, \dots, x_n \in \mathbf{R}^N$ (for some $N \geq n$) such that $x_i \cdot x_j \in \mathbf{Z}$ for all i, j , the integrability question asks whether there are vectors $y_1, y_2, \dots, y_n \in \mathbf{Z}^k$ for some $k \geq n$ such that $y_i \cdot y_j = x_i \cdot x_j$ for all i, j . In [1] it is shown that for $n \leq 5$ this holds with $k \leq n + 3$, but that it does not hold in general for $n \geq 6$. This result is originally due to Ko and Mordell in 1937; counterexamples for $n = 6, 7$, and 8 are the “root lattices” $E_n \subset \mathbf{R}^8$. We cannot apply these results directly to the shrinking question for two reasons: we wish to consider n -dimensional lattices given in $\mathbf{Z}^n$ itself (i.e., $n = N$), and we require the shrinking to take place without enlarging the ambient dimension (i.e., $k = n$). For $n = 5$ and 6 the question of the sufficiency of our criterion remains open. On the other hand for $n = 7$ and 8 the root lattice E_n can be given as a lattice in $\mathbf{R}^n$ with coordinates which are all either integers or half-integers, but not as a sublattice of $\mathbf{Z}^n$. Thus $L = 2E_n$ is a sublattice of $\mathbf{Z}^n$ with the property that $x \cdot y \equiv 0 \pmod{4}$ for all $x, y \in L$, which cannot be shrunk by a factor of 2 in $\mathbf{Z}^n$.

In the following sections we will show that for $k \leq 4$ the necessary condition is sufficient and the lower bound is actually attained. We sum this up in the following theorem.

THEOREM 3. *Let $X = \{x_1, x_2, \dots, x_n\} \subset \mathbf{Z}^k$, where $k \leq 4$. Set*

$$g = \gcd_{\mathbf{Z}}(\{x \cdot y \mid x, y \in X\})$$

and let d be the square part of g . Then there exists a similarity R with $R(X) \subset \mathbf{Z}^k$ and scale factor $1/\sqrt{g}$ (when k is even) or $1/d$ (when k is odd). This achieves the best possible shrinkage of the given configuration X .

For completeness, here we give the proof for $k = 1$. The idea is essentially the same as in the higher dimensional cases, but is almost trivial. We are given integers $x_1, x_2, \dots, x_n$ and we must find the smallest positive real number r such that each rx_i is again integral. If $d = \gcd_{\mathbf{Z}}(x_1, \dots, x_n)$, this is equivalent to the single condition $rd \in \mathbf{Z}$, so the smallest such r is clearly $r = 1/d$. The map $x \mapsto rx$ is a direct similarity with scale factor $1/d$. This proves the theorem for the case $k = 1$, since it is clear that $g = \gcd(\{x_i x_j\}) = d^2$. To shrink a one-dimensional configuration $X = \{x_1, x_2, \dots, x_n\} \subset \mathbf{Z} \subset \mathbf{R}$ (not necessarily including 0) we therefore compute

$$d = \gcd_{\mathbf{Z}}(x_2 - x_1, \dots, x_n - x_1)$$

and map $x_i \mapsto (x_i - x_1)/d$ for $1 \leq i \leq n$. The computation is dominated by the single g.c.d. computation which takes $O(n \log N)$ steps where

$$N = \max(|x_1|, |x_2|, \dots, |x_n|).$$

Next we turn to the two-dimensional case. Here it turns out that the solution is almost identical to that for the one-dimensional case just given, replacing the integers by the ring of Gaussian integers, which also has a Euclidean algorithm.

3. The two-dimensional case. We will use the fact that the set

$$\mathbf{Z}[i] = \{a + bi \mid a, b \in \mathbf{Z}\}$$

of Gaussian integers forms a commutative ring with a Euclidean algorithm; thus any two Gaussian integers α, β have a greatest common divisor that is well defined up to multiplication by ± 1 or $\pm i$. We define $\gcd_{\mathbf{Z}[i]}(\alpha, \beta)$ to be this g.c.d., normalized to lie in the first quadrant; thus $\gcd_{\mathbf{Z}[i]}(\alpha, \beta) = 1$ if and only if α and β have no nontrivial common factor.

Identify a point (x, y) in $\mathbf{R}^2$ with the complex number $z = x + iy$, where $i = \sqrt{-1}$. Points with integer coordinates then correspond to Gaussian integers, and points with rational coordinates to elements of the field $\mathbf{Q}(i)$. The n given integer points $x_1, \dots, x_n$ thus correspond to n Gaussian integers $\alpha_1, \dots, \alpha_n$.

Our main result is the following theorem.

THEOREM 4. *The polygon defined by $x_1, \dots, x_n$ can be shrunk if and only if $\gcd_{\mathbf{Z}[i]}(\alpha_2 - \alpha_1, \dots, \alpha_n - \alpha_1) \neq 1$. The maximal shrinkage factor can be computed in $O(n \log N)$ steps, where $N = \max(|\alpha_1|, \dots, |\alpha_n|)$.*

The idea of the proof is as follows: viewing the points x_i as elements of the ring of Gaussian integers, any shrinkage will be achieved by dividing the points by some fixed Gaussian integer. We can thus find the maximal shrinkage by computing the g.c.d. of the given points, after first shifting the origin. We prove this by a succession of lemmas.

LEMMA 5. *Let $T: \mathbf{R}^2 \rightarrow \mathbf{R}^2$ be a direct similarity. Then T has matrix $\begin{pmatrix} x & -y \\ y & x \end{pmatrix}$ with x, y in $\mathbf{R}$ and $x^2 + y^2 \neq 0$.*

Proof. The matrix M of T satisfies $M^t M = \text{scalar}$ and $\det M > 0$, by the definitions in the previous section. An easy computation shows that M must then have the given form. $\square$

LEMMA 6. *The set $\{\begin{pmatrix} x & -y \\ y & x \end{pmatrix} \mid x, y \in \mathbf{Q}\}$ of matrices forms a field isomorphic to $\mathbf{Q}(i)$ via the mapping $\begin{pmatrix} x & -y \\ y & x \end{pmatrix} \mapsto x + yi$.*

Proof. The proof is a standard exercise. $\square$

Any fixed nonzero complex number α determines a transformation T_α of $\mathbf{R}^2$, namely $z \mapsto \alpha z$ (identifying $\mathbf{R}^2$ with $\mathbf{C}$ as always). If $\alpha = x + iy$ then T_α has matrix $\begin{pmatrix} x & -y \\ y & x \end{pmatrix}$, so that T_α is a direct similarity with scale factor $|\alpha| = \sqrt{x^2 + y^2}$. Conversely, any direct similarity has the form T_α by Lemma 5. Moreover, it is clear that T_α preserves the rational points $\mathbf{Q}^2 \subset \mathbf{R}^2$ if and only if x and y are rational, i.e., if and only if $\alpha \in \mathbf{Q}(i)$.

LEMMA 7. *Let $\alpha_1, \dots, \alpha_n$ be a finite set of points in $\mathbf{Z}^2$ (identified with $\mathbf{Z}[i]$). The similarity T_α takes each α_j to an integral point if and only if α is in the fractional ideal of $\mathbf{Q}(i)$ inverse to the ideal generated by $\alpha_1, \dots, \alpha_n$.*

Proof. We require $\alpha\alpha_j \in \mathbf{Z}[i]$ for $j = 1, \dots, n$. This is equivalent to $\alpha A \subseteq \mathbf{Z}[i]$, where A is the ideal $(\alpha_1, \dots, \alpha_n)$, which in turn is equivalent to $\alpha \in A^{-1}$. $\square$

For the best shrinkage we must find $\alpha \in A^{-1}$ with minimal norm: a generator of A^{-1} achieves this. Write $A = (\gamma)$ where $\gamma = \gcd_{\mathbf{Z}[i]}(\alpha_1, \dots, \alpha_n)$; then $A^{-1} = (\gamma)^{-1} = (\beta)$ with $\beta = \gamma^{-1}$. Now T_β is the best shrinkage, and it is trivial if and only if $|\beta| = 1$, which is equivalent to $\gcd_{\mathbf{Z}[i]}(\alpha_1, \dots, \alpha_n) = 1$.

Hence we have a solution to the two-dimensional shrinking problem. To find the best origin-preserving shrinkage, compute γ , the g.c.d. of the given points $\alpha_1, \dots, \alpha_n$ viewed as Gaussian integers, and then divide each α_j by γ . The best choice of origin is, as observed in the previous section, to place it at one of the given points. So the best shrinkage is finally obtained by shifting the origin to α_1 , computing γ as the g.c.d. of $\alpha_2 - \alpha_1, \dots, \alpha_n - \alpha_1$, and then dividing by γ .

We can now present the algorithm.

ALGORITHM. Shrink Polygons.

Input: $\{x_j = (a_j, b_j)\} *$ a set of n points which defines a polygon in $\mathbf{Z}^2 *$;

Output: $\{y_j = (a_j, b_j)\} *$ a set of n points which defines a similar polygon of minimum size in $\mathbf{Z}^2 *$;

- (1) BEGIN
- (2) FOR $j = 1$ TO n DO: $\alpha_j \leftarrow (a_j - a_1) + i(b_j - b_1)$;
- (3) $\gamma \leftarrow \gcd_{\mathbf{Z}[i]}(\alpha_2, \dots, \alpha_n)$;
- (4) FOR $j = 1$ TO n DO: $a_j + ib_j \leftarrow \alpha_j / \gamma$;
- (5) END.

All that remains to be shown is that the running time is $O(n \log N)$ steps, where N is the maximum norm of the α_j . But this is clear, since the running time of the algorithm is dominated by the g.c.d. calculation on line (3). By [7] this computation can be done in $O(n \log N)$ steps. This completes the proof of Theorem 4.

We note that we made no assumptions regarding convexity or holes; thus our algorithm works for all polygons in $\mathbf{Z}^2$.

Finally we must show that the shrinkage obtained by the algorithm agrees with that claimed in Theorem 3. For this it suffices to show that if for some prime number p we have $x_j \cdot x_k \in p\mathbf{Z}$ for all j, k , then $\gcd_{\mathbf{Z}[i]}(\alpha_1, \dots, \alpha_n) \neq 1$. Here as before we set $\alpha_j = a_j + ib_j$ where $x_j = (a_j, b_j)$.

If $p = 2$ the conditions imply that $a_j \equiv b_j \pmod{2}$ for all j , so that $1 + i$ divides each α_j .

If $p \equiv 3 \pmod{4}$ then $a^2 + b^2 \equiv 0 \pmod{p}$ implies that $a \equiv b \equiv 0 \pmod{p}$ so that the given conditions imply that each α_j is divisible by p .

If $p \equiv 1 \pmod{4}$ then $p = \pi \bar{\pi}$ for some $\pi \in \mathbf{Z}[i]$, and $N(\alpha_j) = \alpha_j \bar{\alpha}_j = a_j^2 + b_j^2 \equiv 0 \pmod{p}$ implies that each α_j is divisible either by π or by $\bar{\pi}$ (or by both). Finally the condition $a_j a_k + b_j b_k \equiv 0 \pmod{p}$ implies that each $\alpha_j \bar{\alpha}_k$ is divisible by p , so that at least one of π or $\bar{\pi}$ divides all of the α_j .

(Here we have been using well-known properties of the ring $\mathbf{Z}[i]$, including unique factorization: see [3, § 3.8] for details.)

4. The three-dimensional case. In three and four dimensions our algorithm will use quaternions, making use of both their geometrical and arithmetical properties.

Let $\mathbf{H} = \{x + yi + zj + tk \mid x, y, z, t \in \mathbf{R}\}$, the four-dimensional real algebra of quaternions. The quaternion $q = x + yi + zj + tk$ has a *conjugate* $\bar{q} = x - yi - zj - tk$, a *trace* $\text{Tr}(q) = q + \bar{q} = 2x$ and a *norm* $N(q) = q\bar{q} = x^2 + y^2 + z^2 + t^2$. Clearly $q = 0 \Leftrightarrow N(q) = 0$, and if $q \neq 0$ then q has an inverse $q^{-1} = N(q)^{-1}\bar{q}$; so $\mathbf{H}$ is a division algebra. We will identify $\mathbf{R}^3$ with the subspace V of $\mathbf{H}$ spanned by i, j , and k (the quaternions with zero trace), via $(x, y, z) \mapsto xi + yj + zk$. It is well known that the group of rotations of $\mathbf{R}^3$ can be parametrized by nonzero quaternions using the conjugation action (see [6, pp. 42–44]). Specifically, any nonzero quaternion q induces a transformation R_q from V to V , which is in fact a rotation: $v \mapsto qvq^{-1}$. In fact, every rotation of V arises in this way, although we will not need this.

The second important ingredient in the algorithm is the ring S of Hurwitz quaternions which lies in $\mathbf{H}$; it is spanned over $\mathbf{Z}$ by the four elements $\frac{1}{2}(1 + i + j + k)$, i, j , and k . We define

$$S_0 = S \cap V = \{ai + bj + ck \mid a, b, c \in \mathbf{Z}\},$$

so that under the above identification, S_0 is identified with $\mathbf{Z}^3$. The crucial fact for the efficiency of our algorithm is that the noncommutative ring S has a right-Euclidean algorithm. We summarise here for convenience the facts about quaternion arithmetic which we will use below, giving references to [2, § 91] for details.

D1: Every pair of elements α and β of S has a greatest common right divisor γ which is characterised by the properties that γ divides both α and β on the right, and that every common right divisor of α and β is a right divisor of γ . We denote the greatest common right divisor or g.c.r.d. of α and β by $\text{gcd}(\alpha, \beta)$. [2, § 91, Thm. 2].

D2: The g.c.r.d. is uniquely determined up to multiplication on the left by a unit (invertible element) of S . [2, § 91, Thm. 2].

D3: If $\gamma = \text{gcd}(\alpha, \beta)$ then there exist δ and η in S such that $\gamma = \delta\alpha + \eta\beta$. [2, § 91, Thm. 2].

D4: If the norm of $\alpha \in S$ is divisible by an integer $n > 1$ then $\gcd(\alpha, n)$ is not a unit. [2, § 91, Lemma 3].

D5: The g.c.r.d. of two quaternions in S may be computed by a Euclidean algorithm which uses a simple generalisation of the usual (integer) division algorithm to S . [2, § 91, Lemmas 1 and 2].

Similar properties hold for the greatest common left divisor, or g.c.l.d., which will be used in the next section.

As in the case of two dimensions, we are given a polyhedron with vertices $x_1, \dots, x_n$ in $\mathbb{Z}^3$, and we must find $y_1, \dots, y_n$ in $\mathbb{Z}^3$ which are the vertices of a “similar” polyhedron which is as small as possible. We may assume that the x_r do not all lie on a plane, otherwise the problem is degenerate.

We know from Lemma 2 that if shrinkage by a factor d is possible then d^2 must divide each of the numbers $x_r \cdot x_s$, and hence divide their greatest common divisor g . We now prove Theorem 3 for $k = 3$, which states that the necessary condition is in fact sufficient. For this it suffices to show that if g is divisible by p^2 for some prime $p \in \mathbb{Z}$, then we can shrink by the factor p . We therefore assume that $x_r \cdot x_s \equiv 0 \pmod{p^2}$ for all r, s , and we must find R in $\text{SO}(3, \mathbb{Q})$ such that $Rx_r \in (p\mathbb{Z})^3$ for $r = 1, \dots, n$; the required similarity is then $M = p^{-1}R$.

The trivial case occurs when each x_r is already in $(p\mathbb{Z})^3$, and then we may take $R = I$ (the 3×3 identity matrix) and $M = p^{-1}I$. This is necessarily the case when $p = 2$, since $a^2 + b^2 + c^2 \equiv 0 \pmod{4}$ implies that $a \equiv b \equiv c \equiv 0 \pmod{2}$.

We may now assume that p is odd. Then our problem may be translated into the language of quaternions as follows. The given points $x_r = (a_r, b_r, c_r) \in \mathbb{Z}^3$ correspond to quaternions $\alpha_r = a_r i + b_r j + c_r k \in S_0$, and we must find π in $\mathbb{H}^* = \mathbb{H} - \{0\}$ such that $\pi \alpha_r \pi^{-1} \in pS_0$ for all r . It will turn out that we may take π to be $\gcd(p, \alpha_1, \alpha_2, \dots, \alpha_n)$.

Let X be the $n \times 3$ matrix with x_r in row r . Our condition implies that $XX^t \equiv 0 \pmod{p}$. We may view X as the matrix of a linear map $F^3 \rightarrow F^n$ (where $F = \mathbb{Z}/p\mathbb{Z}$), and similarly $X^t: F^n \rightarrow F^3$. Now $XX^t \equiv 0 \pmod{p}$ implies that $\text{im}(X^t) \subseteq \ker(X)$; hence $\text{rk}_p(X^t) \leq 3 - \text{rk}_p(X)$ by elementary linear algebra, where $\text{rk}_p(\)$ denotes rank modulo p . Since $\text{rk}_p(X^t) = \text{rk}_p(X)$ it follows that $\text{rk}_p(X)$ is 0 or 1. The case where $\text{rk}_p X = 0$ is the trivial case which was handled earlier, so we may assume that $\text{rk}_p X = 1$.

From $\text{rk}_p(X) = 1$ it follows immediately that $x_r \times x_s \equiv 0 \pmod{p}$ for all r, s . Hence p divides $\alpha_r \alpha_s$ for all r and s , since the “real” part of $\alpha_r \alpha_s$ is $-x_r \cdot x_s$ while the “imaginary” or vector part is $x_r \times x_s$. In addition, we know that p^2 divides $N(\alpha_r)$ for all r , since $N(\alpha_r) = -x_r \cdot x_r$.

The desired result now follows from the following result concerning quaternions, which implies the existence of $\pi \in S$ with $N(\pi) = p$ such that each α_r is of the form $\bar{\pi} \beta_r \pi$ with $\beta_r \in S_0$. Then the rotation R_π (given by $\alpha \mapsto \pi \alpha \pi^{-1}$) sends each α_r to $\pi \alpha_r \pi^{-1} = \pi \bar{\pi} \beta_r \pi \pi^{-1} = p \beta_r$, as required.

PROPOSITION 8. *Let p be an odd prime.*

(a) *Let $\alpha \in S - pS$, and suppose that $p | N(\alpha)$. Then $\pi = \gcd(p, \alpha)$ satisfies $N(\pi) = p$.*

(b) *Let $\alpha \in S_0 - pS_0$, and suppose that $p^2 | N(\alpha)$. Set $\pi = \gcd(p, \alpha)$. Then $\alpha = \bar{\pi} \beta \pi$ for some $\beta \in S_0$.*

(c) *Suppose that α_1 and α_2 satisfy the conditions of part (b), and also that $p | \alpha_1 \alpha_2$. Set $\pi = \gcd(p, \alpha_1)$. Then there exist $\beta_1, \beta_2 \in S_0$ such that $\alpha_r = \bar{\pi} \beta_r \pi$ for $r = 1, 2$.*

Proof. (a) By **D4** we have $N(\pi) > 1$. Writing $p = \eta \pi$ with $\eta \in S$ we then have $N(\eta) = 1$ or p ; but if $N(\eta) = 1$ then α is divisible by p , giving a contradiction.

(b) By part (a) we have $\alpha = \gamma\pi$ with $\gamma \in S$. Applying part (a) again to $\bar{\gamma}$ we also have $\bar{\gamma} = \delta\pi_1$ with $\pi_1, \delta \in S$ and $N(\pi_1) = p$. Then on the one hand $\alpha = \bar{\pi}_1\bar{\delta}\pi$; on the other hand, $\alpha = -\bar{\alpha} = -\bar{\pi}\delta\pi_1$. The uniqueness property **D2** now gives $\pi_1 = \eta\pi$ for some unit $\eta \in S$, and hence $\alpha = \bar{\pi}(\bar{\eta}\bar{\delta})\pi$. This gives the desired result with $\beta = \bar{\eta}\bar{\delta}$: the fact that β has trace zero follows from the relation $p\beta = \pi\alpha\pi^{-1}$.

(c) Applying part (b) separately to α_1 and α_2 , we have $\alpha_r = \bar{\pi}_r\beta_r\pi_r$ with $\beta_r \in S$ and $N(\pi_r) = p$, where $\pi_r = \gcd(p, \alpha_r)$, for $r = 1, 2$. By **D3** there exist $\xi_r, \eta_r \in S$ such that $\xi_r\alpha_r + \eta_rp = \pi_r$ for $r = 1, 2$. Then

$$\pi_1\bar{\pi}_2 \equiv (\xi_1\alpha_1)(\bar{\xi}_2\bar{\alpha}_2) = \xi_1\alpha_1\bar{\alpha}_2\bar{\xi}_2 = -\xi_1\alpha_1\alpha_2\bar{\xi}_2 \equiv 0 \pmod{p}$$

so that $\pi_1\bar{\pi}_2 = \delta p$ for some $\delta \in S$ with $N(\delta) = 1$. Since $p = \pi_2\bar{\pi}_2$ this gives $\pi_1 = \delta\pi_2$, and so $\alpha_2 = \bar{\pi}_2\beta_2\pi_2 = \bar{\pi}_1(\bar{\delta}\beta_2\bar{\delta})\pi_1$. Hence the desired result with $\pi = \pi_1$, on replacing β_2 by $\bar{\delta}\beta_2\bar{\delta}$. $\square$

As well as proving Theorem 3 for $k = 3$, we have now shown how to go about shrinking by a prime p whose square divides the g.c.d. of the scalar products of the x_r . We pick a vector (a, b, c) in our list which is not trivially a multiple of p , set $\alpha = ai + bj + ck$, and compute $\pi = \gcd(p, \alpha)$. Then the rotation R_π defined by $\alpha \mapsto \pi\alpha\pi^{-1}$ maps each x_r into $p\mathbf{Z}^3$, after which the factor of p can be divided out.

However, in practice we need not shrink by one prime at a time, as the next result shows.

PROPOSITION 9. *Let $x_1, x_2, \dots, x_n \in \mathbf{Z}^3$. Write $x_r = (a_r, b_r, c_r)$ for $r = 1, 2, \dots, n$. Suppose that*

- (1) $x_r \cdot x_s \in d^2\mathbf{Z}$ for $1 \leq r \leq s \leq n$ and some odd integer d ;
- (2) $\gcd(a_1, b_1, c_1, a_2, \dots, c_n) = 1$.

Set $\alpha_r = a_r i + b_r j + c_r k$ and $\pi = \gcd(d, \alpha_1, \alpha_2, \dots, \alpha_n)$. Then $N(\pi) = d$, and each α_r has the form $\bar{\pi}\beta_r\pi$ with $\beta_r \in S_0$, so that $R_\pi(x_r) \in (d\mathbf{Z})^3$ for $r = 1, 2, \dots, n$.

Proof. Write d as a product of (not necessarily distinct) odd primes, say $d = p_1 p_2 \cdots p_k$. We prove the Proposition by induction on k . The case $k = 1$ (so that d is an odd prime) was handled above, so suppose that $k \geq 2$ and the result holds for $k - 1$.

The hypotheses clearly still hold if we replace d by $d_0 = d/p_k$, which is a product of $k - 1$ primes, so by induction we have $\alpha_r = \bar{\pi}_0\gamma_r\pi_0$ with $\gamma_r \in S_0$ and $N(\pi_0) = d_0$. Write $\gamma_r = a'_r i + b'_r j + c'_r k$ and set $y_r = (a'_r, b'_r, c'_r)$. Then $R_{\pi_0}(x_r) = d_0 y_r$; since R_{π_0} preserves the scalar product it follows that p_k^2 divides each $y_r \cdot y_s$. Second, if $g = \gcd(a'_1, b'_1, c'_1, a'_2, \dots, c'_n) > 1$, then g would divide each γ_r and hence each α_r , contradicting condition (2). It follows that the conditions of the Proposition are satisfied with y_r and p_k in place of x_r and d ; we may apply the induction hypothesis again (with $k = 1$) to obtain $\gamma_r = \bar{\pi}_k\beta_r\pi_k$ with $\beta_r \in S_0$ and $N(\pi_k) = p_k$. Setting $\pi = \pi_k\pi_0$ we have, for each r , $\alpha_r = \bar{\pi}\beta_r\pi$ and $N(\pi) = d_0 p_k = d$.

To complete the proof we must show that π is indeed a g.c.r.d. of d and $\alpha_1, \alpha_2, \dots, \alpha_n$. Certainly π is a common right divisor of these, since $\alpha_r = \bar{\pi}\beta_r\pi$ and $d = N(\pi) = \bar{\pi}\pi$. Thus if $\delta = \gcd(d, \alpha_1, \alpha_2, \dots, \alpha_n)$ then $\delta = \eta\pi$ for some $\eta \in S$, and we must show that $N(\eta) = 1$.

Since δ is a right divisor of d we have $d = \theta\delta$ for some $\theta \in S$. Then $\bar{\pi}\pi = d = \theta\eta\pi$ so that $\theta\eta = \bar{\pi}$, or $\pi = \bar{\eta}\bar{\theta}$. Now $\delta = \eta\pi = N(\eta)\bar{\theta}$ is a right divisor of each α_r , from which it follows that $N(\eta)$ is an integer which divides each α_r . From condition (2) we must have $N(\eta) = 1$ as required. $\square$

In our algorithm we will first divide out by the “trivial” shrinkage factor $f = \gcd(a_1, b_1, c_1, a_2, \dots, c_n)$. Then, using Proposition 9, we compute the nontrivial “ro-

tational" shrinkage, which will be of the form $N(\pi)^{-1}R_\pi$ for some $\pi \in S$ with odd norm. Writing $\pi = x + yi + zj + tk$, the rotation R_π has matrix

$$R_\pi = \frac{1}{N(\pi)} \begin{pmatrix} x^2 + y^2 - z^2 - t^2 & 2(yz - xt) & 2(xz + yt) \\ 2(xt + yz) & x^2 - y^2 + z^2 - t^2 & 2(zt - xy) \\ 2(yt - xz) & 2(xy + zt) & x^2 - y^2 - z^2 + t^2 \end{pmatrix}.$$

Hence the final transformation has the form $x \mapsto f^{-1}N(\pi)^{-1}R_\pi(x)$ for some $\pi \in S$ and $f \in \mathbf{Z}$.

The question of the running time of the algorithm will be considered below.

ALGORITHM. Shrink Polyhedra.

Input: $\{x_r = (a_r, b_r, c_r)\}$ * a set of n points which defines a polyhedron in $\mathbf{Z}^3$ *;

Output: $\{y_r = (a_r, b_r, c_r)\}$ * a set of n points which defines a similar polyhedron of minimum size in $\mathbf{Z}^3$ *;

- (1) BEGIN
- (2) FOR $r = 1$ TO n DO: $x_r \leftarrow (a_r, b_r, c_r) \leftarrow x_r - x_1$;
- (3) $f \leftarrow \gcd_{\mathbf{Z}}(a_2, b_2, c_2, a_3, \dots, c_n)$;
- (4) FOR $r = 1$ TO n DO: $x_r \leftarrow (a_r, b_r, c_r) \leftarrow x_r/f$;
- (5) $g \leftarrow \gcd_{\mathbf{Z}}(x_2 \cdot x_2, x_2 \cdot x_3, \dots, x_2 \cdot x_n, x_3 \cdot x_3, \dots, x_n \cdot x_n)$;
- (6) $d \leftarrow \text{squarepart}(g)$;
- (7) $\pi \leftarrow \gcd_{\mathbf{Z}}(d, a_1i + b_1j + c_1k, a_2i + b_2j + c_2k, \dots, a_ni + b_nj + c_nk)$;
- (8) FOR $r = 1$ TO n DO: $(a_r, b_r, c_r) \leftarrow R_\pi(x_r)/d$;
- (9) END.

PROPOSITION 10. *The algorithm Shrink Polyhedra takes as input a set of n points $x_1, x_2, \dots, x_n$ in $\mathbf{Z}^3$ which are the vertices of a polyhedron, and returns a set of n points in $\mathbf{Z}^3$ which are the vertices of a similar polyhedron of minimal size. It does so in time polynomial in the input size, plus the time it takes to compute the square part of $\gcd_{\mathbf{Z}}(\{x'_r \cdot x'_s \mid 1 \leq r \leq s \leq n\})$, where $x'_r = x_r - x_1$.*

Proof. Our work is essentially done. By Theorem 3, shrinking can be done if and only if (after translating so that one of the given points is at the origin) each $x_r \cdot x_s \in d^2\mathbf{Z}$ for some integer $d > 1$. After translating the origin to x_1 in line (2), we perform the "trivial" shrinking (which is possible if the coordinates of the points have a nontrivial common divisor) in lines (3)–(4). In lines (5)–(6) we compute the remaining shrinking factor d , and finally in lines (7)–(8) we carry out the nontrivial shrinking, using the result of Proposition 9. The matrix R_π in line (8) is the matrix of the rotation $\alpha \mapsto \pi\alpha\pi^{-1}$ given above.

The running time is dominated by the single computation of the square part of g , which occurs once in line (6). Unfortunately at present even the best probabilistic algorithms to do this are exponential, for they involve factoring g , and thus are of complexity $\exp(c\sqrt{\log g \log \log g})$.

The g.c.d. computation in line (3) takes $O(n \log a)$ steps where $a = \max_r(|a_r|, |b_r|, |c_r|)$. The g.c.d. computation in line (7) is a computation over the quaternions (see [3, Lemma 7.15] or [2, § 1, Lemmas 1 and 2] for a discussion and [5] for a precise time bound), and the time bound for this computation is similar. The time to factor g in line (6) is exponentially greater than these computations. $\square$

Finally we note (again) that we made no assumptions concerning convexity or holes, so the algorithm works for nonconvex polyhedra with holes as well.

5. The four-dimensional case. In this section we will prove Theorem 3 for $k = 4$, and give an algorithm which achieves maximum shrinking of a given configuration of points in $\mathbf{Z}^4$ in polynomial time. Again we will use quaternions.

Identifying $\mathbf{H}$ with $\mathbf{R}^4$, we first observe that for fixed $h \in \mathbf{H}$ the left and right multiplication maps $\lambda_h: x \mapsto hx$ and $\rho_h: x \mapsto xh$ from $\mathbf{R}^4$ to $\mathbf{R}^4$ are in fact similarities with scale factor $\sqrt{N(h)}$. This follows immediately from the equations

$$N(hx) = N(h)N(x) = N(x)N(h) = N(xh)$$

for $x \in \mathbf{H}$, since $N(x)$ is the square length of the vector x . In fact, every similarity of $\mathbf{R}^4$ has the form $x \mapsto h_1 x h_2$ for suitable nonzero quaternions h_1 and h_2 , with scale factor $\sqrt{N(h_1 h_2)}$, but we will not need to use this.

After shifting the origin and identifying the given points with elements of S (the ring of Hurwitz quaternions), the algorithm will consist of first dividing on the left by the greatest common left divisor (g.c.l.d.), then dividing on the right by the greatest common right divisor (g.c.r.d.) of the resulting points. This is guaranteed to give the maximum shrinkage, achieving the bound given in Theorem 3. Some care is needed to ensure that the transformed points do have integer coordinates, since S includes certain quaternions whose coordinates have a denominator of 2. We start with a lemma which essentially solves this problem. Let $S_1 = \{a + bi + cj + dk \mid a, b, c, d \in \mathbf{Z}\}$, which is a subgroup of index 2 in S .

LEMMA 11. *Let $\alpha \in S$. Then there exists a unit $\eta \in S$ such that $\eta\alpha \in S_1$ and $\alpha\eta \in S_1$.*

Proof. For the proof see [2, § 91, Lemma 7]. $\square$

Remarks. 1. It follows that every set of elements of S has both a g.c.r.d. and a g.c.l.d. in S_1 .

2. The unit group G of S has order 24 ([2, p. 153]) and the subgroup $H = G \cap S_1 = \{\pm 1, \pm i, \pm j, \pm k\}$ has order 8. Since H clearly preserves S_1 , in implementing Lemma 11 we need only try two possible values of η if α is not already in S_1 , namely representatives of the two other cosets of H in G . For these we may take η and $\bar{\eta}$, where $\eta = \frac{1}{2}(1 + i + j + k)$. In implementing the g.c.r.d. algorithm, we therefore assume that where possible the delivered value of $\pi = \text{gcd}(\alpha, \beta)$ will satisfy $\alpha\pi^{-1}, \beta\pi^{-1} \in S_1$. Similarly for the g.c.l.d. algorithm. Clearly this will not affect the order of the running time of these algorithms.

Now let $x_1, x_2, \dots, x_n$ be a set of points in $\mathbf{Z}^4$ with the property that $x_r \cdot x_s \in d\mathbf{Z}$ for all r, s . We must show that there exists a similarity R of $\mathbf{R}^4$ with scale factor $1/\sqrt{d}$ such that $R(x_r) \in \mathbf{Z}^4$ for all r . As usual it suffices to consider the case when $d = p$ is a prime. The following proposition now gives the desired result.

PROPOSITION 12. *Let $x_r = (a_r, b_r, c_r, d_r) \in \mathbf{Z}^4$ for $r = 1, 2, \dots, n$. Suppose that some prime p divides $x_r \cdot x_s$ for all r, s . Set $\alpha_r = a_r + ib_r + jc_r + kd_r \in S_1$. Then there exists $\pi \in S$ such that $N(\pi) = p$, and either $\pi^{-1}\alpha_r \in S_1$ for all r , or $\alpha_r\pi^{-1} \in S_1$ for all r . Hence either $\lambda_{\pi^{-1}}(x_r) \in \mathbf{Z}^4$ for all r or $\rho_{\pi^{-1}}(x_r) \in \mathbf{Z}^4$ for all r .*

Proof. Let X be the $n \times 4$ matrix with x_r in row r . We then have $XX^t \equiv 0 \pmod{p}$, from which it follows that the rank $\text{rk}_p(X)$ of X modulo p is at most 2. If $\text{rk}_p(X) = 0$ then each x_r is already in $p\mathbf{Z}^4$, so that shrinkage by a factor of p is trivially possible. Hence we may assume that $x_1 \notin p\mathbf{Z}^4$ and that for $r \geq 3$ we have $x_r \equiv \mu_r x_1 + \nu_r x_2 \pmod{p}$ for some integers μ_r and ν_r .

Now any common left or right divisor of p, α_1 , and α_2 will automatically divide α_r for $r \geq 3$. For simplicity write $\alpha = \alpha_1$ and $\beta = \alpha_2$. The given conditions now become

$$N(\alpha) \equiv N(\beta) \equiv \text{Tr}(\alpha\bar{\beta}) \equiv 0 \pmod{p},$$

and it suffices to show that α and β have a common factor π of norm p either on the

right or on the left, with the additional property that $\alpha\pi^{-1}, \beta\pi^{-1} \in S_1$, or $\pi^{-1}\alpha, \pi^{-1}\beta \in S_1$, respectively.

First consider the case where $p = 2$. The elements $1 + i$, $1 + j$, and $1 + k$ all have norm 2 and divide every element of S with even norm on both left and right. To ensure that $\pi^{-1}\alpha$ and $\pi^{-1}\beta$ are both in S_1 we must investigate further. Elementary calculation shows that (for $\alpha \in S_1$):

$$(1 + i)^{-1}\alpha \in S_1 \quad \text{if } \alpha \equiv 0, 1 + i + j + k, 1 + i, \text{ or } j + k \pmod{2S_1};$$

$$(1 + j)^{-1}\alpha \in S_1 \quad \text{if } \alpha \equiv 0, 1 + i + j + k, 1 + j, \text{ or } i + k \pmod{2S_1};$$

$$(1 + k)^{-1}\alpha \in S_1 \quad \text{if } \alpha \equiv 0, 1 + i + j + k, 1 + k, \text{ or } i + j \pmod{2S_1}.$$

Now the condition that $2 \mid \text{Tr}(\alpha\bar{\beta})$ gives us the further information that α and β both lie in one of the three sets $\{0, 1 + i + j + k, 1 + i, j + k\}$, $\{0, 1 + i + j + k, 1 + j, i + k\}$, and $\{0, 1 + i + j + k, 1 + k, i + j\}$ when reduced modulo 2. Hence at least one of $1 + i$, $1 + j$ or $1 + k$ satisfies the required condition on π .

Now let p be an odd prime. If p divides both α and β then the result is trivial, since we have $p = \pi\bar{\pi}$ for some π in S by [2, § 91, Lemma 7].

If p divides α but not β , then by Proposition 8(a), β has a right divisor of norm p , which clearly also divides α .

Now assume that neither α nor β is divisible by p . The quotient algebra $A = S/pS$ can be viewed as the quaternion algebra $(-1, -1/F)$ over the field $F = \mathbb{Z}/p\mathbb{Z}$. Such an algebra is necessarily split, that is, isomorphic to the 2×2 matrix algebra $M(2, F)$. (A is simple by [4, p. 14] and noncommutative; now $A \cong M(2, F)$ follows from Wedderburn's structure theorem for simple algebras, together with Wedderburn's theorem on the non-existence of finite noncommutative division algebras.) Under this isomorphism, the quaternion norm corresponds to the determinant, and the quaternion trace to the matrix trace. It then follows from elementary linear algebra (see Lemma 13 below) that either $\alpha\bar{\beta} \equiv 0 \pmod{p}$ or $\beta\alpha \equiv 0 \pmod{p}$.

Suppose that $\alpha\bar{\beta} \equiv 0 \pmod{p}$. We have $\alpha = \alpha_1\pi_1$ and $\beta = \beta_1\pi_2$ where $\pi_1 = \text{gcd}(p, \alpha)$ and $\pi_2 = \text{gcd}(p, \beta)$ both have norm p by Proposition 8(a). As in Proposition 8(c) we now deduce that p divides $\pi_2\bar{\pi}_1$, hence that $\pi_2\pi_1^{-1}$ is a unit, so that π_1 is a common right divisor of α and β .

Similarly if $\beta\alpha \equiv 0 \pmod{p}$ then $\text{gcd}(p, \alpha) = \text{gcd}(p, \beta) = \pi$ with $N(\pi) = p$, either by conjugation or by a "left-handed" version of Proposition 8. So in this case, π is a common left divisor of α and β .

To prove the last part, we may assume that $\pi \in S_1$ by Lemma 11. Then (assuming that π divides α on the left, say), $\bar{\pi}\alpha \in pS \cap S_1 = pS_1$, since p is odd, so that $\pi^{-1}\alpha \in S_1$; similarly for β . $\square$

LEMMA 13. Let F be any field, and let A and B be 2×2 matrices over F such that

$$\det(A) = \det(B) = \text{tr}(AB) = 0.$$

Then either $AB = 0$ or $BA = 0$.

Proof. If $A = 0$ or $B = 0$ then the result is trivial; otherwise we have $\text{rk}(A) = \text{rk}(B) = 1$. Moreover $\det(AB) = \text{tr}(AB) = 0$, so by the Cayley-Hamilton Theorem we have $(AB)^2 = 0$. Suppose that $AB \neq 0$; then $\text{rk}(AB) = 1$ also.

View each matrix as acting on the two-dimensional space F^2 of column vectors by left multiplication. Now $(AB)^2 = 0$ implies that $\text{im}(AB) \subseteq \ker(AB)$, but since $\dim \ker(AB) = \dim \text{im}(AB) = 1$, equality holds. Hence

$$\ker(B) \subseteq \ker(AB) = \text{im}(AB) \subseteq \text{im}(A);$$

but $\dim \ker(B) = \dim \text{im}(A) = 1$, so again equality holds. Hence $BA = 0$. $\square$

The algorithm to do the shrinking in four dimensions can now be presented.

ALGORITHM. Shrink Polytopes.

Input: $\{x_r = (a_r, b_r, c_r, d_r)\}$ * a set of n points which defines a polytope in $\mathbb{Z}^4$;

Output: $\{y_r = (a_r, b_r, c_r, d_r)\}$ * a set of n points which defines a similar polyhedron of minimum size in $\mathbb{Z}^4$;

- (1) BEGIN
- (2) FOR $r = 1$ TO n DO: $x_r \leftarrow (a_r, b_r, c_r, d_r) \leftarrow x_r - x_1$;
- (3) FOR $r = 1$ TO n DO: $\alpha_r \leftarrow a_r + ib_r + jc_r + kd_r$;
- (4) $\pi_1 \leftarrow \text{gcd}(\alpha_1, \dots, \alpha_n)$;
- (5) FOR $r = 1$ TO n DO: $\beta_r \leftarrow \pi_1^{-1} \alpha_r$;
- (6) $\pi_2 \leftarrow \text{gcd}(\beta_1, \dots, \beta_n)$;
- (7) FOR $r = 1$ TO n DO: $a_r + ib_r + jc_r + kd_r \leftarrow \gamma_r \leftarrow \beta_r \pi_2^{-1}$;
- (8) FOR $r = 1$ TO n DO: $y_r \leftarrow (a_r, b_r, c_r, d_r)$;
- (9) END.

PROPOSITION 14. *The algorithm Shrink Polytopes takes as input a set of n points $x_1, x_2, \dots, x_n$ in $\mathbb{Z}^4$ which are the vertices of a four-dimensional polytope, and returns a set of n points in $\mathbb{Z}^4$ which are the vertices of a similar polytope of minimal size. It does so in $O(n \log N)$ steps, where $N = \max \{|x_1|, |x_2|, \dots, |x_n|\}$.*

Proof. We first shift the origin in line (2), then convert the points x_r to elements $\alpha_r \in S_1$ in line (3). By Lemma 2 and Proposition 12, shrinking is possible if and only if the α_r have a nontrivial common divisor on the right or left. So if we divide out by their g.c.l.d. and then by the g.c.r.d. of the resulting points, we are guaranteed to have shrunk the given configuration to a minimal similar one.

In line (4) we compute $\pi_1 = \text{gcd}(\alpha_1, \alpha_2, \dots, \alpha_n)$, and then $\beta_r = \pi_1^{-1} \alpha_r \in S$ in line (5). By the second remark following Lemma 11 and Proposition 12, we may assume that each β_r is in fact in S_1 . Next in line (6) we compute $\pi_2 = \text{gcd}(\beta_1, \beta_2, \dots, \beta_n)$, and then $\gamma_r = \beta_r \pi_2^{-1} \in S$ in line (7). Again we may assume that each $\gamma_r \in S_1$. The map $\lambda(\pi_1^{-1})\rho(\pi_2^{-1})$ which maps $\alpha_r \mapsto \pi_1^{-1} \alpha_r \pi_2^{-1} = \gamma_r$ is then a similarity which shrinks the configuration by a factor of $(N(\pi_1)N(\pi_2))^{1/2}$. The resulting points γ_r have trivial g.c.r.d. and g.c.l.d.; it follows from Proposition 12 that the corresponding vectors $y_r \in \mathbb{Z}^4$ satisfy $\text{gcd}_{\mathbb{Z}}(\{y_r \cdot y_s\}) = 1$, so by Lemma 2 no further shrinkage is possible.

The running time of the algorithm is again dominated by the two g.c.d. computations on lines (4) and (6). This gives a running time of $O(n \log N)$ where N is the maximum of the $|x_r|$. $\square$

Acknowledgments. The authors thank Mike Harrison for bringing this problem to their attention, and the referee for the reference to Conway and Sloane's paper [1].

REFERENCES

- [1] J. H. CONWAY AND N. J. A. SLOANE, *Low-dimensional lattices V: Integral coordinates of integral lattices*, Proc. Roy. Soc. London, to appear.
- [2] L. E. DICKSON, *Algebras and Their Arithmetics*, Dover, New York, 1960.
- [3] I. N. HERSTEIN, *Topics in Algebra*, 2nd ed., Xerox College Publishing, Lexington, MA, 1982.
- [4] R. S. PIERCE, *Associative Algebras*, Springer-Verlag, Berlin, Heidelberg, New York, 1982.
- [5] M. RABIN AND J. SHALLIT, *Randomized algorithms in number theory*, Comm. Pure Appl. Math., 39 (1986), pp. 239–256.
- [6] E. G. REES, *Notes on Geometry*, Springer-Verlag, Berlin, Heidelberg, New York, 1983.
- [7] H. ROLLETSCHEK, *On the number of divisions of the Euclidean algorithm applied to Gaussian integers*, J. Symb. Comput., 2 (1986), pp. 261–291.
- [8] E. SNAPPER AND R. J. TROYER, *Metric Affine Geometry*, Academic Press, New York, 1971.

OPTIMAL ASSEMBLY OF AN s -STAGE k -OUT-OF- n SYSTEM*

D. Z. DU† AND F. K. HWANG‡

Abstract. Consider a system with n components and a reliability function $R(x_1, \dots, x_n)$ where x_j is the reliability of component j . Suppose that each component consists of the same set of parts, say h_i parts of type i for $i = 1, \dots, m$, which must all work for the component to work. The problem is to assign nh_i parts of type i , $i = 1, \dots, m$, with various part reliabilities to the n components to maximize the system reliability.

An assignment is called monotone if all the best parts go to one component, all the next best parts go to the second component, and so on. The optimality of monotone assignments for systems with symmetric and asymmetric reliability functions is studied. In particular, it is proved that monotone assignments are optimal for s -stage k -out-of- n systems.

Key words. k -out-of- n systems, reliability, majorization

AMS(MOS) subject classification. 26D20

1. Introduction. Consider a system with components $X_1, \dots, X_n$ and the reliability function $R(x_1, \dots, x_n)$ where x_j can be thought of as the probability that X_j works (we will refer to the system by $S(X_1, \dots, X_n)$). Suppose that each component consists of the same set of parts, say h_i parts of type i for $i = 1, \dots, m$, and that a component works if and only if all its parts work. The problem is to assign the $n \sum_{i=1}^m h_i$ parts, nh_i parts of type i , to the n components to maximize the system reliability. An assignment is called *monotone* if all the best parts go to one component, all the next best parts go to a second component, and so on.

$S(X_1, \dots, X_n)$ is called a k -out-of- n system if it fails if and only if at least k components fail. The series system and the parallel system are two special cases of the k -out-of- n system with $k = 1$ and $k = n$, respectively. A monotone assignment has been proved optimal when S is a k -out-of- n system. Derman, Lieberman, and Ross [1] first proved it for general k -out-of- n systems with $h_i = 1$ for all $i = 1, \dots, m$. Hwang, Sun, and Yao [6] proved it for parallel systems with $m = 1$. El-Newehi, Proschan, and Sethuraman [2] independently proved the same, and later [3], extended to general k -out-of- n systems and also with general m . Hwang [5] also independently proved it for general k -out-of- n systems and $m = 1$. It should be noted that even though the components do not have to have the same number of parts in the results reported in [2], [3], [5], and [6], the reliability functions are symmetric in their components.

In this paper we extend the study of monotone assignments from symmetric reliability functions to asymmetric ones. In particular, we prove the optimality of monotone assignment for the more general s -stage k -out-of- n systems. A 1-stage k -out-of- n system is just a k -out-of- n system. For $s \geq 2$ an s -stage k -out-of- n system is a k -out-of- n system whose i th component is an $(s - 1)$ -stage k_i -out-of- n_i system. We will refer to the components of the j -stage systems as the j -stage components. Such a system will be denoted by $S_s(X_1, \dots, X_N)$ and its reliability by $R_s(x_1, \dots, x_n)$, where X_i is the i th s -stage component and N is the total number of components in stage s . Our results pose no constraint on the values of m or h_i .

* Received by the editors March 20, 1989; accepted for publication (in revised form) August 24, 1989.

† Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

‡ AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

2. The methodology. A set $A = \{a_1 \geq a_2 \geq \cdots \geq a_n\}$ is said to *majorize* another set $B = \{b_1 \geq b_2 \geq \cdots \geq b_n\}$ if

$$\sum_{i=1}^k a_i \geq \sum_{i=1}^k b_i \quad \text{for } k = 1, \cdots, n-1,$$

and

$$\sum_{i=1}^n a_i = \sum_{i=1}^n b_i.$$

A symmetric function $f(x_1, \cdots, x_n)$ is called *Schur convex* if

$$f(a_1, \cdots, a_n) \geq f(b_1, \cdots, b_n)$$

for all A majorizing B . Theorem 1 is proved by Schur and Ostrowski (see [7]).

THEOREM 1. Let $I \subset \mathbb{R}$ be an open interval and let $f : I^n \rightarrow \mathbb{R}$ be differentiable. The function f is Schur convex if and only if

- (i) f is symmetric,
- (ii) for all $i \neq j$ and all $x \in I^n$,

$$(x_i - x_j) \left(\frac{\partial f}{\partial x_i} - \frac{\partial f}{\partial x_j} \right) \geq 0.$$

Since condition (ii) occurs frequently in the text, we will refer to it as *condition (*)*. We now give an equivalent form of Schur convexity.

THEOREM 2. The following two conditions are equivalent.

- (i) $f(a_1, \cdots, a_n) \geq f(b_1, \cdots, b_n)$ for all A majorizing B .
- (ii) $f(c_1 + u_1, \cdots, c_n + u_n) \geq f(c_1 + v_1, \cdots, c_n + v_n)$ for all $c_1 \geq c_2 \geq \cdots \geq c_n$, $u_1 \geq u_2 \geq \cdots \geq u_n$ and $\{u_1, \cdots, u_n\} = \{v_1, \cdots, v_n\}$ (i.e., $(u_1, \cdots, u_n)$ is a permutation of $(v_1, \cdots, v_n)$).

Proof. It is straightforward to verify that $\{c_1 + u_1, \cdots, c_n + u_n\}$ majorizes $\{c_1 + v_1, \cdots, c_n + v_n\}$. Hence (i) implies (ii). We now prove that (ii) implies (i).

It is well known [7] that A majorizing B , $A \neq B$, implies the existence of $A = A_0, A_1, \cdots, A_m = B$ ($m \geq 1$) such that A_i majorizes A_{i+1} and differs from A_{i+1} in exactly two elements. Therefore we may assume without loss of generality that in (i) A differs from B in only two elements with indices z and t . Suppose that $z < t$. Let $a_z - b_z (= b_t - a_t) = w$. Define

$$\begin{aligned} c_i &= a_i \quad \text{for } i \leq z, \\ &= a_z \quad \text{for } z < i < t, \\ &= a_z - w \quad \text{for } t \leq i, \end{aligned}$$

and

$$\begin{aligned} d_i &= 0 \quad \text{for } i \leq z, \\ &= a_i - a_z \quad \text{for } z < i < t, \\ &= a_i - a_z + w \quad \text{for } t \leq i. \end{aligned}$$

Note that

$$d_t = a_t - a_z + w = b_t - a_z \leq b_{t-1} - a_z = a_{t-1} - a_z = d_{t-1}.$$

Therefore, we have $c_1 \geq c_2 \geq \cdots \geq c_n$ and $d_1 \geq d_2 \geq \cdots \geq d_n$. Furthermore, $c_i + d_i = a_i$ for $i = 1, \cdots, n$ and

$$c_i + d_i = b_i \quad \text{for } i \neq z, \quad i \neq t,$$

$$c_z + d_t = a_z + a_t - a_z + w = b_t,$$

$$c_t + d_z = a_z - w = b_z.$$

Let σ be the permutation interchanging z and t and fixing other elements. Then $f(c_1 + d_1, \cdots, c_n + d_n) \geq f(c_1 + d_{\sigma(1)}, \cdots, c_n + d_{\sigma(n)})$ holds only if A majorizes B . The proof is complete.

Define condition (ii)' to be

(ii)' $f(u_1 + u_2, \cdots, u_{2n-1} + u_{2n}) \geq f(v_1 + v_2, \cdots, v_{2n-1} + v_{2n})$ for all $u_1 \geq u_2 \geq \cdots \geq u_{2n}$ and $\{u_1, \cdots, u_{2n}\} = \{v_1, \cdots, v_{2n}\}$.

Since $\{u_1 + u_2, \cdots, u_{2n-1} + u_{2n}\}$ majorizes $\{v_1 + v_2, \cdots, v_{2n-1} + v_{2n}\}$, we have the following corollary.

COROLLARY. (i) in Theorem 2 implies (ii)'.

Unfortunately, a proof of (ii)' implying (i) along the line of Theorem 2 cannot be obtained as illustrated by the following example. Let $A = \{3, 3, 0, 0\}$ and $B = \{3, 2, 1, 0\}$. Then A majorizes B . The only solution of $\{u_1 \geq u_2 \geq \cdots \geq u_8\}$ such that $u_{2i-1} + u_{2i} = a_i$ for $i = 1, 2, 3, 4$ is $u_1 = u_2 = u_3 = u_4 = 1.5$ and $u_5 = u_6 = u_7 = u_8 = 0$. But there is no solution for v_i such that $v_{2i-1} + v_{2i} = b_i$ for $i = 1, 2, 3, 4$. However, we still believe and conjecture that (i) and (ii)' are equivalent.

Define

$$L(y_1, \cdots, y_n) = R(e^{y_1}, \cdots, e^{y_n}).$$

THEOREM 3. If condition (*) holds for the symmetric function L , then the monotone assignment is optimal for the system $S(X_1, \cdots, X_n)$. Furthermore, if $h_i = 1$ for $i = 1, \cdots, m$, then the monotone assignment is always optimal (independent of the actual reliabilities of the parts) if and only if condition (*) holds for L .

Proof. We first show that Theorem 3 holds if and only if it holds when each component consists of two parts. The "only if" statement is obvious. To prove the "if" statement order the components according to the reliabilities where $X_{[i]}$ is more reliable than $X_{[j]}$ if $i < j$. Suppose that in an optimal assignment there exist $i < j$ such that a part in $X_{[i]}$ is inferior to another part of the same type in $X_{[j]}$. By combining all other parts in $X_{[i]}$ ($X_{[j]}$) into one part, then the assignment of the two parts of $X_{[i]}$ and $X_{[j]}$ is not monotone and can be improved, contradicting the assumption of an optimal assignment. We now prove Theorem 3 for two parts.

Case 1. The two parts are of different types.

Let $p_1 \geq p_2 \geq \cdots \geq p_n$ denote the working probabilities of the n parts of one type and $q_1 \geq q_2 \geq \cdots \geq q_n$ denote those of the other type. Then monotone assignment being optimal means that

$$R(p_1 q_1, \cdots, p_n q_n) \geq R(p_1 s_1, \cdots, p_n s_n),$$

where $\{q_1, \cdots, q_n\} = \{s_1, \cdots, s_n\}$, which is equivalent to

$$L(\ln p_1 + \ln q_1, \cdots, \ln p_n + \ln q_n) \geq L(\ln p_1 + \ln s_1, \cdots, \ln p_n + \ln s_n).$$

By Theorems 1 and 2 the above inequality holds if and only if condition (*) holds for L .

Case 2. The two parts are of the same type.

Let $p_1 \geq p_2 \geq \cdots \geq p_{2n}$ denote the working probabilities of the n parts. Then monotone assignment being optimal means that

$$R(p_1 p_2, \cdots, p_{2n-1} p_{2n}) \geq R(q_1 q_2, \cdots, q_{2n-1} q_{2n}),$$

where $\{p_1, \cdots, p_{2n}\} = \{q_1, \cdots, q_{2n}\}$, which is equivalent to

$$L(\ln p_1 + \ln p_2, \cdots, \ln p_{2n-1} + \ln p_{2n}) \geq L(\ln q_1 + \ln q_2, \cdots, \ln q_{2n-1} + \ln q_{2n}).$$

By Theorem 1 and the corollary of Theorem 2, the above inequality holds if condition (*) holds for L .

COROLLARY. L being Schur convex is necessary and sufficient for a monotone assignment to be always optimal.

An attempt to extend the Schur–Ostrowski theorem to the case that f is asymmetric was made by Hwang [4]. Let π be a partial order defined on a set $N = \{1, \cdots, n\}$ and let $i\pi j$ mean that i is larger than j under π . Let $A = \{a_1, \cdots, a_n\}$, $B = \{b_1, \cdots, b_n\}$ be two sets of weights. For $N' \subseteq N$, let $A(N')(B(N'))$ denote the sum of $a_i(b_i)$ over all i such that either $i \in N'$ or $i\pi j$ for some $j \in N'$. Then we say A majorizes B on π if $A(N) = B(N)$ and for every $N' \subset N$, $A(N') \geq B(N')$. Hwang proved the following theorem.

THEOREM 4. Let $f(x_1, \cdots, x_n)$ have continuous partial derivatives and be defined over the domain $D \subseteq I^n$. Then

$$f(a_1, \cdots, a_n) \geq f(b_1, \cdots, b_n)$$

for all A majorizing B on π if and only if f is such that for every i and j , $i\pi j$ implies $(\partial f / \partial x_i) \geq (\partial f / \partial x_j)$ for all $x \in D$.

Consider the function f in Theorem 4. Let $\sigma = (\sigma_1, \cdots, \sigma_n)$ denote any permutation of $\{1, \cdots, n\}$. For any $x \in D$, $x_1 \geq x_2 \geq \cdots \geq x_n$, we say $x \in D_\theta$ if

$$f(x_{\theta_1}, \cdots, x_{\theta_n}) = \max_{\sigma} \{f(x_{\sigma_1}, \cdots, x_{\sigma_n})\}.$$

Since f is continuous, D_θ is a region. For $x \in D_\theta$ define π_θ to be a linear order such that $i\pi_\theta j$ if and only if $\theta_i < \theta_j$ (implying $x_{\theta_i} \geq x_{\theta_j}$). We also define condition (θ^*) the same as condition (*) except changing the region from I^n to D_θ for all nonempty D_θ .

THEOREM 5. Suppose that for every σ such that D_σ is nonempty we have $f(a_1, \cdots, a_n) \geq f(b_1, \cdots, b_n)$ for all A majorizing B on π_σ . Then for any $c_1 \geq c_2 \geq \cdots \geq c_n$ and $u_1 \geq u_2 \geq \cdots \geq u_n$, $f(c_{\alpha_1} + u_{\beta_1}, \cdots, c_{\alpha_n} + u_{\beta_n})$ achieves its maximum at some $\alpha = \beta$ (α and β are two permutations).

Proof. Suppose to the contrary that the maximum is achieved at

$$f(c_{\alpha_1} + u_{\beta_1}, \cdots, c_{\alpha_n} + u_{\beta_n})$$

where $\alpha \neq \beta$. Let θ be such that $B = (c_{\alpha_1} + u_{\beta_1}, \cdots, c_{\alpha_n} + u_{\beta_n}) \in D_\theta$. Consider the vector A which has $c_i + u_i$ at the i th largest position of π_θ for $i = 1, \cdots, n$. Then A majorizes B on π_θ . By the assumption of Theorem 5, $f(A) \geq f(B)$, i.e., A , which satisfies $\alpha = \beta$, also achieves maximum.

THEOREM 6. If condition (θ^*) holds for the function L , then the monotone assignment is optimal for the system $S(X_1, \cdots, X_n)$.

Proof. Consider $x \in D_\theta$. By our definition of π_θ , $i\pi_\theta j$ implies $\theta_i < \theta_j$, hence $x_i \geq x_j$. Therefore condition (θ^*) on a function f is equivalent to the condition “ $i\pi_\theta j$ implies $\partial f / \partial x_i \geq \partial f / \partial x_j$ for all $x \in D_\theta$.” By Theorem 4, $f(a_1, \cdots, a_n) \geq f(b_1, \cdots, b_n)$ for all A majorizing B on π_θ . By Theorem 5, $f(c_{\alpha_1} + u_{\beta_1}, \cdots, c_{\alpha_n} + u_{\beta_n})$ achieves its maximum at some $\alpha = \beta$. The rest of the proof is analogous to the proof of Theorem 3.

3. Optimal assignments for s -stage k -out-of- n systems. Pledger and Proschan [8] proved that $L_1(\ln x_1, \dots, \ln x_N)$ is Schur convex. Note that for $s \geq 2$, $R_s(x_1, \dots, x_N)$ is not symmetric in general since X_i and X_j can belong to different components of stage $s - 1$ which have different parameters (k, n) as k -out-of- n systems. Therefore, the Schur convex idea cannot be extended to the multistage case. However, we now have the machinery built in § 2 to deal with the asymmetric case. We prove the following Theorem.

THEOREM 7. $L_s(x_1, \dots, x_N)$ satisfies condition (θ^*) .

Proof. Theorem 7 is true for $s = 1$ since by Theorem 1 a Schur convex function always satisfies condition $(*)$. We prove the general $s \geq 2$ case by induction. Consider $x \in D_\theta$. To prove that condition (θ^*) holds for $L(x_1, \dots, x_n)$, it is equivalent to prove

$$(x_i - x_j) \left(x_i \frac{\partial R}{\partial x_i} - x_j \frac{\partial R}{\partial x_j} \right) \geq 0 \quad \text{for all } x \in D_\theta.$$

Let Y_i denote the i th component of stage 1. Suppose that X_i is in Y_z and X_j in Y_t . If $z = t$, then since Y_t is an $(s - 1)$ -stage k_t -out-of- n_t system, condition $(*)$ holds by induction. Therefore we assume that $z \neq t$. Denote by $w_z(w_t)$ the working probability of component $z(t)$. Then

$$\begin{aligned} R(x_1, \dots, x_n) &= w_z w_t Q'' + \{w_z(1 - w_t) + w_t(1 - w_z) + w_z w_t\} Q' + Q \\ &= w_z w_t (Q'' - Q') + (w_z + w_t) Q' + Q, \end{aligned}$$

where $Q'(Q'')$ is the probability of exactly $k - 1$ ($k - 2$) components, other than Y_z and Y_t , working and Q is the probability of at least k components, other than Y_z and Y_t , working. Write

$$w_z = ax_i + b,$$

$$w_t = cx_j + d.$$

For $s = 2$, $a(c)$ is the probability of exactly $k_z - 1$ ($k_t - 1$) components of stage s , other than $X_i(X_j)$, in $Y_z(Y_t)$ working; and $b(d)$ is the probability of at least $k_z(k_t)$ components of stage s , other than $X_i(X_j)$, in $Y_z(Y_t)$ working. For $s > 2$, w_z and w_t are still linear functions of x_i and x_j since they are compositions of such linear functions. Now

$$\begin{aligned} x_i \frac{\partial R}{\partial x_i} - x_j \frac{\partial R}{\partial x_j} &= \{x_i a(cx_j + d) - x_j c(ax_i + b)\} (Q'' - Q') + (x_i a - x_j c) Q' \\ &= (x_i ad - x_j bc) (Q'' - Q') + (x_i a - x_j c) Q' \\ &= (x_i - x_j) \{bc(Q'' - Q') + cQ'\} + x_i [\{ad(Q'' - Q') + aQ'\} - \{bc(Q'' - Q') + cQ'\}]. \end{aligned}$$

Consider the system S' obtained from S by interchanging X_i and X_j . Since θ is an optimal permutation for $x \in D_\theta$, the reliability does not increase through this interchange. Hence

$$\begin{aligned} (x_i a + b)(x_j c + d)(Q'' - Q') + (x_i a + b + x_j c + d)Q' + Q \\ \geq (x_j a + b)(x_i c + d)(Q'' - Q') + (x_j a + b + x_i c + d)Q' + Q, \end{aligned}$$

or

$$(x_i - x_j) [\{ad(Q'' - Q') + aQ'\} - \{bc(Q'' - Q') + cQ'\}] \geq 0.$$

By noting that

$$bc(Q'' - Q') + cQ' = bcQ'' + cQ'(1 - b) \geq 0,$$

we conclude

$$(x_i - x_j) \left(x_i \frac{\partial R}{\partial x_i} - x_j \frac{\partial R}{\partial x_j} \right) \geq 0.$$

COROLLARY. $L_s(x_1, \dots, x_N)$ is Schur convex if the s -stage k -out-of- n system is symmetrical in every stage.

THEOREM 8. *The monotone assignment is optimal for any s -stage k -out-of- n system.*

Proof. The proof follows immediately from Theorems 6 and 7.

4. Conclusion. Previous results have shown that if the system reliability is Schur convex, then a monotone assignment is always optimal. By deriving an equivalent form of Schur convexity, we proved that the condition is also necessary if each type has only one part, i.e., $h_i = 1$. We conjectured that the same holds for general h_i . We then dealt with the difficult case when the system reliability function is asymmetric. We used the framework laid down by Hwang in extending Schur convexity to asymmetric functions. We obtain conditions under which a monotone assignment is optimal for the asymmetric case. In particular, we showed that the s -stage k -out-of- n systems satisfy these conditions by a novel argument.

REFERENCES

- [1] C. Derman, G. J. Lieberman, and S. M. Ross, *On optimal assembly of systems*, Naval Res. Logist. Quart., 19 (1972), pp. 569–574.
- [2] E. El-Newehi, F. Proschan, and J. Sethuraman, *Optimal allocation of components in parallel-series and series-parallel systems*, J. Appl. Probab., 23 (1986), pp. 770–777.
- [3] ———, *Optimal assembly of systems using Schur functions and majorization*, Naval Res. Logist., 34 (1987), pp. 705–712.
- [4] F. K. Hwang, *Majorization on a partially ordered set*, Proc. Amer. Math. Soc., 76 (1979), pp. 199–203.
- [5] ———, *Optimal assignment of components to a two-stage k -out-of- n system*, Math. Oper. Res., 14 (1989), pp. 376–382.
- [6] F. K. Hwang, J. Sun, and E. Y. Yao, *Optimal set partitioning*, SIAM J. Algebraic Discrete Methods, 6 (1985), pp. 163–170.
- [7] A. W. Marshall and I. Olkin, *Inequalities via majorization—an introduction*, in General Inequalities 3, E. E. Beckenbach and W. Walter, eds., Springer-Verlag, Boston, 1983, pp. 165–187.
- [8] G. Pledger and F. Proschan, *Comparisons of order statistics and of spacings from heterogeneous distributions*, in Optimization Methods in Statistics, J. S. Rustagi, ed., Academic Press, New York, 1971, pp. 89–113.

CANONICAL ANTICHAINS ON THE CIRCLE AND APPLICATIONS*

P. FRANKL†

Abstract. A family $\mathcal{F}$ of subsets of the n -set $[n]$ is a k -antichain if it contains no $F_0, \dots, F_k$ satisfying $F_0 \subset \dots \subset F_k$. Extending theorems of Sperner, Erdős, Bollobás, and Purdy the maximum size of k -antichains is determined under the additional assumptions that (i) $\mathcal{F}$ is self-complementary ($F \in \mathcal{F}$ implies $([n] - F) \in \mathcal{F}$) (ii) complement-free ($F \in \mathcal{F}$ implies $([n] - F) \notin \mathcal{F}$). These results are corollaries of more general results on convex hulls of f -vectors. An easy proof of a theorem of Alon and a two-family version of it are also provided.

Key words. antichain, shadow, cyclic permutation

AMS(MOS) subject classification. 05A05

1. Introduction. Let $[n]$ be the n -element set $\{1, 2, \dots, n\}$. For a cyclic permutation $\pi = (x_1, \dots, x_n)$ of $\{1, 2, \dots, n\}$ let us define the arc $A(x_i, x_j)$ by

$$A(x_i, x_j) = \begin{cases} \{x_i, x_{i+1}, \dots, x_j\} & \text{if } 1 \leq i \leq j \leq n \\ \{x_j, x_{j+1}, \dots, x_n, x_1, \dots, x_i\} & \text{if } 1 \leq j < i \leq n. \end{cases}$$

Set $\mathcal{A}(\pi) = \{A(x_i, x_j) : 1 \leq i, j \leq n\} \cup \{\emptyset\}$, the set of arcs. Note that $|\mathcal{A}(\pi)| = n(n-1) + 2$, i.e., we do not distinguish between $A(x_1, x_n), A(x_2, x_1), \dots, A(x_n, x_{n-1})$ which all denote $[n]$.

The interest in $\mathcal{A}(\pi)$ goes back to Katona ([K1], [K2]) who used these families to give a short proof of the Erdős–Ko–Rado Theorem [EKR].

Given a family $\mathcal{F} \subset 2^{[n]}$ and a cyclic permutation π , define $\mathcal{F}(\pi) = \mathcal{F} \cap \mathcal{A}(\pi)$.

The f -vector $f(\mathcal{F})$ of a family is $(f_0, \dots, f_n) \in R^{n+1}$ where f_i is the number of sets of size i in $\mathcal{F}$.

The connection between $\mathcal{F}$ and $\mathcal{F}(\pi)$ is given by Katona's identity—the summation is over all $(n-1)!$ cyclic permutations

$$(1) \quad \frac{1}{(n-1)!} \sum_{\pi} f(\mathcal{F}(\pi)) = \left(f_0, nf_1 / \binom{n}{1}, nf_2 / \binom{n}{2}, \dots, nf_{n-1} / \binom{n}{n-1}, f_n \right).$$

This identity can be proved by double counting or by observing that both sides express the expected number of i -element sets being arcs with respect to a random cyclic permutation.

For a real vector $y = (y_0, \dots, y_n)$ let $z(y) = (z_0, \dots, z_u)$ be defined by

$$z_i = \begin{cases} y_i & \text{for } i = 0 \text{ or } n \\ y_i \binom{n}{i} / n & \text{for } 1 \leq i < n. \end{cases}$$

In many extremal problems we are looking for $M = \max(f(\mathcal{F}), y)$ where the maximum is over all families satisfying a given condition (where (x, y) denotes the usual scalar product).

* Received by the editors September 6, 1988; accepted for publication August 15, 1989.

† Centre National de la Recherche Scientifique, Paris, France and AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

Then Katona's identity implies the (often best possible) upper bound

$$(2) \quad M \leq \max_{\pi, \mathcal{F}} (f(\mathcal{F}(\pi)), z(y)).$$

Various forms of (2) have been successfully applied by many authors.

Let us now recall two of the most basic results of extremal set theory: Sperner's theorem and Erdős's generalization thereof. A sequence of distinct sets $F_0, F_1, \dots, F_k$ is called a *chain of length k* if $F_0 \subset F_1 \subset \dots \subset F_k$ holds. A family $\mathcal{F} \subset 2^{[n]}$ is called a *k -antichain* if it contains no chain of length k . For $k = 1$ we speak simply of antichains. For $0 \leq r_1 < \dots < r_k \leq n$ the family

$$(3) \quad \binom{X}{r_1} \cup \dots \cup \binom{X}{r_k} \text{ is a } k\text{-antichain of size } \binom{n}{r_1} + \dots + \binom{n}{r_k}.$$

This number is maximal if

$$\{r_1, \dots, r_k\} = \left\{ \left\lfloor \frac{n-k+1}{2} \right\rfloor, \dots, \left\lfloor \frac{n+k-1}{2} \right\rfloor \right\} \quad \text{or} \quad \left\{ \left\lceil \frac{n-k+1}{2} \right\rceil, \dots, \left\lceil \frac{n+k-1}{2} \right\rceil \right\}$$

(for $n+k$ odd the two sets coincide).

SPERNER THEOREM [S]. *If $\mathcal{F} \subset 2^{[n]}$ is an antichain, then $|\mathcal{F}| \leq \binom{n}{\lfloor n/2 \rfloor}$, with equality holding if and only if $\mathcal{F} = \binom{X}{\lfloor n/2 \rfloor}$ or $\mathcal{F} = \binom{X}{\lceil n/2 \rceil}$.*

ERDŐS THEOREM [E]. *If $\mathcal{F} \subset 2^{[n]}$ is a k -antichain then $|\mathcal{F}|$ is maximal if and only if $\mathcal{F}$ is of the form (3) with*

$$\{r_1, \dots, r_k\} = \left\{ \left\lfloor \frac{n-k+1}{2} \right\rfloor, \dots, \left\lfloor \frac{n+k-1}{2} \right\rfloor \right\} \quad \text{or} \quad \left\{ \left\lceil \frac{n-k+1}{2} \right\rceil, \dots, \left\lceil \frac{n+k-1}{2} \right\rceil \right\}.$$

The Kruskal-Katona Theorem ([Kr], [K3]; for a short proof see [F]) makes it possible to determine the set of possible f -vectors of antichains (cf. [C], [DGH]).

However, for k -antichains, $k \geq 2$, no such characterization is known. The main result of the present paper is such a characterization for k -antichains $\mathcal{F} \subset \mathcal{A}(\pi)$.

Since the particular choice of π is irrelevant, we set $\mathcal{A} = \mathcal{A}(\pi)$ and assume that π is the identity. For a vector $b = (b_0, b_1, \dots, b_n) \in \mathbb{R}^n$ define

$$\delta(b) = \min \{i : b_i > 0\},$$

$$\Delta(b) = \max \{i : b_i > 0\},$$

$$|b| = \sum_i b_i.$$

Set also $\mathbb{N}^{n+1} = \{(b_0, \dots, b_n) : b_i \text{ is nonnegative integer, } 0 \leq i \leq n\}$.

THEOREM 1.1. *A vector $f = (f_0, \dots, f_n) \in \mathbb{N}^{n+1}$ is the f -vector of some antichain $\mathcal{F} \subset \mathcal{A}$ if and only if one of (i)–(iv) holds.*

$$(i) \quad f = (0, \dots, 0);$$

$$(ii) \quad f = (1, 0, \dots, 0);$$

$$(iii) \quad f = (0, \dots, 0, 1);$$

$$(iv) \quad 0 < \delta(f) \leq \Delta(f) < n \text{ and } \delta(f) + |f| - \Delta(f) \leq n.$$

For vectors $b = (b_0, \dots, b_n), h = (h_0, \dots, h_n) \in \mathbb{N}^{n+1}$ we define a partial order $>$ by setting $b > h$ if and only if $\Delta(b) \leq \delta(h)$ holds.

THEOREM 1.2. *A vector $f = (f_0, \dots, f_n) \in \mathbb{N}^{n+1}$ is the f -vector of some k -antichain $\mathcal{F} \subset \mathcal{A}$ if and only if (i) and (ii) hold.*

$$(i) \quad f_0 \leq 1, f_n \leq 1, f_i \leq n \text{ for } 0 < i < n;$$

(ii) *there exist vectors $h_1 < h_2 < \dots < h_k$, each of them the f -vector of some antichain in $\mathcal{A}$, such that $f = h_1 + \dots + h_k$ holds.*

2. The proof of Theorem 1.1. First we prove the sufficiency of (i)–(iv). This is trivial for (i)–(iii). Let us consider (iv).

Define the *canonical family* $\mathcal{A}(f) = \{F_1, \dots, F_{|f|}\}$ explicitly by

$$F_i = A(i, i+j-1)$$

where j is determined by

$$f_1 + \dots + f_{j-1} < i \leq f_1 + \dots + f_j$$

and $i+j$ is computed modulo n . Now $f(\mathcal{F}) = (f_0, \dots, f_n)$ is clear from the construction.

Observe that $|F_1| \leq \dots \leq |F_{|f|}|$ holds. Also, $A(i', i' + j' - 1) \subset A(i, i + j - 1)$ for $1 \leq i' < i \leq n$ and $j' \leq j$ is equivalent to $i + j \geq i' + j' + n$. This inequality would imply in our case $|f| + \Delta(f) \geq 1 + \delta(f) + n$, a contradiction.

Now we prove the necessity. It is trivial for $|f| \leq 1$ and also for $\delta(f) = \Delta(f)$. Using this as the base case, we apply induction on $\Delta(f) - \delta(f)$.

Let $G_1, \dots, G_s$ be all the $\delta(f)$ -element sets in the antichain $\mathcal{F} \subset \mathcal{A}$ with $f(\mathcal{F}) = (f_0, \dots, f_n)$.

Since $\Delta(f) - \delta(f) \geq 1$, $s < n$ follows. Thus we may suppose that $A(n, \delta(f) - 1) \notin \mathcal{F}$, $A(1, \delta(f)) \in \mathcal{F}$.

Set $l = \delta(f)$ and let $i_1, i_2, \dots, i_{f_l}$ be the starting points of the arcs of length l in $\mathcal{F}$. Define $\mathcal{F}'$ by replacing these arcs in $\mathcal{F}$ by those arcs of length $l + 1$ which start at $i_1, i_2, \dots, i_{f_l}$ and n . Then $\mathcal{F}'$ is an antichain with $\delta(\mathcal{F}') = \delta(\mathcal{F}) + 1$, $\Delta(\mathcal{F}') = \Delta(\mathcal{F})$. Thus by induction we have

$$n \geq |\mathcal{F}'| + \Delta(\mathcal{F}') - \delta(\mathcal{F}') = |\mathcal{F}| + \Delta(\mathcal{F}) - \delta(\mathcal{F}). \quad \square$$

3. The proof of Theorem 1.2. Let $\mathcal{F} \subset \mathcal{A}$ be a k -antichain. It is easy to find a partition $\mathcal{F} = \mathcal{G}_1 \cup \dots \cup \mathcal{G}_k$ such that $\mathcal{G}_i$ is an antichain for $i = 1, \dots, k$. For example, use induction on k , the case $k = 1$ being trivial. For $k \geq 2$ define

$$\mathcal{G}_1 = \{G \in \mathcal{F} : \exists F \in \mathcal{F}, F \neq G, F \subset G\}.$$

Then $\mathcal{G}_1 \subset \mathcal{F}$ is an antichain and $\mathcal{F} - \mathcal{G}_1$ is a $(k - 1)$ -antichain, which can be partitioned into $k - 1$ antichains by induction.

The principal difficulty comes from the fact that $f(\mathcal{G}_1) > \dots > f(\mathcal{G}_k)$ does not necessarily hold.

In many cases, such a partition does not exist at all; for example, for

$$\mathcal{F} = \{\{1\}, \{1, 2\}, \{2, 3, 4\}, \{2, 3, 4, 5\}\}.$$

Fortunately, we are only dealing with f -vectors and not with the actual k -antichains. The proof of the theorem is based on the following lemma.

LEMMA 3.1. *Let $\mathcal{G}, \mathcal{H} \subset \mathcal{A}$ be antichains. Then there exist antichains $\mathcal{G}', \mathcal{H}' \subset \mathcal{A}$ with*

$$f(\mathcal{G}) + f(\mathcal{H}) = f(\mathcal{G}') + f(\mathcal{H}') \text{ and } f(\mathcal{G}') > f(\mathcal{H}').$$

Proof. We may assume that $f(\mathcal{G}) > f(\mathcal{H})$ does not hold and consequently, $b_0 = b_n = 0$ (e.g., $b_0 = 1$ implies $\mathcal{H} = \{\emptyset\}$ and thus $f(\mathcal{G}) > f(\mathcal{H})$).

Define $g', h' \in \mathbb{N}^{n+1}$ by $g' + h' = g + h$, $g' > h'$, $\mathcal{A}(g')$ is an antichain and $|g'|$ is maximal with respect to these properties (the canonical family $\mathcal{A}(g')$ was defined in § 2).

The maximality of g' means that

$$|g'| + 1 + \delta(h') - \delta(g') \geq n + 1.$$

If $\mathcal{A}(h')$ is an antichain then we are done.

Suppose that $\mathcal{A}(h')$ is not an antichain, that is,

$$|h'| + \Delta(h') - \delta(h') \geq n + 1.$$

Adding these two inequalities gives

$$(3.1) \quad |g| + |h| + \Delta(h') - \delta(g') \geq 2n + 1.$$

Using Theorem 1.1 and the fact that $\mathcal{G}$ and $\mathcal{H}$ are antichains gives

$$(3.2) \quad |g| + |h| + \Delta(h) + \Delta(g) - \delta(h) - \delta(g) \leq 2n.$$

Note that $\Delta(h') = \max \{ \Delta(h), \Delta(g) \}$ and $\delta(g') = \min \{ \delta(h), \delta(g) \}$ hold. Consequently, subtracting (3.2) from (3.1) gives

$$\max \{ \delta(h), \delta(g) \} - \min \{ \Delta(h), \Delta(g) \} \geq 1$$

which implies that either $\delta(h) > \Delta(g)$ or $\delta(g) > \Delta(h)$, that is either $h > g$ or $g > h$ holds. $\square$

Now the proof of Theorem 1.2 is easy. For a k -antichain $\mathcal{F} \subset \mathcal{A}$ let $\mathcal{G}_1, \dots, \mathcal{G}_k \subset \mathcal{A}$ be antichains satisfying $f(\mathcal{G}_1) + \dots + f(\mathcal{G}_k) = f(\mathcal{F})$ and such that the number of relations $f(\mathcal{G}_i) > f(\mathcal{G}_j)$, $1 \leq i < j \leq k$ is maximized.

We claim that $f(\mathcal{G}_i) > f(\mathcal{G}_j)$ holds for all $1 \leq i < j \leq k$. If not, then choose (i, j) , violating this with i as small as possible and for given i , let j be as large as possible.

By Lemma 3.1 we can replace $\mathcal{G}_i, \mathcal{G}_j$ by antichains $\mathcal{H}_i, \mathcal{H}_j \subset \mathcal{A}$ with $f(\mathcal{H}_i) > f(\mathcal{H}_j)$.

To get a contradiction we show that all the relations $f(\mathcal{G}_{i'}) > f(\mathcal{G}_{j'})$, $1 \leq i' < j' \leq k$ are maintained. This is trivial for $\{i, j\} \cap \{i', j'\} = \emptyset$. For $i = i'$ it follows from $\Delta(\mathcal{H}_i) \leq \Delta(\mathcal{G}_i)$, for $j = j'$ from $\delta(\mathcal{H}_j) \geq \delta(\mathcal{G}_j)$.

If $i = j'$ then the minimal choice of i implies $f(\mathcal{G}_i) > f(\mathcal{G}_j)$ and, consequently, $f(\mathcal{G}_{i'}) > f(\mathcal{H}_i)$. Finally, if $i' = j$ then the maximal choice of j implies $f(\mathcal{G}_i) > f(\mathcal{G}_{j'})$ which leads to $f(\mathcal{H}_j) > f(\mathcal{G}_{j'})$.

Thus we proved $f(\mathcal{G}_1) > \dots > f(\mathcal{G}_k)$; to end the proof of the necessity set $h_i = f(\mathcal{G}_i)$.

To prove sufficiency form the canonical family $\mathcal{A}(f)$. We claim that it is a k -antichain. Indeed, if

$$\mathcal{A}(f) = \{F_1, \dots, F_{|f|}\} \quad \text{with} \quad |F_1| \leq \dots \leq |F_{|f|}|$$

then define

$$\mathcal{G}_i = \{F_j : |h_1| + \dots + |h_{i-1}| < j \leq |h_1| + \dots + |h_i|\}.$$

Note that $\mathcal{A}(f) = \mathcal{G}_1 \cup \dots \cup \mathcal{G}_k$ and $\mathcal{G}_i$ is isomorphic to $\mathcal{A}(h_i)$; consequently, it is an antichain. $\square$

Note that we have actually proved the theorem in the following form.

COROLLARY 3.2. *A vector $f = (f_0, \dots, f_n) \in \mathbb{N}^{n+1}$ is the f -vector of some k -antichain $\mathcal{F} \subset \mathcal{A}$ if and only if the canonical family $\mathcal{A}(f)$ is a k -antichain.*

4. Convex hulls on the circle. For a class $\mathcal{B}$ of families $\mathcal{F} \subset 2^{[n]}$ we can look at the convex hull $C(\mathcal{B})$ of all the points $f(\mathcal{F}) \in \mathbb{R}^{n+1}$, $\mathcal{F} \in \mathcal{B}$.

A set $D = D(\mathcal{B}) \subset C(\mathcal{B})$ is called the *dominating set* for $\mathcal{B}$ if for every vector $f \in C(\mathcal{B})$ there exist vectors $f_1, \dots, f_s \in D$ and nonnegative real numbers $\alpha_1, \dots, \alpha_s$ summing to 1 with $f \leq \sum_{0 \leq i \leq s} \alpha_i f_i$ holding coordinatewise.

It is not difficult to see (cf. [EFK1]) that the set of vertices of $C(\mathcal{B})$ can be obtained from D by changing arbitrary sets of coordinates of vectors of D to zero (therefore $|C(\mathcal{B})| \leq 2^n |D|$ holds).

A k -antichain, $\mathcal{F}$ is called self-complementary (complement-free) if $F \in \mathcal{F}$ implies $F^c = ([n] - F) \in \mathcal{F}$ ($F^c \notin \mathcal{F}$), respectively.

The dominating set for k -antichains $\mathcal{F} \subset \mathcal{A}$ was determined in [EFK2].

THEOREM 4.1. *The dominating set for k -antichains $\mathcal{F} \subset \mathcal{A}$ consists of all vectors $(f_0, \dots, f_n)$ with exactly k nonzero coordinates, where $f_i \neq 0$ implies $f_i = 1$ for $i = 0, n$ and $f_i = n$ for $1 \leq i \leq n-1$.*

We shall prove the following results.

THEOREM 4.2. *The dominating set for self-complementary k -antichains $\mathcal{F} \subset \mathcal{A}$ consists of the following vectors $f = (f_0, \dots, f_n)$ satisfying $f_i = f_{n-i}$ for all i :*

(i) *f has exactly k -nonzero coordinates where $f_i \neq 0$ implies $f_i = 1$ or n accordingly as $i = 0, n$ or $1 \leq i < n$ hold;*

(ii) *there exists $1 \leq i < n/2$ with $f_i = i = f_{n-i}$ and $(f_0, \dots, f_{i-1}, 0, f_{i+1}, \dots, f_{n-i-1}, 0, f_{n-i+1}, \dots, f_n)$ is of the form (i) with k replaced by $k-1$.*

THEOREM 4.3. *The dominating set for complement-free antichains $\mathcal{F} \subset \mathcal{A}$ consists of the following vectors $F = (f_0, \dots, f_n)$ satisfying $f_i f_{n-i} = 0$ for $i < n/2$:*

(i) *f has exactly k -nonzero coordinates where $f_i = 1$ or n accordingly as $i = 0, n$ or $0 < i < n$ hold. Moreover $f_{n/2} = 0$ if n is even.*

(ii) *n is even and there exists some $0 < i < n, i \neq n/2$ such that $f_i = \min\{i, n-i\}, f_{n/2} = n/2$ and replacing these two coordinates by zero in f results in a vector coming from (i) with k replaced by $k-1$.*

Note that in both Theorems 4.2 and 4.3 each vector in (ii) has $k+1$ nonzero coordinates.

Let us mention that Theorems 4.2 and 4.3 were proved by Engel and Erdős [EE] for the special case $k = 1$. Their proof does not seem to generalize for $k \geq 2$.

We prove both theorems together, in passing giving a new (shorter) proof of Theorem 4.1 as well.

The proof, of course, is based on Theorem 1.2. Actually, we use it in the form of Corollary 3.2.

Let $f = (f_0, \dots, f_n)$ be the f -vector of some k -antichain in $\mathcal{A}$. We prove the statement by induction on the number of nonzero coordinates. If this number is 1, then the statements are obvious. If $f_0 = 1$ or $f_n = 1$ or $f_i = n$ for some $1 \leq i < n$, then replace it by zero (also f_{n-i} in the self-complementary case). What remains is the f -vector of a $(k-1)$ -antichain ($k-2$ in the self-complementary case) by Corollary 3.2, with fewer nonzero coordinates, and the result follows by induction.

Let now f_i and f_j be the first two nonzero coordinates. We may assume that $1 \leq i < j < n, f_i < n, f_j < n$ hold.

Let $f'(f'')$ be the vector obtained from f by replacing (f_i, f_j) by $(f_i + f_j, 0)$ and $(0, f_i + f_j)$, respectively. Since these vectors have fewer nonzeros and $f = f'f_i/(f_i + f_j) + f''f_j/(f_i + f_j)$, it would be sufficient to show that f' and f'' are the f -vectors of some k -antichains in $\mathcal{A}$. (In the self-complementary case, if $j < n/2$, then we replace (f_{n-j}, f_{n-i}) by $(0, f_i + f_j)$ or $(f_i + f_j, 0)$, as well.) From Corollary 3.2 it is clear that this is the case unless $f_i + f_j > n$ holds.

In this case we replace (f_i, f_j) by $(n, f_i + f_j - n)$ and $(f_i + f_j - n, n)$, respectively, to conclude the proof.

There are only two cases which remain. The first is when f is the f -vector of a self-complementary k -antichain and $j > n/2$ holds. Now $f_j = f_{n-j}$ implies that $i = n - j$ and there are no other nonzero coordinates. If $k \geq 2$, then the statement is obvious. If $k = 1$, then Theorem 1.1 implies $f_i + f_{n-i} + n - 2i \leq n$, that is, $f_i = f_{n-i} \leq i$ in our case, as desired.

The other case is when $j = n/2$. In the self-complementary subcase it follows that $f_l = 0$ except for $l = i, n/2, n - i$. For $k \geq 3$ there is nothing to prove. If $k = 1$ then

Theorem 1.1 implies $|f| \leq 2i$. Now

$$f \leq \frac{f_i}{i}(0, \dots, \overset{i}{i}, \dots, \overset{n-i}{i}, \dots, 0) + \frac{i-f_i}{i}(0, \dots, \overset{n/2}{n}, \dots, 0)$$

follows from $2i \leq n$.

Finally, if $k = 2$, then self-complementarity and Theorem 1.2 imply that $(0, \dots, f_i, \dots, \frac{1}{2}f_{n/2}, \dots, 0, \dots, 0)$ and $(0, \dots, 0, \dots, \frac{1}{2}f_{n/2}, \dots, f_i, \dots, 0)$ are f -vectors of antichains. Thus, by Theorem 1.1, $|f| \leq n + 2i$ follows. Consequently, f is dominated by a convex linear combination of

$$(0, \dots, \overset{i}{i}, \dots, \overset{n-i}{n}, \dots, 0) \quad \text{and} \quad (0, \dots, \overset{i}{i}, \dots, \overset{n/2}{n}, \dots, \overset{n-i}{i}, \dots, 0),$$

as desired.

The very last case is when $j = n/2$ and the k -antichain is supposed to be complement-free (the case $i = n/2$ is handled by symmetry). Arguing, if necessary for the f -vector $(f_0, \dots, f_n)$, we may assume that there are at most three positive coordinates, say, $f_i, f_{n/2}, f_l$. For $k \geq 3$, the statement is trivial. For $k = 1$ it follows from Theorem 1.1 that replacing (f_i, f_l) by $(f_i + f_l, 0)$ and also by $(0, f_i + f_l)$ produces the f -vector of some antichain. The number of nonzero coordinates is at most two, so the statement follows by induction unless $f_l = 0$.

Suppose that $f_l = 0$ and note that the complement-free property implies $f_{n/2} \leq n/2$. If $f_i \leq i$ then f is dominated by

$$\left(0, \dots, \overset{i}{i}, \dots, \overset{n/2}{n/2}, \dots, 0\right).$$

Thus suppose that $f_i > i$. From Theorem 1.1 we know that $f_{n/2} \leq (n/2) + i - f_i$. Consider the convex linear combination

$$\frac{2f_{n/2}}{n}(0, \dots, \overset{i}{i}, \dots, \overset{n/2}{n/2}, \dots, 0) + \left(1 - \frac{2f_{n/2}}{n}\right)(0, \dots, \overset{i}{n}, \dots, 0).$$

Its $n/2$ th coordinate is $f_{n/2}$ and its i th coordinate is

$$n - \frac{2(n-i)}{n}f_{n/2} \geq n - \frac{2(n-i)}{n}\left(\frac{n}{2} + i - f_i\right) = f_i + (f_i - i)\frac{n-2i}{n} \geq f_i,$$

as desired.

The very final case is $k = 2$. Using Theorem 1.2 suppose that $f = h_1 + h_2$ with $h_1 > h_2$. If $\delta(h_2) = l$, then we can replace f_l by n and still have the f -vector of a 2-antichain. Similarly, for $\Delta(h_1) = i$ we can set $f_i = n$ and return to some earlier case. Suppose $\Delta(h_1) = \delta(h_2) = n/2$. Let (a, b, c) stand for the vector

$$(0, \dots, \overset{i}{a}, \dots, \overset{n/2}{b}, \dots, \overset{l}{c}, \dots, 0).$$

By the $k = 1$ case we have

$$h_1 \leq \alpha_1(n, 0, 0) + \beta_1(i, n/2, 0), \quad \alpha_1 + \beta_1 = 1, \quad 0 \leq \alpha_1 \leq 1,$$

$$h_2 \leq \alpha_2(0, 0, n) + \beta_2(0, n/2, n-j), \quad \alpha_2 + \beta_2 = 1, \quad 0 \leq \alpha_2 \leq 1.$$

Since $f_{n/2} \leq n/2$, we may assume that $\beta_1 + \beta_2 \leq 1$ and write

$$f = h_1 + h_2 \leq \beta_1(i, n/2, n) + \beta_2(n, n/2, n-j) + (1 - \beta_1 - \beta_2)(n, 0, n),$$

which concludes the whole proof. $\square$

5. Convex hulls in general and applications. For a vector $f = (f_0, \dots, f_n)$ define its normalized blow-up $b(f) = (b_0, \dots, b_n)$ by $b_0 = f_0$, $b_n = f_n$ and $b_i = \binom{n}{i} f_i / n$.

Let us recall the following simple result from [EFK2].

LEMMA. *Blowing up the circle. If D is the dominating set for $\mathcal{B}$ on the circle, and for every $d \in D$ the vector $b(d)$ is the f -vector of some family in $\mathcal{B}$, then the dominating set for $\mathcal{B}$ is $b(D) = \{b(d) : d \in D\}$.*

It is easy to check that the conditions of this lemma are satisfied in the case of Theorems 4.1 and 4.3. Corresponding to a coordinate $b_i = \binom{n}{i}$ one takes $\binom{[n]}{i}$, to $b_i = (i/n)\binom{n}{i} = \binom{n-1}{i-1}$ one takes $\{F \in \binom{[n]}{i} : n \in F\}$, and to $b_i = (n - i/n)\binom{n}{i} = \binom{n-1}{i-1}$ one takes $\binom{[n-1]}{i-1}$.

Thus the following three results hold.

THEOREM 5.1 ([EFK2]). *For the class $\mathcal{B}(k)$ of k -antichains in $2^{[n]}$, $D(\mathcal{B}(k))$ consists of those $(f_0, \dots, f_n)$ for which $f_i = 0$ or $\binom{n}{i}$ for $0 \leq i \leq n$, with the latter holding exactly k times.*

THEOREM 5.2. *For the class $\mathcal{B}^s(k)$ of self-complementary k -antichains in $2^{[n]}$ the dominating set $D(\mathcal{B}^s(k))$ consists of those $f = (f_0, \dots, f_n)$ satisfying $f_i = f_{n-i}$ for all i which either belong to $D(\mathcal{B}(k))$ or f has $k+1$ nonzero coordinates, $f_i = \binom{n}{i}$ holds for $k-1$ of them while the remaining two are $f_j = f_{n-j} = \binom{n-1}{j-1}$ for some $1 \leq j < n/2$.*

THEOREM 5.3. *The dominating set of the class $\mathcal{B}^c(k)$ of complement-free k -antichains in $2^{[n]}$ consists of those $f = (f_0, \dots, f_n)$ which satisfy $f_i f_{n-i} = 0$ for all $i < n/2$ and either belong to $D(\mathcal{B}(k))$ or have $k+1$ nonzero coordinates, n is even, $k-1$ coordinates satisfy $f_i = \binom{n}{i}$, $f_{n/2} = \binom{n-1}{n/2-1}$ and for the remaining coordinate $f_j = \binom{n-1}{j-1}$ or $\binom{n-1}{n-j-1}$ holds accordingly as $j < n/2$ or $j > n/2$.*

Obviously $|\mathcal{F}| \leq \max_{d \in D(\mathcal{B})} |d|$ holds in general for all $\mathcal{F} \in \mathcal{B}$ and an arbitrary class $\mathcal{B}$. This shows that Theorem 5.1 implies the Erdős theorem. Theorems 5.2 and 5.3 translate as follows.

THEOREM 5.4. *Suppose that $\mathcal{F} \subset 2^{[n]}$ is a self-complementary k -antichain. Then (i) or (ii) holds.*

(i) $n+k$ is odd and

$$|\mathcal{F}| \leq \sum_{\frac{n-k}{2} < i < \frac{n+k}{2}} \binom{n}{i}.$$

(ii) $n+k$ is even and

$$|\mathcal{F}| \leq \binom{n-1}{\frac{n-k}{2}-1} + \sum_{\frac{n-k}{2} < i < \frac{n+k}{2}} \binom{n}{i} + \binom{n-1}{\frac{n+k}{2}}.$$

THEOREM 5.5. *Suppose that $\mathcal{F} \subset 2^{[n]}$ is a complement-free k -antichain, $k \leq n/2$. Then (i) or (ii) holds.*

(i) n is odd and

$$|\mathcal{F}| \leq \sum_{0 \leq i < k} \binom{n}{\frac{n+1}{2} + i}.$$

(ii) n is even and

$$|\mathcal{F}| \leq \binom{n-1}{\frac{n}{2}} + \binom{n-1}{\frac{n}{2}+k} + \sum_{0 < i < k} \binom{n}{\frac{n}{k}+i}.$$

Note that both results are best possible. The cases $k = 1$ of them are theorems of Bollobás [B] and Purdy [P], respectively; cf. also [CG]. The optimal families are unique up to complementation for $k \geq 2$.

As another application we deduce a theorem of Alon.

THEOREM 5.6 ([A], [AM]). *Suppose that $\mathcal{F} \subset 2^{[n]}$ is a family of sets in which there are no two sets F_1, F_2 with $F_1 \cap F_2 = \emptyset$ and $|F_1 \cup F_2| \leq n - k$. Then (i) or (ii) holds*

(i) $n - k$ is odd and

$$|\mathcal{F}| \leq \sum_{i > \frac{n-k+1}{2}} \binom{n}{i}.$$

(ii) $n - k$ is even and

$$|\mathcal{F}| \leq \binom{n-1}{\frac{n-k}{2}-1} + \sum_{i > \frac{n-k}{2}} \binom{n}{i}.$$

Proof. Define $\mathcal{F}^c = \{[n] - F : F \in \mathcal{F}\}$. Consider $\mathcal{G} = \mathcal{F} \cap \mathcal{F}^c$. If there are $G, G' \in \mathcal{G}$ with $G \subset G'$, $|G' - G| \geq k$ then $G \cap ([n] - G') = \emptyset$ and $|G \cup ([n] - G')| \leq n - k$ contradict the assumption on $\mathcal{F}$. Thus no such G, G' exist. In particular, $\mathcal{G}$ is a self-complementary k -antichain.

Now $2|\mathcal{F}| \leq 2^n + |\mathcal{G}|$ and the upper bound on $|\mathcal{G}|$ provided by Theorem 5.4 implies the result. $\square$

We can also prove the following two-family version.

THEOREM 5.7. *Let $\mathcal{F}, \mathcal{G} \subset 2^{[n]}$, $0 \leq k < n$ be such that $|F \cup G| > n - k$ holds for all $F \in \mathcal{F}, G \in \mathcal{G}$ with $F \cap G = \emptyset$. Then*

$$(5.1) \quad |\mathcal{F}| + |\mathcal{G}| \leq \sum_{i \geq \lceil \frac{n-k+1}{2} \rceil} \binom{n}{i} + \sum_{i \geq \lceil \frac{n-k+1}{2} \rceil} \binom{n}{i} \text{ holds.}$$

Proof. Consider $\mathcal{F} \cap \mathcal{G}^c$. The conditions imply that it is a k -antichain. Using Erdős' theorem we infer

$$|\mathcal{F}| + |\mathcal{G}| = |\mathcal{F} \cup \mathcal{G}^c| + |\mathcal{F} \cap \mathcal{G}^c| \leq 2^n + \sum_{\lceil \frac{n-k+1}{2} \rceil \leq i \leq \lfloor \frac{n+k-1}{2} \rfloor} \binom{n}{i},$$

proving (5.1). $\square$

Remark. We can easily prove that in both Theorem 5.6 and 5.7 the optimal families are unique.

REFERENCES

- [A] N. ALON (1983), Ph.D. Thesis, Hebrew University, Jerusalem.
- [AM] N. ALON AND V. D. MILMAN (1983), *Embedding of l_x^k in finite dimensional Banach spaces*, Israel J. Math., 45, pp. 265–280.
- [B] B. BOLLOBÁS (1973), *Sperner systems consisting of pairs of complementary subsets*, J. Combin. Theory Ser. A, 15, pp. 363–366.
- [C] G. F. CLEMENTS (1973), *A minimization problem concerning subsets of finite set*, Discrete Math., 4, pp. 123–128.
- [CG] G. F. CLEMENTS AND H. D. O. F. GRONAU (1981), *On maximal antichains containing no set and its complement*, Discrete Math., 33, pp. 239–247.
- [DGH] D. E. DAYKIN, J. GODFREY, AND A. J. W. HILTON (1974), *Existence theorems for Sperner families*, J. Combin. Theory Ser. A, 17, pp. 245–251.
- [E] P. L. ERDÖS (1945), *On a lemma of Littlewood and Offord*, Bull. Amer. Math. Soc., 51, pp. 898–902.
- [EE] K. ENGEL AND P. L. ERDÖS (1989), *Sperner families satisfying additional conditions*, Graphs Combin., 5, pp. 47–56.
- [EFK1] P. L. ERDÖS, P. FRANKL, AND G. O. H. KATONA (1984), *Intersecting Sperner families and their convex hulls*, Combinatorica, 4, pp. 21–34.
- [EFK2] ——— (1985), *Extremal hypergraph problems and convex hulls*, Combinatorica, 5, pp. 11–26.
- [EKR] P. ERDÖS, C. KO, AND R. RADO (1961), *Intersection theorems for systems of finite sets*, Quart. J. Math. Oxford, 12, pp. 313–320.
- [F] P. FRANKL (1984), *A new short proof of the Kruskal–Katona theorem*, Discrete Math., 48, pp. 327–329.
- [K1] G. O. H. KATONA (1972), *A simple proof of the Erdős–Ko–Rado theorem*, J. Combin. Theory Ser. B, 13, pp. 183–184.
- [K2] ——— (1974), *Extremal problems for hypergraphs*, in Combinatorics, Part 2, M. Hall and J. H. van Ling, eds., Mathematical Centre Tracts 55–57, Mathematisch Centrum, Amsterdam, 1974, pp. 13–42.
- [K3] ——— (1968), *A theorem on finite sets*, in Theory of Graphs, P. Erdős and G. O. H. Katona, eds., Akadémiai Kiadó, Budapest, pp. 187–207.
- [Kr] J. B. KRUSKAL (1963), *The number of simplices in a complex*, in Mathematical Optimization Techniques, R. Bellman, ed., University of California Press, Berkeley, pp. 251–278.
- [P] G. PURDY (1977), *A result on Sperner collections*, Utilitas Math., 12, pp. 95–99.
- [S] E. SPERNER (1982), *Ein Satz über Untermengen einer endlichen Menge*, Math. Z., 27, pp. 544–548.

A NEW BASIS FOR TRADES*

G. B. KHOSROVSHAHI† AND S. AJOODANI-NAMINI†

Abstract. A very fast algorithm to generate a semitriangular basis for trades consisting of minimal trades (sparsest basis) is given. By augmenting the elements of this basis, we construct an infinite family of 2-designs.

Key words. trades, minimal trades

AMS(MOS) subject classification. 05B05

1. Introduction and notations. Let v, k, t and λ be positive integers such that $t \leq k \leq v$. Let $v\Sigma i$ denote the set of all i -subsets of a v -set V ($0 < i < v$). An element of $v\Sigma i$ consisting of elements $x_1, \dots, x_i$ is denoted by $x_1 \cdots x_i$. A $t - (v, k, \lambda)$ design is a collection $\mathcal{B}$ of elements of $v\Sigma k$ (called blocks) with the property that every element of $v\Sigma t$ occurs in exactly λ blocks of $\mathcal{B}$. Any collection of blocks is usually referred to as an incomplete block design. A (v, k, t) trade of volume s (or simply a trade) consists of two incomplete block designs T_1 and T_2 each of s blocks such that for every element of $v\Sigma t$, the number of blocks containing this element is the same as in T_1 and T_2 . A trade is void if $s = 0$. The definitions of t -designs and trades allow repeated blocks. Both T_1 and T_2 must cover the same set of elements of V which is called the *foundation* of the trade.

Hwang [5] has shown that

- (1) the minimum foundation size of nonvoid (v, k, t) trade is $k + t + 1$,
- (2) for $v \geq k + t + 1$, the minimum volume of a nonvoid (v, k, t) trade is 2^t , and
- (3) the nonvoid trades with foundation size $k + t + 1$ and volume 2^t exist and have unique structures. These trades are called *minimal trades*.

Following Graver and Jurkat [3], Graham, Li, and Li [2] in a fundamental paper cast these minimal trades in terms of polynomials. Consider the polynomial

$$(*) \quad \phi(x_1, \dots, x_v) = (x_1 - x_2)(x_3 - x_4) \cdots (x_{2t+1} - x_{2t+2})x_{2t+3} \cdots x_{k+t+1}.$$

If we multiply the factors out and identify each x_i with i , then the result is clearly a minimal trade.

Now, let M be the free $\mathbb{Z}$ -module generated by all subsets of V . In this terminology, an incomplete block design D is $\sum_{B \in v\Sigma k} f_B B$, where f_B denotes the multiplicity of the block B in D . Thus an element $\sum f_B B$ is a t -design if $f_B \geq 0$ and such that for t -subsets X of V ,

$$\sum_{B \supseteq X} f_B = \lambda.$$

A submodule of M which is defined by

$$N_k = \left\{ \sum_{B \in v\Sigma k} t_B B \mid \sum_{B \supseteq X} t_B = 0 \quad \forall t\text{-subsets } X \subset V \right\},$$

* Received by the editors October 30, 1988; accepted for publication (in revised form) July 26, 1989. This research was supported in part by a grant from the University of Tehran and a travel grant from the Atomic Energy Organization of Iran.

† Department of Mathematics and Computer Science, University of Tehran, Tehran, Iran.

is precisely the collection of (v, k, t) trades. It has been shown [2] that for $v \geq k + t + 1$ and $k \geq t + 1$, we have

$$\dim N_k = \binom{v}{k} - \binom{v}{t}.$$

Finally we consider the $\binom{v}{t}$ by $\binom{v}{k}$ “ t -set inclusion” matrix $P_{v,k,t} = (p_{X,Y})$ with $|X| = t$, $|Y| = k$ and

$$p_{X,Y} = \begin{cases} 1 & \text{if } X \subseteq Y, \\ 0 & \text{otherwise.} \end{cases}$$

$P_{v,k,t}$ is a full rank matrix [1], [2]. The minimal trades form a sparsest basis for the $\ker P_{v,k,t}$.

In [3] a generating system for a special structure called “ (k, t) -pod” was given. In [2], linearly dependent generators were ruled out and a basis for trades was obtained. In [4], Hedayat and Hwang presented a computer algorithm for generating such a basis. They generated $v!$ permutations and chose from them only $\binom{v}{k} - \binom{v}{t}$ ones which satisfy the criterion given in [2] and then applied it to $(*)$. Clearly, the number of computations performed in this algorithm is of the order of an exponential function and the algorithm suffers from a complexity of computations in time. Later on, Khosrovshahi and Mahmoodian [6] gave another algorithm which utilizes the Gauss elimination procedure to reduce $P_{v,k,t}$ to echelon form. Since these computations are performed in rational arithmetic, it requires a large space of computer memory to store many large matrices. Therefore, this algorithm suffers from a complexity of computations too, not in time, but in space and, moreover, part of the elements of the basis are not minimal trades.

In this paper, we take a fresh look at the elements of the basis of trades and we give an algorithm which (i) does not involve any kind of complexity; (ii) can be easily carried out manually; (iii) the elements of the basis are minimal trades, the whole basis is in *semitriangular* form, and the elements of the basis have some sort of canonical property; (iv) and finally, by augmenting the elements of the basis, it is possible to construct t -designs. We give an example.

Finally, we would like to note that trades are very important and complicated combinatorial objects. In fact, they can be viewed as building blocks of t -designs. If all the trades of all possible volumes could be catalogued, then the problem of existence of t -designs could become much more tractable. Recently, Ray-Chaudhuri and Singhi [7] utilized simple trades (called by them elementary signed t -designs) to prove a very important existence theorem on t -designs.

2. A basis for trades. In this section, we will establish the existence of a semitriangular basis for trades.

DEFINITION 1. Let $w = (w^1, w^2, \dots, w^n)$ be a nonzero element of $\mathbb{Z}^n$ and suppose j , $1 \leq j \leq n$, is the smallest natural number for which $w^j \neq 0$. Then we call j the *starting index* of w .

DEFINITION 2. Let $B = \{w_1, \dots, w_k\}$ be an ordered set of k nonzero vectors in $\mathbb{Z}^n$ and suppose λ_i is the starting index of w_i . Then we call B *semitriangular* if

$$1 \leq \lambda_1 < \dots < \lambda_k \leq n.$$

In terms of trades, these definitions simply mean taking a subset of trades with distinct “smallest blocks” (and ordering them according to the lexicographical order of the smallest blocks).

LEMMA 1. *Every semitriangular set of vectors is linearly independent.*

We are looking for a semitriangular basis, consisting of minimal trades, for trades. Every minimal trade T is associated with the following type of polynomial [2]:

$$(**) \quad P(x_1, \dots, x_0) = (x_{b_1} - x_{c_1}) \cdots (x_{b_{t+1}} - x_{c_{t+1}}) x_{b_{t+2}} x_{b_{t+3}} \cdots x_{b_k},$$

in which all the b_i 's and c_i 's are distinct. First we look for the starting index of T . Since the starting indices of T and $-T$ are equal, therefore, we can assume that

$$(1) \quad \begin{cases} b_i < c_i, & 1 \leq i \leq t+1, \\ b_1 < b_2 < \cdots < b_{t+1}, \\ b_{t+2} < \cdots < b_k. \end{cases}$$

Now, the "smallest block" (i.e., in lexicographical ordering) of T is $B_s = b_1 \cdots b_k$. Therefore, the starting index of T is the number of B_s in the lexicographic order.

LEMMA 2. *The block $B = j_1 \cdots j_k$ such that $j_1 < j_2 < \cdots < j_k$ is the "smallest block" of a minimal trade T if and only if*

$$(2) \quad \begin{cases} j_l \leq v - k - t + 2l - 2, & 1 \leq l \leq t+1, \\ j_l \leq v - k + l, & t+2 \leq l \leq k. \end{cases}$$

Proof (Necessity). Let T be a minimal trade with "smallest block" B_s , then the distinct elements $c_1, \dots, c_{t+1}; b_1, \dots, b_k$ exist which satisfy (1) and $B_s = j_1 \cdots j_k = b_1 \cdots b_k$.

Let $1 \leq m \leq t+1$. From (1) and the definition of B_s , it follows that

$$j_m \leq b_n < c_n, \quad n = m, m+1, \dots, t+1.$$

Therefore, there are at least $t - m + 2$, c_i 's which are greater than j_m . Moreover, there are exactly $k - m$, b_i 's greater than j_m . Hence, at least $k - 2m + 2 + t$ positive integers s exist such that $j_m < s \leq v$. Hence,

$$j_m \leq v - k - t - 2 + 2m.$$

Moreover, we have $j_{t+2} < j_{t+3} < \cdots < j_k \leq v$. Therefore,

$$j_m \leq v - k + m, \quad t+2 \leq m \leq k.$$

Sufficiency. Suppose $j_1, \dots, j_k$ ($j_1 < j_2 < \cdots < j_k$) satisfy (2). We let $b_m = j_m$, for $m = 1, \dots, k$. Now, it suffices to choose distinct $c_1, \dots, c_{t+1}$ such that

$$b_m < c_m \leq v, \quad \text{for } 1 \leq m \leq t+1.$$

First, we observe that

$$b_{t+1} = j_{t+1} \leq v - k - t - 2 + 2t = v - k + t.$$

Hence, there are at least $k - t$ positive integers n such that $b_{t+1} < n \leq v$. From these integers, we have already chosen $k - t - 1$ numbers to be equal to b_i 's. Hence, there is at least one integer distinct from all b_i 's and such that $b_{t+1} < n \leq v$. Therefore, we can define c_{t+1} to be one of these integers. Now, suppose that $c_{m+1}, \dots, c_{t+1}$ are chosen. Since $b_m \leq v - k - t + 2m - 2$, hence there are at least $k + t + 2 - 2m$ positive integers such that $b_i < b_m$, and since for $i < m$, we have $b_i < b_m$ and so far we have chosen $t + 1 - m$ of c_i 's and exactly $k - m$ of these b_i 's are greater than b_m . So there are at least $k + t + 2 - 2m - (t + 1 - m) - (k - m) = 1$ positive integers, n , distinct from b_i 's and c_i 's which satisfy $b_m < n \leq v$. Thus we can choose c_m . This ends the proof.

THEOREM. *There exists a semitriangular basis for trades.*

Proof. It is sufficient to prove that there are exactly $\binom{v}{k} - \binom{v}{t}$ blocks which satisfy (2). To accomplish this, we let $A(v, k, t) = \{j_1, \dots, j_k \mid j_i \text{'s satisfy (2)}\}$. In addition we require that $k \geq t + 1$, and $v \geq k + t + 1$, and also we suppose that $t \geq 0$. Although for $t = 0$ trades are not defined, the set $A(v, k, t)$, for $t = 0$ makes sense.

The proof follows by induction on t .

For $t = 0$, (2) becomes

$$\begin{cases} j_i < v - k, \\ j_l \leq v - k + l, & 1 < l \leq k, \end{cases}$$

and the only block which does not satisfy the above conditions is $(v - k)(v - k + 1) \cdots v$. Therefore, $|A(v, k, 0)| = \binom{v}{k} - 1 = \binom{v}{k} - \binom{v}{0}$.

Now assume that for $t < n$, we have $|A(v, k, t)| = \binom{v}{k} - \binom{v}{t}$. For $t = n$, we define the following sets:

$$H_i = \{j_1 \cdots j_k \mid j_1 \cdots j_k \in A(v, k, t), j_1 = i\},$$

$$K_i = \{i_1 \cdots i_{k-1} \mid i(i_1 + i)(i_2 + i) \cdots (i_{k-1} + i) \in H_i\},$$

for $1 \leq i \leq v - k - n$. Clearly, H_i 's are disjoint, $|H_i| = |K_i|$,

$$A(v, k, n) = \bigcup_{i=1}^{v-k-n} H_i,$$

and

$$|A(v, k, t)| = \sum_{i=1}^{v-k-n} |H_i|.$$

PROPOSITION. $K_i = A(v - i, k - 1, n - 1)$.

Proof.

$$\begin{aligned} i_1 \cdots i_{k-1} \in K_i &\Leftrightarrow i(i + i_1)(i + i_2) \cdots (i + i_{k-1}) \in A(v, k, t) \\ &\stackrel{\text{by (2)}}{\Leftrightarrow} \begin{cases} i \leq v - k - n, \\ i + i_m \leq v - k - n - 2 + 2(m + 1), & \text{for } 1 \leq m \leq n, \\ i + i_m \leq v - k + (m + 1), & \text{for } n + 1 \leq m \leq k - 1. \end{cases} \\ &\Leftrightarrow \begin{cases} i_m \leq (v - i) - (k - 1) - (n - 1) - 2 + 2m, & \text{for } 1 \leq m < n = (n - 1) + 1, \\ i_m \leq (v - i) - (k - 1) + m, & \text{for } (n - 1) + 2 = n + 1 \leq m < k - 1. \end{cases} \\ &\Leftrightarrow i_1 i_2 \cdots i_{k-1} \in A(v - i, k - 1, n - 1). \end{aligned}$$

From the induction hypothesis, we have

$$|A(v - i, k - 1, n - 1)| = \binom{v - i}{k - 1} - \binom{v - i}{n - 1}.$$

From the proposition, it follows that

$$|H_i| = |K_i| = \binom{v - i}{k - 1} - \binom{v - i}{n - 1}.$$

Now, we finish the proof of the theorem:

$$\begin{aligned}
 |A(v, k, n)| &= \sum_{i=1}^{v-k-n} |H_i| \\
 &= \sum_{i=1}^{v-k-n} \left[\binom{v-i}{k-1} - \binom{v-i}{n-1} \right] \\
 &= \sum_{i=1}^{v-k-n} \binom{v-i}{k-1} - \sum_{i=1}^{v-k-n} \binom{v-i}{n-1} \\
 &= \sum_{i=1}^{v-k-n} \left[\binom{v-i+1}{k} - \binom{v-i}{k} \right] - \sum_{i=1}^{v-k-n} \left[\binom{v-i+1}{n} - \binom{v-i}{n} \right] \\
 &= \binom{v}{k} - \binom{k+n}{k} - \left[\binom{v}{n} - \binom{k+n}{n} \right] \\
 &= \binom{v}{k} - \binom{v}{n} \\
 &= \binom{v}{k} - \binom{v}{t}.
 \end{aligned}$$

3. An algorithm. In this section we present an algorithm to produce a semitriangular basis for trades. The algorithm consists of two parts: first it chooses the starting blocks one at a time, then it constructs the associated trade based on the starting block. Clearly, there is no time and space computational complexity involved in this algorithm.

The following is a description of the algorithm. As before, we make the following notational conventions. We will denote every starting block of a minimal trade of the form (**) by $(x_{b_1} - x_{c_1}) \cdots (x_{b_{t+1}} - x_{c_{t+1}}) x_{b_{t+2}} \cdots x_{b_k}$.

Finally, we define

$$\text{Bound}(i) = \begin{cases} v-k-t-2+2i, & i=1, \dots, t+1, \\ v-k+i, & i=t+2, \dots, k. \end{cases}$$

Table 1. An algorithm to produce a basis for trades.

1. For $i = 1, \dots, k$, set $b_i = i$.
2. For $i = t+1, \dots, 1$
 - └ set $A_i = \{b_i + 1, \dots, v\} - \{c_{i+1}, \dots, c_{t+1}, b_{i+1}, \dots, b_{t+1}\}$
 - └ and $c_i = \min A_i$.
3. Construct a trade using (**).
4. Find the largest value of j such that $1 \leq j \leq k$ and $b_j < \text{bound}(j)$.
If such a j exists, then set $m = b_j$.
 - └ For $i = j, \dots, k$
 - └ $b_j = m + 1 + j - i$
 - └ G_0 to 2.

Otherwise stop.

Notes. (1) In Step 2, since we have $b_i \leq v - k - t - 2 + 2i$, hence

$$\begin{aligned}
 |A_i| &\geq (k+t+2-2i) - (t+1-i) \\
 &= k+1-i \geq 1.
 \end{aligned}$$

Therefore $A_i \neq \emptyset$.

(2) Since

$$\text{Bound}(j) < \text{Bound}(j+1) < \cdots < \text{Bound}(k),$$

and if $b_j < \text{Bound}(j)$, then for $i \geq j$,

$$\begin{aligned} b_i &= b_j + 1 + j - i < \text{Bound}(j) + 1 + j - i \\ &\leq \text{Bound}(i). \end{aligned}$$

(3) By choosing c'_i s differently, one can construct other kinds of bases also.

4. An application. In this section, we demonstrate that by suitable augmentation of the elements of the basis of trades produced in § 3, one can produce the family of simple

$$BIB\left(4n+2, \frac{n(4n+1)(4n+2)}{3}, n(4n+1), 3, 2n\right)$$

designs. The existence of these designs was established in [8]. The idea of the algorithm comes from the fact that the existence of this family of simple designs with $b = \frac{1}{2}\binom{v}{k}$ corresponds to the existence of $(4n+2, 3, 2)$ trades with volume $\frac{1}{2}\binom{v}{k}$. These trades, of course, are linear combinations of the elements of the basis of trades.

In what follows via a simple algorithm we produce the above mentioned trades, and it is assumed that T is a $\binom{v}{k}$ integral vector.

Table 2. An algorithm to produce $BIB(4n+2, \frac{n(4n+1)(4n+2)}{3}, n(4n+1), 3, 2n)$ designs.

1. Set $T = 0$.
2. Find the first nonzero component of T . This component corresponds to a starting block B of a base trade. If such a block does not exist, stop.
3. Construct the base trade T_1 which has B as its starting block.
4. If $T + T_1$ is a simple trade, then set $T = T + T_1$, otherwise set $T = T - T_1$.
5. G_0 to 2.

PROPOSITION. The above algorithm generates a trade with volume $\frac{1}{2}\binom{v}{k}$ for the case $v = 4m+2, k = 3, m = 1, 2, \dots$.

Proof. We use induction on n . For $n = 1$, the statement can be easily verified.

Now, supposing that the statement is correct for $n = m$, we will verify it for $n = m+1$. To do this, we study the algorithm very carefully and analyse the resulting trade in every step of the algorithm.

Notation. In what follows we will denote a minimal trade represented by $(x_1 - x_2)(x_3 - x_4)(x_5 - x_6)$ by $(1\ 3\ 5)(2\ 4\ 6)$.

1. Clearly the first two trades on the basis are

$$G_1 = (1\ 2\ 3)(6\ 5\ 4),$$

$$G_2 = (1\ 2\ 5)(4\ 3\ 6).$$

At the end of this step we will have

$$T = T + G_1 - G_2.$$

2. Now, the next trades on the agenda to be considered are

$$T_{1i} = (1\ 2\ 2i+1)(4\ 3\ 2i+2), \quad i = 3, \dots, 2m.$$

These trades are mutually disjoint and we obtain

$$T = T + \sum_{i=3}^{2m} T_{1i}.$$

Note. At the end of this step all blocks of the form $12i$ have been included in T .

3. Now we consider

$$G_3 = (1\ 3\ 4)(2\ 6\ 5).$$

Note that so far 134 and 135 have not appeared in T and that this trade is also disjoint from T . Therefore,

$$T = T + G_3.$$

4. Up to now, out of the blocks of the form $14j$ only 145 and 146 have appeared in T . Hence, the next batch of trades to be considered are of the form

$$T_{2i} = (1\ 4\ 2i + 1)(2\ 5\ 2i + 2), \quad i = 3, \dots, 2m.$$

These trades are mutually disjoint, and they are disjoint from $G_1 - G_2 + G_3$ also. If $i \neq j$, then T_{1i} and T_{2j} are disjoint, but T_{1i} and T_{2i} are not. Therefore, we have

$$T = T - \sum_{i=3}^{2m} T_{2i}.$$

5. At this stage of the game, if a block $abc \in T$, then $b < 6$. Hence, the next class of trades are

$$S_{ii} = (1\ 2i\ 2i + 1)(2\ 2i + 3\ 2i + 2)$$

$$S_{ij} = (1\ 2i\ 2j + 1)(2\ 2i + 1\ 2j + 2),$$

$$i = 3, \dots, 2m; \quad j = i + 1, \dots, 2m + 2.$$

Clearly $(i, j) \neq (i', j')$, then S_{ij} and $S_{i'j'}$ are disjoint. If $i \neq i'$, then S_{ii} and $S_{i'i'}$ are also disjoint. But S_{ii} and S_{ii+1} are not. At the end of this stage we end up with the following T :

$$T = T + \sum_{i=3}^{2m} \left(S_{ii} - S_{ii+1} + \sum_{j=i+2}^{2m+2} S_{ij} \right).$$

6. Now, the only blocks of the form $1ab$ which have not appeared in T so far are those with $4m + 2 \leq b$. They are

$$V_1 = (1\ 4m + 2\ 4m + 3)(2\ 4m + 5\ 4m + 4),$$

$$V_2 = (1\ 4m + 2\ 4m + 5)(2\ 4m + 3\ 4m + 6),$$

$$V_3 = (1\ 4m + 3\ 4m + 4)(2\ 4m + 6\ 4m + 5).$$

Clearly, to have a simple T , we must have

$$T = T + V_1 - V_2 + V_3.$$

With this addition, all the blocks of the form $1ab$ have appeared in T .

7. Out of the blocks of the form $2ab$, only blocks of the form $23i$ and $24i$ have not appeared in T . Therefore, we have

$$T_{3i} = (2\ 3\ 2i + 1)(5\ 4\ 2i + 2),$$

$$i = 3, \dots, 2m + 2.$$

These trades are mutually disjoint and clearly, among all those trades used on the structure of T , T_{1i} and T_{2i} are not disjoint from $T_{1i} - T_{2i}$. Hence we can have

$$T = T + \sum_{i=3}^{2m+2} T_{3i}.$$

8. Now, every block abc which is present in T at this stage satisfies the following conditions:

$$a < 3 \text{ or } c < 7, \text{ or } a = 3 \text{ and } b \leq 5.$$

The trades that now must be considered are the following:

$$F_{ii} = (3 \cdot 2i \cdot 2i + 1)(4 \cdot 2i + 3 \cdot 2i + 2)$$

$$F_{ij} = (3 \cdot 2i \cdot 2j + 1)(4 \cdot 2i + 1 \cdot 2j + 2)$$

$$i = 3, \dots, 2m, \quad j = 2i + 1, \dots, 2m + 2.$$

These trades are disjoint from T and with a similar argument as in Notation 5, we conclude that

$$T = T + \sum_{i=3}^{2m} \left(F_{ii} - F_{ii+1} + \sum_{j=i+2}^{2m+2} F_{ij} \right).$$

9. Finally we consider the following trades:

$$H_1 = (3 \cdot 4m + 2 \cdot 4m + 3)(4 \cdot 4m + 5 \cdot 4m + 4),$$

$$H_2 = (3 \cdot 4m + 2 \cdot 4m + 5)(4 \cdot 4m + 3 \cdot 4m + 6),$$

$$H_3 = (3 \cdot 4m + 3 \cdot 4m + 4)(4 \cdot 4m + 6 \cdot 4m + 5).$$

$$T = T + H_1 - H_2 + H_3.$$

At this stage of the process of completing T , any block abc has appeared in T if and only if $a \leq 4$. The remaining blocks are formed from $4m + 2$ remaining elements, namely $\{5, 6, \dots, 4m + 6\}$. According to our induction hypothesis, we can find a simple trade of volume $\frac{1}{2} \binom{v}{3}$ based on the $v = 4m + 2$ element. Clearly this trade, T' , is disjoint from T and $T + T'$ will be the desired trade.

Acknowledgments. G. B. Khosrovshahi thanks Professor Abdus Salam, the International Atomic Energy Agency, and UNESCO for hospitality at the International Centre for Theoretical Physics, Trieste, and thanks the Atomic Energy Organization of Iran, (AEOI), for making this trip possible. S. Ajoodani-Namini thanks the Research Council, University of Tehran, for partially supporting this research.

REFERENCES

- [1] W. FOODY AND A. HEDAYAT, *On the theory and applications of BIB designs with repeated blocks*, Ann. Statist., 5 (1977), pp. 932–945.
- [2] R. L. GRAHAM, S.-Y. R. LI, AND W.-C. W. LI, *On the structure of t -designs*, SIAM J. Algebraic Discrete Methods, 1 (1980), pp. 8–14.

- [3] J. E. GRAVER AND W. B. JURKAT, *The modular structure of integral designs*, J. Combin. Theory Ser. (A), 15 (1973), pp. 75–90.
- [4] A. HEDAYAT AND H. L. HWANG, *An algorithm for generating basis of trades on t -design*, Commun. Simul. Computa., 12 (1983), pp. 109–125.
- [5] H. L. HWANG, *On the structure of (v, k, t) trades*, J. Statist. Plann. Inference, 13 (1986), pp. 179–191.
- [6] G. B. KHOSROVSHAHI AND E. S. MAHMOODIAN, *A linear algebraic algorithm for reducing the support size of t -designs and to generate a basis for trades*, Commun. Statist. Simul. Computa., 16 (1987), pp. 1015–1038.
- [7] D. K. RAY-CHAUDHURI AND N. M. SINGHI, *On existence of t -designs with large v and λ* , SIAM J. Discrete Math., 1 (1988), pp. 98–104.
- [8] D. G. SARVATE, *All simple BIB's with block size 3 exist*, Ars. Combin., 21A (1986), pp. 257–270.

COMPUTING THE BANDWIDTH OF INTERVAL GRAPHS*

DANIEL J. KLEITMAN† AND RAKESH V. VOHRA‡

Abstract. In this note, an $O(|V|k)$ algorithm is described for determining whether an interval graph on $|V|$ vertices has a bandwidth less than or equal to a given integer k . While the algorithm is not the first to resolve this problem, it does admit a shorter proof of its correctness than a previous algorithm of the same complexity due to Kratsch (*Information and Computation*, 74 (1987), pp. 140–158).

Key words. interval graph, bandwidth

1. Introduction. Let $G = (V, E)$ be a graph with vertex set V and edge set E . A labeling of G is an assignment of distinct integers from $\{1, 2, \dots, |V|\}$ to the vertices of G . The bandwidth of a labeling of G is the maximum difference between the labels of adjacent vertices. The k -bandwidth problem (BP) is the problem of finding a labeling of G with bandwidth less than or equal to some given integer k or of showing that none exists. BP is NP-complete for general graphs, for a proof see Papadimitriou (1976). It has also been shown to be NP-complete for special classes of graphs, see Monien (1983).

Recently, Johnson (1985) conjectured that BP might be NP-complete even when G is an interval graph. This has been shown not to be the case by Kratsch (1987), who describes a $O(|V|^2)$ algorithm for the problem. In this paper we describe another algorithm for BP that admits a succinct proof of its correctness. Its running time is $O(|V|k)$.

2. Preliminaries. A graph is an intersection graph if there exists a one to one correspondence between its vertices and a family V of sets such that two vertices are adjacent in the graph if and only if the corresponding sets have nonempty intersection. If V is a collection of intervals of the real line, then G is called an interval graph (Golumbic (1980), Lekkerkerker (1962)). The collection V is sometimes called the interval model of G .

In what follows we shall assume that an interval graph is always represented in its interval model form. Even if this is not the case, its interval model can be obtained in $O(|E| + |V|)$ steps using the algorithm of Booth and Lueker (1976). Hence, we will use the words vertex and interval and adjacent and overlap interchangeably. Suppose further that the intervals are ordered by increasing value of their right endpoints.

3. The algorithm. The left and right endpoints of interval i will be denoted by a_i and b_i , respectively. The label assigned to i by the algorithm we describe is denoted by $L(i)$. This is initially set equal to zero. Each i will also be assigned another integer, $M(i)$, called its mark. This represents an upper bound on the label that can be assigned to i if we are to have a bandwidth of $\leq k$. This is initially set equal to n . We are now in a position to describe our algorithm for BP which we call k BAND.

k BAND

Step 1. Set $L(i) = 0$ and $M(i) = n$ for all $i \in V$ where $n = |V|$. Set $U = \{i \in V : L(i) = 0\}$ and $q = 0$.

* Received by the editors May 23, 1989; accepted for publication (in revised form) September 20, 1989. This work was done while both authors were at the Institute for Mathematics and its Applications, University of Minnesota, Minneapolis, Minnesota 55455.

† Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139. This research was supported by National Science Foundation grant DMS-86-06255 and Air Force Office of Scientific Research grant OSR-86-0076.

‡ Department of Management Sciences, Ohio State University, Columbus, Ohio 43210.

- Step 2.* Select $i \in U$ with smallest b_i (break ties by selecting the interval with smallest a_i) and set $q = q + 1$.
- Step 3.* Set $L(i) = q$ and $U = U \setminus \{i\}$. If $U = \emptyset$, stop, all vertices have been labeled.
- Step 4.* If $r \in U$ overlaps i and $M(r) = n$ set $M(r) = \min \{L(i) + k, n\}$.
- Step 5.* Let $S_j^q = \{r \in U: M(r) \leq q + j\}$. If $|S_j^q| \leq j$ for all $j \geq k - q + 1$, go to Step 7.
- Step 6.* There is a j such that $|S_j^q| > j$. Stop, for the bandwidth of the graph is $> k$.
- Step 7.* Find the smallest value j_0 , such that $|S_{j_0}^q| = j_0$.
- Step 8.* Select $i \in S_{j_0}^q$ with smallest b_i (break ties as in Step 2). Set $q = q + 1$ and go to Step 3.

A remark is in order on how the sets S_j^q are updated from one iteration to the next. Suppose we have just labeled vertex q . Let j_0 and i be as in Steps 7 and 8, respectively. Then $S_r^{q+1} = S_{r+1}^q \setminus \{i\}$ for all $r \leq k - 1$. Also, $S_k^{q+1} = S_{k-1}^{q+1} \cup \{r \in U: r \text{ overlaps } i \text{ and } M(r) = n\}$. Finally, $S_j^{q+1} = S_k^{q+1}$ for all $k < j < n - q - 1$ and S_{n-q+1}^{q+1} is the set of unlabeled vertices after Step 3.

Suppose k BAND terminates at Step 3. Then, it does so with a labeling of bandwidth $\leq k$. To see this, observe that when a vertex is labeled q by k BAND, all as yet unlabeled vertices that overlap with the vertex just labeled q are assigned a label no greater than $q + k$ by Step 4.

Suppose now that k BAND terminates at Step 6. This means that for some q (consider the first such q) there is a j' such that $|S_{j'}^q| > j'$. We claim that this implies that $|S_k^q| > k$. Suppose not. We know from the definition of q that $|S_j^{q-1}| \leq j$ for all $j \geq k - q + 2$. Now, $|S_j^q| = |S_{j+1}^{q-1}| - 1 \leq j$ for all $j \leq k - 1$. Also, $S_j^q = S_k^q$ for $k \leq j < n - q$. Therefore $|S_j^q| = |S_k^q| \leq k \leq j$. Finally, $|S_{n-q}^q| = n - q$. Hence, $|S_j^q| \leq j$ for all j , which is a contradiction. Thus, if k BAND terminates at Step 6, $|S_k^q| > k$. This means there are at least $k + 1$ vertices unlabeled after a vertex has been labeled q . Thus, one of the vertices in S_k^q must receive a label of at least $q + k + 1$. However, each vertex in S_k^q is adjacent to at least one vertex with a label $\leq q$. Consequently, it is not possible for k BAND to produce a labeling with bandwidth $\leq k$.

THEOREM. *Algorithm k BAND correctly determines if the bandwidth of an interval graph is $\leq k$ in $O(|V|k)$ steps.*

Proof. Clearly, it is sufficient to show that when k BAND terminates at Step 6 after labeling $q < n$ vertices of an interval graph G , there is no labeling L' of G with bandwidth $\leq k$. Suppose not, and let G be the smallest graph which admits a labeling L' of bandwidth $\leq k$, but with k BAND terminating at Step 6 when applied to G .

As k BAND terminates at Step 6, the set of vertices in S_k^q are unlabeled and $|S_k^q| \geq k + 1$. Let β be such that $L(\beta) = q$. Thus for all $\alpha \in S_k^q$, $a_\alpha \leq b_\beta$ and $M(\alpha) \leq q + k$. If $b_\alpha \geq b_\beta$ for all $\alpha \in S_k^q$, then, the set $S_k^q \cup \{\beta\}$ forms a clique of size $\geq k + 2$. Hence, G can admit no labeling of bandwidth $\leq k$. Thus, there exists $\gamma \in S_k^q$ such that $b_\gamma < b_\beta$ and $b_\gamma < a_\alpha$ for some $\alpha \in S_k^q$.

Since γ is unlabeled by $b_\gamma < b_\beta$, it follows that γ was not selected to be labeled during an iteration of Step 8 of k BAND. Hence, there exists an integer j such that $\gamma \notin S_j^{q-1}$ and $|S_j^{q-1}| = j$. Let j be the smallest such integer. Note that $j \leq k - 1$ since $\gamma \in S_k^q$ and $|S_k^q| \geq k + 1$.

The vertices of S_j^{q-1} all overlap the vertex δ where $L(\delta) = q - j - 1 - k$ or some vertex with a smaller right-hand end point. Since $b_\beta \leq b_\alpha$ for all $\alpha \in S_j^{q-1}$ because β was chosen to be labeled in Step 8, it follows that all vertices in S_j^{q-1} overlap δ as well as all vertices λ that overlap β such that $\beta_\lambda \geq b_\gamma$.

Let $L'_{\max} = \max_{\alpha \in S_j^{q-1}} L'(\alpha)$ and $L'_{\min} = \min_{\alpha \in S_j^{q-1}} L'(\alpha)$. Let θ be some vertex that overlaps all elements of S_j^{q-1} . Then $L'(\theta) \geq L'_{\max} - k$ and $L'(\theta) \leq L'_{\min} + k$. This means that the range of feasible values for $L'(\theta)$ is $\leq 2k + L'_{\min} - L'_{\max} + 1 \leq 2k - j + 2$.

Now consider the graph G' formed by removing all vertices in $S_k^q \setminus S_j^{q-1}$ except γ , as well as all vertices with a mark exceeding $q + k$ and extend γ leftwards until it just overlaps δ . Label the vertices of G' with consecutive integers in the same order as those vertices that are labeled by L' . Call this labeling $\bar{L}$. A total of $|S_k^q \setminus S_j^{q-1}| - 1 \geq k - j + 1$ (since $\beta \in S_k^q$) vertices were removed from S_k^q to form G' . As each vertex in S_k^q overlaps all vertices in S_j^{q-1} , the range of distinct labels assigned by $\bar{L}$ to the vertices in G' that overlap all members of S_j^{q-1} in G' is reduced to $k + 1$. Thus, $\bar{L}$ has a bandwidth of $\leq k$, because γ , and any vertices it might newly overlap in G' , overlap all members of S_j^{q-1} in G and hence all have labels in this range; therefore they must have label difference along any edge in G' of at most k .

There are $q + k + 1 - k - 1 + j = q + j$ vertices in G' , all of whom receive a mark of $\leq q + j - 1$ when k BAND is applied to G' , which means that k BAND cannot produce a labeling of G that is of bandwidth $\leq k$ by the discussion before the theorem. This contradicts the definition of G since G' is a smaller graph having bandwidth $\leq k$ for which an application of k BAND terminates after Step 6, and so the required result is obtained.

Finally, note that one pass through Steps 3–5, 7, and 8 takes $O(k)$ steps. This will be repeated at most $|V|$ times. Hence k BAND is $O(|V|k)$.

Acknowledgment. We would like to thank the Institute for Mathematics and its Applications, University of Minnesota, Minneapolis, Minnesota, for its support and hospitality.

REFERENCES

- K. S. BOOTH AND G. S. LUEKER (1976), *Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms*, J. Comput. System Sci., 13, pp. 335–379.
- M. C. GOLUMBIC (1980), *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York.
- D. S. JOHNSON (1985), *The NP-completeness column: an ongoing guide*, 16th edition, J. Algorithms, 6, pp. 434–451.
- D. KRATSCH (1987), *Finding the minimum bandwidth of an interval graph*, Information and Computation, 74, pp. 140–158.
- C. G. LEKKERKERKER AND J. C. BOLAN (1962), *Representations of a finite graph by a set of intervals on the real line*, Fund. Math., 51, pp. 45–64.
- B. MONIEN (1986), *The bandwidth minimization problem for caterpillars with hair length 3 is NP-complete*, SIAM J. Algebraic and Discrete Methods, 7, pp. 505–512.
- C. H. PAPADIMITRIOU (1976), *The NP-completeness of the bandwidth minimization problem*, Computing, 16, pp. 263–270.

ONE-PAGE BOOK EMBEDDING UNDER VERTEX-NEIGHBORHOOD CONSTRAINTS*

SHLOMO MORAN† AND YARON WOLFSTAHL‡

Abstract. The VLSI-related problem of embedding graphs in books is studied. A *book embedding* of a graph $G = (V, E)$ consists of two parts, namely, (1) an ordering of V along the spine of the book, and (2) an assignment of each $e \in E$ to a page of the book, so that edges assigned to the same page do not intersect. In devising an embedding, one seeks to minimize the number of pages used.

This paper addresses a generalization of the book embedding problem, called the *black/white (b/w) book embedding problem*, where a vertex-neighborhood constraint is imposed on the ordering of the vertices. A *b/w graph* is a pair (G, U) , where $G = (V, E)$ is a graph and $U \subseteq V$ is a set of distinguished *black* vertices (the vertices in $V - U$ are called *white*). A *b/w book embedding* of (G, U) is a book embedding of G , where the black vertices are arranged consecutively along the spine. Given a *b/w graph* (G, U) , the *b/w book embedding problem* is that of finding a *b/w book embedding* of (G, U) , such that the number of pages is minimized. The need for *b/w* embeddings may arise, for example, in applications where the input ports of a VLSI chip are to be separated from the output ports.

The main result of this paper is a characterization of the *b/w* graphs that admit a one-page *b/w* embedding. The characterization is given in terms of a set of *forbidden b/w subgraphs*, the absence of which is necessary and sufficient for one-page *b/w* embedding. For a *b/w* graph with none of these forbidden subgraphs, a one-page *b/w* embedding is constructible in linear time. The construction utilizes a technique called *b/w unfolding*, which is a feature of independent interest.

Key words. book embedding, outerplanar graphs, algorithm

AMS(MOS) subject classifications. 05C45, 68Q35, 94C15

1. Introduction. The embedding of a graph $G = (V, E)$ in a book consists of two parts. The first part is an ordering of V along the spine of the book. The second part is an assignment of each edge $e \in E$ to a page of the book, such that if two edges are assigned to the same page, then they do not intersect. Given a graph G , the *book embedding problem* is that of finding a book embedding of G with the minimum possible number of pages.

The notion of book embedding is useful in modeling layout problems arising in VLSI design (cf. [CLR]). Many of the basic properties of book embeddings were established in [BK1]. In subsequent papers, several authors have investigated various aspects of book embeddings, establishing lower and upper bounds on the minimum number of pages required to embed some classes of graphs (cf. [CLR], [H], [R], [Y1]). We note that it is NP-complete to decide whether a given graph admits a 2-page book embedding [W], [CLR]. Also, given a graph $G = (V, E)$, an ordering of V , and an integer k , it is NP-complete to decide whether G admits a k -page book embedding with the vertices placed on the spine in the specified order [GJMP]. On the other hand, a graph admits a one-page book embedding if and only if it is outerplanar [BK1], and this property is decidable in linear time [M].

This paper addresses a generalization of the book embedding problem, called the *black/white (b/w) book embedding problem*, where a vertex-neighborhood constraint is imposed on the ordering of the vertices. A *b/w graph* is a pair (G, U) , where $G =$

* Received by the editors April 5, 1989; accepted for publication (in revised form) September 20, 1989.

† Faculty of Computer Science, Technion-Israel Institute of Technology, Haifa, Israel 32000. This research was supported in part by the Technion V.P.R. fund-C. Wellner Research Fund.

‡ Faculty of Computer Science, Technion-Israel Institute of Technology, Haifa, Israel 32000.

(V, E) is a graph and $U \subseteq V$ is a set of distinguished *black* vertices (the vertices in $V - U$ are called *white*). A *b/w book embedding* of (G, U) is a book embedding of G , where the black vertices are arranged consecutively along the spine. Given a b/w graph (G, U) , the *b/w book embedding problem* is that of finding a b/w book embedding of (G, U) , such that the number of pages is minimized. The need for b/w embeddings may arise, for example, when the input ports of a VLSI chip are to be separated from the output ports. Also, in certain applications external wires are used to connect some vertices in order to, say, meet some connectivity property (cf. [ET]). Putting those vertices consecutively serves to reduce the total wire length, an important consideration in VLSI design. The b/w book embedding problem resembles a problem previously considered in the context of planar VLSI embedding [BK2], [PRS], [Y2]. There, the authors considered the problem of embedding a binary tree on a chip such that all the leaves are on the boundary. In [CLR, Thm. 1.2] a result concerning b/w embedding of certain bipartite graphs where the black vertices are one part of the graph is given.

The main result of this paper is a characterization of the b/w graphs that admit a one-page b/w embedding. The characterization is given in terms of a set of *forbidden b/w subgraphs*, the absence of which is necessary and sufficient for one-page b/w embedding. We note that our characterization can be described in terms of *forbidden b/w minors*, in a sense similar to that of [RS]. For a b/w graph with none of the forbidden b/w subgraphs, a one-page b/w embedding is constructible in linear time. The construction utilizes a technique called *b/w unfolding*. We believe that this technique is a feature of independent interest, and its power is further demonstrated in a companion paper [MW].

The paper is organized as follows. The formal framework is set in § 2. In § 3 we present the *one-page conditions*, and prove that they are necessary conditions for one-page b/w embedding of a b/w graph. The b/w unfolding technique is presented in § 4, and is thereafter used to constructively prove the sufficiency of the one-page conditions. A summary is given in § 5.

2. The formal framework. The following definitions provide a formal framework for the b/w book embedding problem.

A *book* consists of a *spine* and a number of *pages*. The spine of the book is a line. For simple exposition, view the spine as being horizontal. Each page of the book is a half-plane that has the spine as its boundary. Thus, any half-plane is a one-page book, and a plane with a distinguished horizontal line is a two-page book.

Let $G = (V, E)$ be a graph, and let ρ be a linear ordering of V . For each $v \in V$, denote the rank of v with respect to ρ by $\rho(v)$. Let (v_i, v_j) and (v_k, v_l) be two edges of G where $\rho(v_i) < \rho(v_j)$ and $\rho(v_k) < \rho(v_l)$. If either $\rho(v_i) < \rho(v_k) < \rho(v_j) < \rho(v_l)$ or $\rho(v_k) < \rho(v_i) < \rho(v_l) < \rho(v_j)$, we say that (v_i, v_j) and (v_k, v_l) *intersect*, or are *incompatible* (with respect to ρ). Otherwise, we say that (v_i, v_j) and (v_k, v_l) are *compatible*. A set $E' \subseteq E$ is compatible if any two edges $e_1, e_2 \in E'$ are compatible.

A *k-page book embedding* of $G = (V, E)$ may be formally defined as a pair $\varepsilon = (\rho, \pi)$, where ρ is an ordering of V along the spine and π is a partition of E into k compatible sets. Each page of the book is said to *accommodate* one of these compatible sets. An edge accommodated by some page is also said to be *assigned* to that page. A *k-page b/w book embedding* of a b/w graph (G, U) is a *k-page book embedding* of G , say $\varepsilon = (\rho, \pi)$, where the black vertices are consecutive in ρ .

We say that a set of vertices $S \subseteq V$ is *equally-colored* if the vertices of S are either all black, or all white. Vertices that are not equally-colored are said to be *oppositely-colored*.

A basic property of book embeddings is that the order of the vertices on the spine can be viewed as a circular order. That is, if $\varepsilon = (\rho, \pi)$ is a k -page book embedding of G , and ρ' is a circular permutation of ρ , then $\varepsilon' = (\rho', \pi)$ is also a k -page book embedding of G . Thus, if (G, U) admits a k -page b/w embedding, then it admits a k -page b/w embedding in which the black vertices are placed to the left of the white vertices. Such an embedding is called a *regular* b/w embedding.

A *path* in $G = (V, E)$ is either a single vertex or a sequence of distinct vertices $(v_1, v_2, \dots, v_k)$ where $(v_i, v_{i+1}) \in E$ for $1 \leq i \leq k-1$. Let $p_1 = (v_1, \dots, v_k)$ and $p_2 = (w_1, \dots, w_l)$ be two paths in G , where $(v_k, w_1) \in E$. Then $p_1 \cdot p_2$ is the path $(v_1, \dots, v_k, w_1, \dots, w_l)$. A path in a b/w graph is *mixed* if it contains oppositely-colored vertices. A mixed edge is defined similarly. Note that in a regular one-page b/w embedding, mixed edges must nest, that is: for each pair of such edges, the vertices of one edge lie between the vertices of the other.

3. One-page b/w embedding: A characterization theorem. As mentioned in the introduction, deciding whether there is a two-page book embedding of a given graph is NP-complete. It immediately follows that the b/w book embedding problem is also NP-complete. This negative result motivates the study of one-page b/w embeddings. In particular, a question of primary interest is that of characterizing the b/w graphs which admit a one-page b/w embedding. Such a characterization, in terms of a set of forbidden b/w subgraphs, is developed in the rest of this paper. In this section we present the *one-page conditions*, and prove that they are necessary conditions for one-page b/w embedding of a b/w graph. The sufficiency of the conditions will be proved in the following section.

DEFINITION. An *outerplanar representation* of a graph G is a planar embedding of G where the vertices are arranged in an imaginary circle, and they all lie on the external face. A graph $G = (V, E)$ is *outerplanar* if it admits an outerplanar representation.¹ We hereafter identify an outerplanar representation of G with the circular order of V along the circle associated with the representation.

A basic property of outerplanar graphs is the following:

P₁ A graph G is outerplanar if and only if it admits a one-page book embedding [BK1]. In fact, an outerplanar representation of G defines a one-page book embedding of G .

DEFINITION. A *biconnected component* in a graph G is a maximal component where for every three vertices x, y, z there is a path from x to z that avoids y . Note that by this definition, an edge not on a cycle is a biconnected component.

A biconnected outerplanar graph has a unique Hamiltonian cycle [S1], hence a unique outerplanar representation (up to reflection). Given a biconnected outerplanar graph G , let $\Phi(G)$ denote the circular vertex order identified with the unique outerplanar representation of G . Combining the said above with **P₁**, we have the following important property of biconnected outerplanar graphs:

P₂ A biconnected outerplanar graph G has a unique one-page book embedding (up to reflection and circular permutation of the vertices), which is defined by $\Phi(G)$.

DEFINITIONS. Let (G, U) be a b/w graph. A vertex $v \in V$ is a *U-fork* if G contains three vertex-disjoint mixed paths, each of which does not contain v and has its first vertex adjacent to v . Figure 1 depicts a *U-fork*. Given an outerplanar graph G , a *loop* in G is a biconnected component which is not an edge. A loop $L = (V_L, E_L)$ is an *equally-colored*

¹ Traditionally, an outerplanar graph is defined as one that can be embedded in the plane with all the vertices on the external face; however, one easily verifies that the two definitions are equivalent. The reason for our alternative definition is that it is more useful in the context of book embeddings.

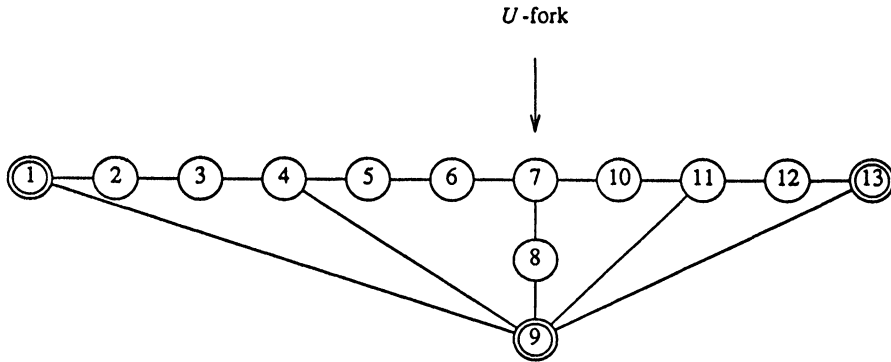


FIG. 1. A graph with a U -fork, where $U = \{1, 9, 13\}$. The three vertex-disjoint mixed paths are $(6, 5, \dots, 1)$, $(10, \dots, 13)$, and $(8, 9)$. Throughout the paper, black vertices are drawn as two concentric circles.

loop if V_L is equally-colored, and it is a *mixed loop* otherwise. Vertices consecutive on $\Phi(L)$ are also said to be *consecutive on L* . A *block* of a mixed loop L is a maximal set of consecutive equally-colored vertices on L . Similarly, a *block* of a mixed path p is a maximal set of consecutive equally-colored vertices of p . A *trail* of a loop L is a path $p = (v_1, \dots, v_n)$ where only the first vertex, v_1 , is on L . The vertex v_1 is called the *base* of p . We also say that p *emanates* from v_1 . Note that by the definition of a loop, trails emanating from different vertices of a loop are vertex-disjoint.

The following conditions will be proved necessary and sufficient for one-page b/w embedding.

DEFINITION. A b/w graph (G, U) satisfies the one-page conditions if

C_1 G admits a one-page embedding (i.e., G is outerplanar).

C_2 There is no U -fork in G .

C_3 In each equally-colored loop, L , there are at most two bases of mixed trails. If L has two such bases, then they are consecutive on L . (See Fig. 2(a).)

C_4 In each mixed loop L

(1) there are exactly two blocks,

(2) each base of a mixed trail is an end-point of a block of L ,

(3) oppositely-colored bases of mixed trails are not consecutive, unless one of them is a single-vertex block, and

(4) there are at most two bases of mixed trails.

(See Fig. 2(b).)

Observe that the one-page conditions can be stated in terms of forbidden b/w subgraphs. For instance, C_1 means that a b/w graph satisfying these conditions cannot contain a b/w subgraph which is not outerplanar, while C_2 implies that such a graph cannot contain a U -fork. The main result of this paper is the following.

CHARACTERIZATION THEOREM (for one-page b/w embedding). A b/w graph (G, U) admits a one-page b/w embedding if and only if it satisfies the one-page conditions.

The proof of the Characterization Theorem is established in two parts, namely, necessity and sufficiency of the one-page conditions. The necessity part is proved in Theorem 1 below. The sufficiency part is deferred to § 4.

THEOREM 1. Assume that (G, U) admits a one-page b/w embedding. Then (G, U) satisfies the one-page conditions.

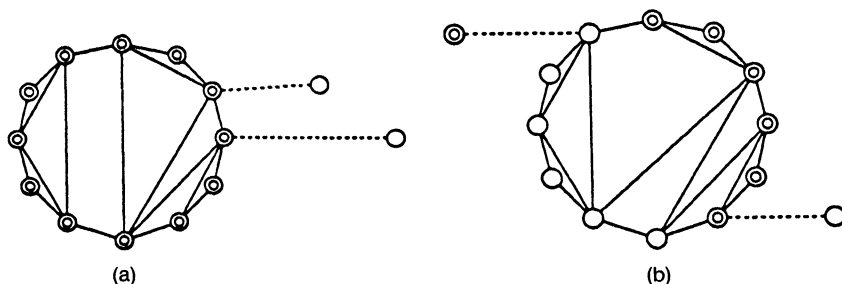


FIG. 2. The one-page loop conditions. (a) An equally-colored loop satisfying C_3 . (b) A mixed loop satisfying C_4 .

Proof. The one-page conditions consist of four conditions, namely C_1 – C_4 . We show that if (G, U) admits a one-page b/w embedding, then each of these conditions must hold. The proof of C_1 is immediate.

Proof of C_2 . Suppose that there is a U -fork, v , in (G, U) , and assume, without loss of generality, that v is black. Then there are three vertex-disjoint mixed paths, each of which does not contain v and has its first vertex adjacent to v . Choose p_i , $1 \leq i \leq 3$, to be three such paths, where in each p_i the last two vertices are the only two oppositely-colored vertices. Let $e_i = (x_i, y_i)$, $1 \leq i \leq 3$ be the mixed edge induced by the last two vertices of p_i , where x_i is black and y_i is white. If (G, U) admits a one-page b/w embedding, then it admits a regular one. In this embedding the mixed edges e_i , $1 \leq i \leq 3$, must nest; without loss of generality, assume that their end-points appear on the spine in the order $(x_1, x_2, x_3, y_3, y_2, y_1)$. The U -fork, v , lies either (a) to the left of x_1 , or (b) between x_1 and x_2 , or (c) between x_2 and x_3 , or (d) to the right of x_3 . In the first two cases, e_2 intersects with any path from v to an end-point of e_3 . In the other two cases, e_2 intersects with any path from v to an end-point of e_1 . Since there are paths from v to end-points of both e_1 and e_3 , it follows that (G, U) does not admit a one-page b/w embedding.

In the following, if S is a subgraph of G , then U/S denotes the set of black vertices of S .

Proof of C_3 . Suppose that in some equally-colored loop, L , there are three bases of mixed trails, or two nonconsecutive ones. Let $(S, U/S)$ be the subgraph of (G, U) induced by the vertices of L and the mixed trails emanating from these bases. Our aim is to show that $(S, U/S)$ does not admit a one-page b/w embedding. If $(S, U/S)$ admits a one-page b/w embedding, then it admits a regular one. In this embedding, by P_2 the vertices of L appear on the spine according to $\Phi(L)$, up to reflection and circular permutation. It follows that the leftmost and rightmost vertices of L on the spine are consecutive on L , and are thus connected by an edge. Call these vertices the L -extremal vertices. Now, since L has three bases of mixed trails, or two nonconsecutive ones, one of these bases must be placed somewhere between the L -extremal vertices. Any mixed trail emanating from this base must intersect with the edge connecting the L -extremal vertices. It follows that $(S, U/S)$, hence (G, U) , does not admit a one-page b/w embedding.

Proof of C_4 . Hereafter, let L be a mixed loop of G .

(1) By P_2 , L has a unique one-page book embedding, up to reflection and circular permutation of the vertices. If there are more than two blocks in L , then this unique one-page book embedding of L is not a b/w embedding of $(L, U/L)$, and hence (G, U) does not admit a one-page b/w embedding.

(2) Suppose that L has a mixed trail $p = (x, \dots, y)$ emanating from a vertex x , such that x is not an end-point of a block of L , and y is the first vertex on p whose color

is opposite to that of x . Let $(S, U/S)$ be the subgraph of (G, U) induced by the vertices of L and p . Our aim is to show that $(S, U/S)$ does not admit a one-page b/w embedding. Assume the converse, and let $\mathcal{E} = (\rho, \pi)$ be a regular one-page b/w embedding of $(S, U/S)$. Let u_1 and u_m be the leftmost and rightmost black vertices of L in ρ , respectively. Similarly, let w_1 and w_n be the leftmost and rightmost white vertices of L in ρ , respectively. (If L has a single-vertex block, then either $u_1 = u_m$ or $w_1 = w_n$.) By $\mathbf{P}_2$, the vertices of L appear in ρ according to $\Phi(L)$, up to reflection and circular permutation. It follows that u_1 and w_n are connected by an edge, and so are u_m and w_1 . Without loss of generality, assume that x is black, so y is white. By the regularity of $\mathcal{E}$, y must be placed either (a) between u_m and w_1 , or (b) between two white vertices w_i and w_{i+1} of L , or (c) to the right of w_n . However, if y is placed according to (a), (b), or (c) above, then the mixed trail p intersects with (u_m, w_1) , (w_i, w_{i+1}) or (u_1, w_n) , respectively. It follows that $(S, U/S)$, hence (G, U) , does not admit a one-page b/w embedding.

(3) Suppose that L has no single-vertex block, and that it has two mixed trails, $p_1 = (u, \dots, x)$ and $p_2 = (w, \dots, y)$, that emanate from oppositely-colored consecutive vertices of L . Without loss of generality, assume that u is black and w is white. Assume also that in each of those two trails, the color of the last vertex is opposite to that of the first vertex. Let $(S, U/S)$ be the subgraph of (G, U) induced by the vertices of L and the trails p_i , $i = 1, 2$. If $(S, U/S)$ admits a one-page b/w embedding, then it admits a regular one, $\mathcal{E} = (\rho, \pi)$. By $\mathbf{P}_2$, the vertices of L appear in ρ according to $\Phi(L)$, up to reflection and circular permutation. Thus, either (a) u is the rightmost black vertex of L on the spine and w is the leftmost white one, or (b) u is the leftmost black vertex of L on the spine and w is the rightmost white one. If case (a) holds, then by the regularity of $\mathcal{E}$, y must be placed to the left of x , so p_1 and p_2 intersect. Suppose then that case (b) holds. In this case, y must be placed to the left of the leftmost white vertex of L , by the regularity of $\mathcal{E}$. Thus, y must be placed to the left of u , for otherwise p_2 intersects some edge of L . By a similar argument, x must be placed to the right of w . It now follows that p_1 and p_2 must intersect. We therefore conclude that $(S, U/S)$, hence (G, U) , does not admit a one-page b/w embedding.

(4) If L has no single-vertex block, then it cannot have more than two bases of mixed trails, for then a contradiction would arise to (1)–(3) above. Thus, one only has to prove the case where L has a single vertex block. Assume then that L has three bases of mixed trails, and that it has a vertex w_0 , without loss of generality, white, that is a single-vertex block. By (1)–(2) above, we find that w_0 is a base of a mixed trail, and so are the two end-points of the black block, called u_1 and u_2 . Let the corresponding mixed trails be $p_0 = (w_0, \dots, u_0)$, $q_1 = (u_1, \dots, w_1)$ and $q_2 = (u_2, \dots, w_2)$, where the color of the last vertex in each trail is opposite to the color of the first vertex. Let $(S, U/S)$ be the subgraph of (G, U) induced by the vertices of L and the trails p_0 , q_1 , and q_2 . Our aim is to show that $(S, U/S)$ does not admit a one-page b/w embedding. Assume the converse, and let $\mathcal{E} = (\rho, \pi)$ be a regular one-page b/w embedding of $(S, U/S)$. By $\mathbf{P}_2$, the vertices of L appear in ρ according to $\Phi(L)$, up to reflection and circular permutation. It follows that the leftmost and rightmost black vertices of L on the spine are u_1 and u_2 ; without loss of generality, let u_1 be the leftmost one. Now, by the regularity of $\mathcal{E}$, w_2 must be placed to the right of u_2 . Similarly, u_0 must be placed to the left of w_0 . Observe that u_0 must be placed to the left of u_1 , for otherwise p_0 and q_2 intersect. Furthermore, u_0 must be placed to the left of u_1 , for otherwise p_0 intersects some edge of L . But then p_0 intersects with q_1 , since w_1 must be placed to the right of u_1 . It follows that $(S, U/S)$, hence (G, U) , does not admit a one-page b/w embedding. $\square$

4. Sufficiency of the one-page condition. In order to complete the proof of the Characterization Theorem, we yet have to prove the sufficiency of the one-page conditions.

For this purpose we introduce a technique for constructing good b/w embeddings, called b/w *unfolding*. Using this technique, we will constructively prove that any b/w graph that satisfies the one-page conditions admits a one-page b/w embedding. Further applications of the b/w unfolding technique are demonstrated in a companion paper [MW].

To describe the b/w unfolding technique (briefly, *unfolding*), we next exemplify the unfolding of a b/w graph consisting of a single mixed path. Recall that a *block* of a mixed path is a maximal set of consecutive equally-colored vertices. Let (G, U) be a b/w graph consisting of a single mixed path $p = \vec{V}_1 \cdot \vec{V}_2 \cdot \dots \cdot \vec{V}_n$, where each $\vec{V}_k = (v_1^k, \dots, v_{j_k}^k)$ is a block. Define $\bar{V}_k$, $1 \leq k \leq n$, to be the path obtained by reversing $\vec{V}_k$. That is, $\bar{V}_k = (v_{j_k}^k, \dots, v_1^k)$. In the following, the paths $\vec{V}_k$, $\bar{V}_k$ are viewed as linear orders of their respective vertices. Given two linear orders $\rho_1 = (v_1, \dots, v_n)$ and $\rho_2 = (w_1, \dots, w_m)$, define $\rho_1 \rho_2$ to be the linear order $\rho_1 \rho_2 = (v_1, \dots, v_n, w_1, \dots, w_m)$. A b/w *unfolding* of $p = \vec{V}_1 \cdot \vec{V}_2 \cdot \dots \cdot \vec{V}_n$ is the ordering

$$\rho(p) = \dots \bar{V}_{2i} \dots \bar{V}_4 \bar{V}_2 \bar{V}_1 \bar{V}_3 \dots \bar{V}_{2i-1} \dots.$$

Two immediate properties of $\rho(p)$ are the following. First, equally-colored vertices of p are consecutive in $\rho(p)$. Second, the edges on p are all compatible with respect to $\rho(p)$. It follows that $\rho(p)$ underlies a one-page b/w embedding of (G, U) . Figure 3 depicts the unfolding.

Given a b/w graph, its structure is typically more complicated than that of the graph described above. In such cases, the unfolding technique essentially involves a careful choice of the path(s) to unfold, such that the rest of the graph can be embedded within a small number of pages. The choice of the specific path(s) to unfold depends on the graph in hand, and is a key to the efficient use of unfolding. In particular, the sufficiency part of the Characterization Theorem is proved through the unfolding of an *ultimate road*, defined below. b/w embedding algorithms that demonstrate different choices of paths to unfold are presented in [MW].

DEFINITIONS. Let (G, U) be a b/w graph. Extend the definition of a trail as follows. Let p and q be paths in G . Say that q is a *trail with respect to p* if $q = (v, v_1, \dots, v_n)$, where v is the only vertex which is both in p and q . The trail q is said to *emanate from v* , and is *strongly mixed* if $q' = (v_1, \dots, v_n)$ is mixed. If there are no strongly mixed trails with respect to p , then p is *ultimate*. A vertex $v \in V$ is a *p -vertex* if it is a vertex on p . A loop L is a *p -loop* if some vertex on L is a p -vertex. Say that p is a *road* if the p -vertices of every p -loop, L , appear consecutively on p , according to $\Phi(L)$.

The concept of *ultimate road* plays a key role in proving the sufficiency of the one-page conditions. In fact, the proof is established by means of an algorithm that unfolds an ultimate road. In the following, we develop a scheme for finding an ultimate road in a b/w graph (G, U) that satisfies the one-page conditions. Rather than searching for an ultimate road in (G, U) , we find an ultimate path in a tree called the *superstructure* of (G, U) , and then transform that path to an ultimate road in (G, U) . This ensures that the obtained road has some additional properties that are desirable for the algorithm. As a first step, we show below that our attention can be restricted to *trimmed* graphs.

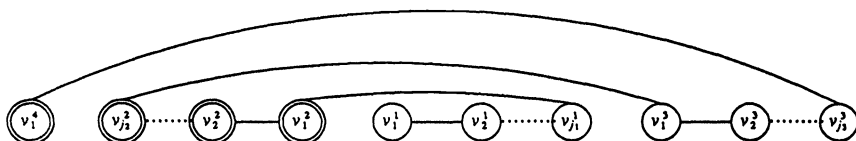


FIG. 3. Unfolding p .

DEFINITIONS. Let (G, U) be a b/w graph where $G = (V_G, E_G)$ is connected. A vertex $v \in V_G$ *separates* x from y , where $x, y \in V_G, x \neq v, y \neq v$, if all the paths connecting x and y pass through v . In this case, we also say that v is a *separation vertex*. If X and Y are disjoint subsets of V_G , and for each $x \in X$ and $y \in Y$, v separates x from y , we say that v *separates* X from Y . A *separatable subgraph* of G is a (possibly disconnected) subgraph $A = (V_A, E_A)$, such that some separating vertex $v \in V_G - V_A$ separates V_A from $(V_G - V_A) - \{v\}$. If, in addition, $V_A \cup \{v\}$ is equally-colored, we say that A is a *U-separatable subgraph* of G , and that v *U-separates* it from G . If G contains no *U-separable subgraphs*, then (G, U) is said to be *trimmed*. We note that *trimming* (G, U) , i.e., bringing it to a state where no *U-separable subgraphs* exist, can be done in $O(|E|)$ time, using a DFS (Depth First Search)-based algorithm for finding biconnected components [E].

DEFINITION. Let $G = (V, E)$ be an outerplanar graph, let $v_1 \in V$ be a vertex of G and let $\rho = (v_2, v_3, \dots, v_n)$ be a linear order of $V - \{v_1\}$. We say that ρ is *canonical with respect to* v_1 if $\rho' = (v_1, v_2, \dots, v_n)$ underlies a one-page book embedding of G (or, equivalently, if $\rho'' = (v_2, \dots, v_n, v_1)$ underlies a one-page book embedding of G).

Given an outerplanar graph $G = (V, E)$ and $v \in V$, a canonical ordering of V with respect to v always exists, by property P_1 . Such an ordering can be found using the algorithm of [M] for recognizing outerplanar graphs.

PROPOSITION 1. *Let (G, U) be a b/w graph, where $G = (V, E)$ is outerplanar. Let (G', U') be the b/w graph obtained by trimming (G, U) , where $G' = (V', E')$ and $U' = U \cap V'$. Then, given a one-page b/w embedding of (G', U') , one can construct a one-page b/w embedding of (G, U) in linear time.*

Proof. For each vertex $v \in V'$, define $A(v)$ to be the largest subgraph which v *U-separates* from G , such that $A(v)$ contains only vertices of $V - V'$ (if such subgraph exists). Also, if $A(v)$ is the subgraph (V_A, E_A) , then $E_A^*(v)$ is the edge set of the subgraph induced on $\{v\} \cup V_A$. The one-page b/w embedding of (G, U) is constructed as follows. The vertices of V' are ordered as in the one-page embedding of (G', U') . The vertices of each subgraph $A(v)$ are placed immediately to the left of v , in a canonical ordering with respect to v . We next show that the edges of G are all compatible.

- By definition of canonical ordering, the edges of each set $E_A^*(v)$ are compatible.
- The vertices of each $A(v)$ are arranged consecutively near v , so the edges of $E_A^*(v)$ are compatible with those of G' .
- For different v and w , the subgraphs $A(v)$ and $A(w)$ are vertex-disjoint, and the vertices of each such subgraph are placed consecutively. Thus, the edges of any two sets $E_A^*(v)$ and $E_A^*(w)$ are compatible.

Finally, the subgraphs $A(v)$ can all be found in $O(|E|) = O(|V|)$ time, using the DFS-based algorithm for finding separating vertices and biconnected components [E]. A canonical ordering of the vertices of each $A(v) = (V_A, E_A)$ can be found in $O(|V_A|)$ time, using the algorithm of [M]. The linearity of the construction follows. $\square$

Recall that our main goal in this chapter is to prove that the one-page conditions are sufficient for one-page b/w embedding. By Proposition 1, we need only consider trimmed graphs for that matter.

DEFINITIONS. Let (G, U) be a trimmed b/w graph that satisfies the one-page conditions. Let $\Lambda(G) = \{L_1, \dots, L_k\}$ be the set of the loops of G , and let $S(G) = \{s_1, \dots, s_p\}$ be the set of the separation vertices and leaves (vertices of degree one) of G . An edge $e \in E$ is *simple* if it is not an edge of a loop.

The *superstructure* of (G, U) is the pair $H(G, U) = (T(G), f)$, defined as follows (see Fig. 4):

- (1) $T(G)$ is the tree $T(G) = (V_T, E_T)$, where $V_T = V_\Lambda \cup V_S$, $V_\Lambda = \{v_{L_i} \mid L_i \in \Lambda(G)\}$, $V_S = \{v_{s_i} \mid s_i \in S(G)\}$, and $E_T = \{(v_{s_i}, v_{s_j}) \mid s_i, s_j \in S(G), (s_i, s_j) \text{ is a simple edge in } E\} \cup \{(v_{s_i}, v_{L_j}) \mid s_i \in S(G), L_j \in \Lambda(G), s_i \text{ is a vertex of } L_j\}$.
- (2) $f: V_T \rightarrow \{B, W, B\&W\}$ is an assignment of *supercolors* (*s-colors*) to V_T , where
 - For each black [white] $s_i \in S(G)$, $f(v_{s_i}) = B$ [$f(v_{s_i}) = W$].
 - For each loop $L_i = (V_i, E_i)$, let $W_i = V_i - \{s_j \mid s_j \in S(G)\}$. If W_i is equally-colored, say black [white], then $f(v_{L_i}) = B$ [$f(v_{L_i}) = W$]. Otherwise, $f(v_{L_i}) = B\&W$.

Note that each separation vertex on a loop is a base of a mixed trail, since (G, U) is trimmed. Moreover, since (G, U) satisfies the one-page conditions, each loop has at most two separation vertices (see C_3 – C_4). Thus f is well-defined, as $W_i \neq \emptyset$ for each loop L_i . Two vertices $v_i, v_j \in V_T$ are said to be *oppositely s-colored* if $f(v_i) = B$ and $f(v_j) = W$. A path in $H(G, U)$ is *h-mixed* if it contains oppositely *s-colored* vertices. A *strongly h-mixed trail* with respect to a path p is a trail with respect to p , say $q = (v, v_1, \dots, v_n)$, where $q' = (v_1, \dots, v_n)$ contains oppositely *s-colored* vertices. A path p in $H(G, U)$ is *h-ultimate* if there are no strongly *h-mixed* trails with respect to p .

LEMMA 1. *Let (G, U) be a trimmed b/w graph that satisfies the one-page conditions. Then an h-ultimate path in $H(G, U)$ can be found in linear time.*

Proof. An algorithm for finding the desired path is described (see Fig. 5).

Step 1. Find the loops of G , the separation vertices, and the leaves. Then, construct the superstructure $H(G, U)$. If $H(G, U)$ contains $B\&W$ vertices, let $p_{B\&W}$ be the path in $H(G, U)$ that starts and ends in $B\&W$ vertices, and contains all the other $B\&W$ vertices (that such a path exists is proved below).

Step 2. If $H(G, U)$ contains $B\&W$ vertices, extend $p_{B\&W}$ to obtain a path $p = p_1 \cdot p_{B\&W} \cdot p_2$ of the maximum possible length. Otherwise, find a path p of maximum length in $H(G, U)$. The resulting path is *h-ultimate*.

Correctness. In Step 1, if $H(G, U)$ contains $B\&W$ vertices, then there is indeed a path $p_{B\&W}$ as defined. For otherwise, the minimal subtree containing all $B\&W$ vertices has a vertex of degree at least 3, v_{s_i} , which corresponds to a separation vertex s_i in G . The latter separation vertex is easily seen to be a *U-fork* in G , contradicting the fact that

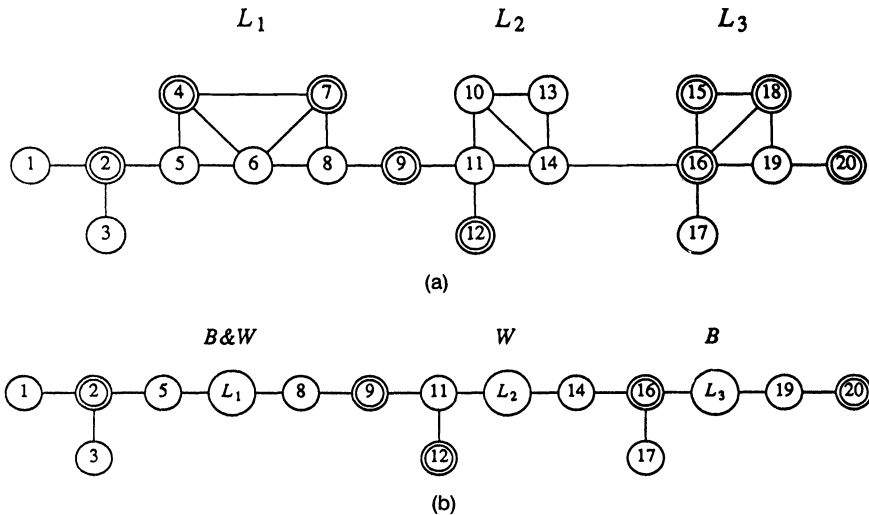


FIG. 4. Definition of hypertree. (a) A trimmed outerplanar graph, G , where $U = \{2, 4, 7, 9, 12, 15, 16, 18, 20\}$. (b) The hypertree $H(G, U)$.

(G, U) satisfies the one-page conditions. Consider the path p obtained in Step 2. Note that if p starts [ends] in a non- $B\&W$ vertex, then this vertex is followed [preceded] in p by an oppositely s -colored vertex, since (G, U) is trimmed. We can thus view p as the path $p = (v_1, \dots, v_n)$, where (a) v_1, v_2 are oppositely s -colored or $f(v_1) = B\&W$, and (b) v_{n-1}, v_n are oppositely s -colored or $f(v_n) = B\&W$. We next show, using the assumption that (G, U) satisfies the one-page conditions, that there is no strongly h -mixed trail with respect to p . Observe first that no such trail emanates from $v \in \{v_1, v_2, v_{n-1}, v_n\}$, for then the length of p can be increased. Suppose that a strongly h -mixed trail emanates from $v \in \{v_3, \dots, v_{n-2}\}$. Then v has in p two oppositely s -colored predecessors, or a $B\&W$ predecessor. Similarly, v has in p two oppositely s -colored successors, or a $B\&W$ successor. Note that v either corresponds to a loop v_{L_i} in G , or to a separation vertex v_{s_i} in G . We can verify that in the first case L_i contains a U -fork, and in the second case s_i is itself a U -fork. Either way, a contradiction arises to the fact that (G, U) satisfies the one-page conditions. It follows that the path p is indeed an h -ultimate path.

Complexity. The loops and the separation vertices of G can be found in linear time, using a DFS-based algorithm for finding biconnected components [E]. Finding the leaves, constructing $H(G, U)$, and constructing $p_{B\&W}$ (by initiating DFS from some $B\&W$ vertex) is easily seen to be also implementable within the same time bound. If $p_{B\&W}$ is not empty, then it can be extended to a path of the maximum possible length using two applications of BFS (Breadth First Search) [E], one from each of its end-points. Otherwise, if $p_{B\&W}$ is empty, a longest path in $H(G, U)$ can also be found using two applications of BFS—one from an arbitrary vertex v , and the second from a leaf furthest from v . As the time complexity of BFS is linear, the lemma follows. $\square$

LEMMA 2. *Assume that (G, U) is trimmed and satisfies the one-page conditions. Given an h -ultimate path in $H(G, U)$, an ultimate road in (G, U) is constructible in linear time.*

Proof. Some definitions are first required. Recall that $\Phi(L)$ denotes the circular vertex order identified with the unique outerplanar representation of a loop L . Let L be an *equally-colored* loop of G with two bases of mixed trails, say x and y . By C_3 , these bases are consecutive on L . For such L , define $\Phi_{x \rightarrow y}(L)$ to be the road in G that starts in x , contains all the vertices of L , and ends in y . Next, let L be a *mixed* loop of G . If L has two bases of mixed trails, say x and y , then they are end-points of blocks, by C_4 . For such L , let $\Phi_{x \rightarrow y}(L)$ denote the road that starts in x , proceeds through all the vertices of the oppositely colored block, and ends in y . If a mixed loop L has a single base, x , of a mixed trail, let $\Phi_{\bar{x}}(L)$ denote the road that starts in x , proceeds through all the vertices of the oppositely-colored block (in clockwise direction, if x is a single-vertex block), and ends in the end-point of that block. Define $\Phi_{\bar{x}}(L)$ to be the road obtained by reversing $\Phi_{\bar{x}}(L)$.

We can now present the construction. Let p be the given h -ultimate path in $H(G, U)$. An ultimate road in (G, U) , called q , is constructed in the following manner. If p consists of a single vertex, then G is biconnected, or, equivalently, consists of a single loop L . In this case, choose x and y to be two equally-colored end-points of a block of $\Phi(L)$, and set $q = \Phi_{x \rightarrow y}(L)$. Otherwise, proceed as follows. Set $q = p$. For each vertex v_L in q that corresponds to a loop L in G , replace v_L and its neighbors, if such exist, by

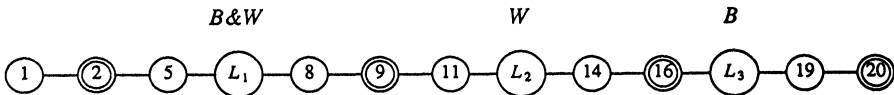


FIG. 5. The h -ultimate path obtained by applying Lemma 1 to the hypertree in Fig. 4(b).

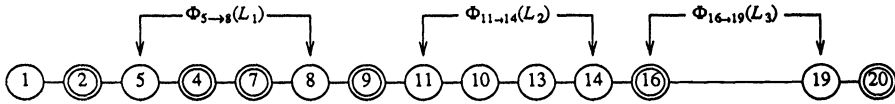


FIG. 6. The ultimate road obtained in Lemma 2 for the graph of Fig. 4(a).

a road r defined as follows. If v_L has one neighbor to its left [right], say x , then $r = \Phi_{\bar{x}}(L)$ [$r = \Phi_{\bar{x}}(L)$]. If v_L has two neighbors, say x and y to its left and right, respectively, then $r = \Phi_{x \rightarrow y}(L)$. That q is an ultimate road in (G, U) follows directly from the fact that p is an h -ultimate path in $H(G, U)$. The construction can clearly be done in $O(|V|)$ time. It is exemplified in Fig. 6. $\square$

DEFINITIONS. Let (G, U) be a b/w graph where $G = (V, E)$ is outerplanar, and let p be a road in (G, U) . Recall that a p -vertex is a vertex on p . Say that a vertex $v \in V$ is *free* (with respect to p) if v is not a p -vertex. A p -vertex v of a loop L is a p -terminal if v is the first, or last, vertex of L that appears in p . If $v \in V$ is a p -vertex of a loop L , but not a p -terminal of L , then v is called an *internal p -vertex* of L . A loop with two p -terminals is called a *2-terminal loop*. Note that a vertex v can be a p -terminal of at most two 2-terminal loops. In this case, v is the first p -terminal of one loop and the second p -terminal of the other. A *normalized ultimate road* is an ultimate road p that has the following two properties:

N_1 there is no 2-terminal equally-colored loop with free vertices, and

N_2 there is no 2-terminal mixed loop with oppositely-colored internal p -vertices.

LEMMA 3. Assume that (G, U) is trimmed and satisfies the one-page conditions. Let p be an ultimate road in (G, U) , as constructed in Lemma 2 from an h -ultimate path of $H(G, U)$. Then q is normalized.

Proof. The proof follows from the definition of the roads $\Phi_{x \rightarrow y}(L)$ in Lemma 2. $\square$

LEMMA 4. Assume that (G, U) satisfies the one-page conditions, and let p be a normalized ultimate road in (G, U) , as constructed in Lemma 2 from an h -ultimate path in $H(G, U)$. Let $\rho(p)$ be an unfolding of p , and let L be a 2-terminal loop. Then,

(1) the p -vertices of L are arranged in $\rho(p)$ according to $\Phi(L)$, such that equally-colored vertices are arranged consecutively.

(2) the free vertices of L (if any) can be added to $\rho(p)$, in such a way that all the vertices of L are arranged according to $\Phi(L)$, and equally-colored vertices are arranged consecutively.

Proof. Recall that an unfolding of $p = \vec{V}_1 \cdot \vec{V}_2 \cdot \vec{V}_3 \cdot \dots$ is the order $\rho(p) = \dots \vec{V}_{2i} \dots \vec{V}_4 \vec{V}_2 \vec{V}_1 \vec{V}_3 \dots \vec{V}_{2i-1} \dots$. Note that $\rho(p)$ preserves the order of equally-colored vertices, a property hereafter referred to as the *order preserving* property of $\rho(p)$. This property together with N_1 directly implies the lemma when L is equally-colored, so assume that L is mixed. Let $\Phi(L) = p_1 \cdot p_2$, where $p_1 = (v_1, v_2, \dots, v_n)$ contains the p -vertices, starting from the first p -terminal, and $p_2 = (v_{n+1}, \dots, v_m)$ contains the free vertices. We first consider the case where v_1 belongs to an odd-numbered block of p .

(1) Using N_2 and the fact that the free vertices are equally-colored (by the ultimateness of p), we find that the vertices $(v_2, \dots, v_{n-1})$, together perhaps with v_n , constitute a block of L . Furthermore, by the definition of $\Phi_{x \rightarrow y}(L)$ in Lemma 2, the color of that block is opposite to that of v_1 . By the order preserving property of $\rho(p)$ and the fact that v_1 belongs to an odd-numbered block of p , the order of $v_1, \dots, v_{n-1}, v_n$ in $\rho(p)$ is either $\rho_1 = (v_{n-1}, v_{n-2}, \dots, v_1, v_n)$ or $\rho_2 = (v_n, v_{n-1}, \dots, v_1)$, depending on whether v_n has the color of v_1 or the opposite color, respectively. Whichever the case is, the obtained order is as claimed.

(2) Place the free vertices of L , in the order $v_m, v_{m-1}, \dots, v_{n+1}$, immediately to the right of v_1 . Thus, if the p -vertices of L appear in $\rho(p)$ in the order ρ_1 or ρ_2 defined above, then the obtained order for all the vertices of L is $\rho'_1 = (v_{n-1}, v_{n-2}, \dots, v_1, v_m, v_{m-1}, \dots, v_{n+1}, v_n)$ or $\rho'_2 = (v_n, v_{n-1}, \dots, v_1, v_m, v_{m-1}, \dots, v_{n+2}, v_{n+1})$, respectively. The free vertices of L are equally-colored, by the ultimateness of p . Their color is equal to that of v_1 , by the construction of the roads in Lemma 2. Thus the obtained order is as claimed. (As an example, consider loop L_1 of Fig. 4(a). Here, $(v_1, \dots, v_n) = (5, 4, 7, 8)$ and $v_{n+1} = v_m = \{6\}$. The order obtained in the lemma for the vertices of L_1 is $\rho'_1 = (7, 4, 5, 6, 8)$, depicted in Fig. 7).

Symmetrically, we can verify that if v_1 belongs to an even-numbered block of p , then the obtained order is either $\rho'_1 = (v_n, v_{n+1}, \dots, v_{m-1}, v_m, v_1, \dots, v_{n-1})$ or $\rho'_2 = (v_{n+1}, \dots, v_{m-1}, v_m, v_1, \dots, v_{n-1}, v_n)$, depending on whether v_n has the color of v_1 or the opposite color, respectively. Either way, the order is as claimed in the lemma. $\square$

The sufficiency of the one-page conditions is established in Theorem 2 below. The proof is constructive, and relies on the unfolding technique. Throughout the proof, the unfolding of a path p is viewed as an iterative procedure for placing the vertices of p on the spine according to $\rho(p)$. This procedure scans the vertices of p one by one, placing each vertex on the spine as it is encountered. A vertex of an odd-numbered block is placed to the right of the rightmost vertex currently on the spine. Similarly, a vertex of an even-numbered block is placed to the left of the leftmost vertex currently on the spine.

THEOREM 2. *Given a trimmed b/w graph (G, U) that satisfies the one-page conditions, a one-page b/w embedding of (G, U) can be found in linear time. (If (G, U) does not satisfy the one-page conditions, then one of the forbidden b/w subgraphs can be found within the same time bound.)*

Proof. A linear time algorithm is developed below, which produces a one-page b/w embedding of any trimmed b/w graph (G, U) that satisfies the one-page conditions. The algorithm is based on unfolding a normalized ultimate road, p , in (G, U) . Informally, the algorithm scans the vertices of p one by one, placing each p -vertex v according to the order prescribed by unfolding p (recall the iterative unfolding procedure described above). Immediately after placing v , the algorithm places the vertices of a subgraph associated with v , called $A(v)$ (to be defined). The free vertices of the 2-terminal mixed loops are not contained in any subgraph $A(v)$. For each such loop, L , these vertices are placed after the second p -terminal of L is placed.

Some definitions are required next. Let p be an ultimate road in (G, U) . For each p -vertex v , let $A(v)$ be the largest subgraph which v separates from G , such that $A(v)$ contains no vertices of p . The facts that (G, U) is trimmed and that p is ultimate imply that for each p -vertex, v , the vertices of $A(v)$ are equally-colored, and their color is opposite to that of v .

We are now ready to present the algorithm.

Algorithm one-page

Input: A trimmed b/w graph (G, U) , where $G = (V, E)$.

Output: A 1-page b/w embedding of (G, U) , provided (G, U) satisfies the one-page conditions.

Method:

- 1 Apply Lemma 1 to find an h -ultimate path q in $H(G, U)$. Apply Lemma 2 to transform q to a normalized ultimate road in (G, U) . Let $p = (v_1^1, \dots, v_{j_1}^1, v_1^2, \dots, v_{j_2}^2, \dots, v_1^n, \dots, v_{j_n}^n)$ be the obtained road.
- 2 **While** the vertices of p are not all placed on the spine,
- 3 **do**

- 4 Let v_l^k be the leftmost vertex of p that is not yet placed on the spine.
- 5 **If** k is odd [even], **then** place v_l^k on the spine such that it is the rightmost [leftmost] vertex.
- 6 **If** k is odd [even], **then** place the vertices of $A(v_l^k)$ to the left [right] of the currently leftmost [rightmost] vertex, in a canonical ordering with respect to v_l^k .
- 7 **If** v_l^k is the second p -terminal of a mixed 2-terminal loop $L = (V_L, E_L)$, **then** place the (equally-colored) free vertices of L as prescribed in Lemma 4, such that V_L is ordered according to $\Phi(L)$.
- 8 **od**

Analysis of the algorithm. First, note that the algorithm places each p -vertex immediately near an equally-colored vertex (provided such a vertex exists on the spine). The vertices, the subgraphs $A(v)$, and the free vertices of the 2-terminal loops are also placed in such a way. It follows that on completion of the algorithm, equally-colored vertices are arranged consecutively. We next show that the edges can all be assigned to a single page.

By $Old(v)$ we denote the set of edges $(x, y) \in E$ such that x and y were placed before the vertex v . Let v_1 and v_2 be two consecutive vertices on p , where v_1 precedes v_2 . Then $New(v_1) = E_1 \cup E_2$, where E_1 is the set of edges incident to v_1 , and E_2 is the set of edges $(x, y) \in E$ such that x and y were placed after v_1 , and before v_2 . Hereafter, if $A(v)$ is the subgraph (V_A, E_A) , then $E_A^*(v)$ is the edge set of the subgraph induced on $\{v\} \cup V_A$.

To prove that one page suffices for the embedding, we use induction on the number of vertices placed by the algorithm on the spine. Specifically, the induction hypothesis states that *for each p -vertex v_l^k , the set $Old(v_l^k) \cup New(v_l^k)$ is compatible*. As the induction basis can be easily verified, we proceed to the induction step. Suppose that the hypothesis holds for the first $m - 1$ p -vertices placed by the algorithm. Consider the m th p -vertex, v_l^k . Let w be the predecessor of v_l^k on p . Since $Old(v_l^k) = Old(w) \cup New(w)$, the hypothesis implies that $Old(v_l^k)$ is compatible. As no vertex x , where $(x, y) \in New(v_l^k)$, is ever placed between vertices of $Old(v_l^k)$, it suffices to show that $New(v_l^k)$ is compatible.

Case 1. v_l^k is not a vertex of a 2-terminal loop. Here, $New(v_l^k) = \{(w, v_l^k)\} \cup E_A^*(v_l^k)$. The edge (w, v_l^k) is nested within $E_A^*(v_l^k)$, whose compatibility follows from the definition of canonical ordering. Thus, $New(v_l^k)$ is compatible.

Case 2. v_l^k is the first p -terminal of a 2-terminal loop. If v_l^k is also the second p -terminal of some 2-terminal loop, see Cases 4–5 below. Otherwise, the proof is identical to that of Case 1.

Case 3. v_l^k is an internal vertex of a 2-terminal loop $L = (V_L, E_L)$. Since (G, U) is trimmed, the p -terminals of L are bases of mixed trails. Thus, v_l^k is not a base of a mixed trail, for otherwise L would have three such bases, contradicting the fact that (G, U) satisfies the one-page conditions (see C_3 – C_4). It follows that $New(v_l^k)$ is just the set of the edges of L that are incident to v_l^k . This set is compatible, since the vertices of L that are currently on the spine are arranged, by Lemma 4, according to $\Phi(L)$.

Case 4. v_l^k is a second p -terminal of a mixed 2-terminal loop $L = (V_L, E_L)$. Let F be the set of free vertices of L . Observe that $New(v_l^k) = E'_L \cup E_A^*(v_l^k)$, where E'_L is the set of edges of L incident to vertices of $F \cup \{v_l^k\}$. By Lemma 4, $F \cup \{v_l^k\}$ are placed such that V_L is ordered according to $\Phi(L)$, so E'_L is compatible. The set E'_L is nested within $E_A^*(v_l^k)$, whose compatibility follows from the definition of canonical ordering. Thus, $New(v_l^k)$ is compatible.

Case 5. v_l^k is a second p -terminal of a equally-colored 2-terminal loop. This case is similar to Case 4.

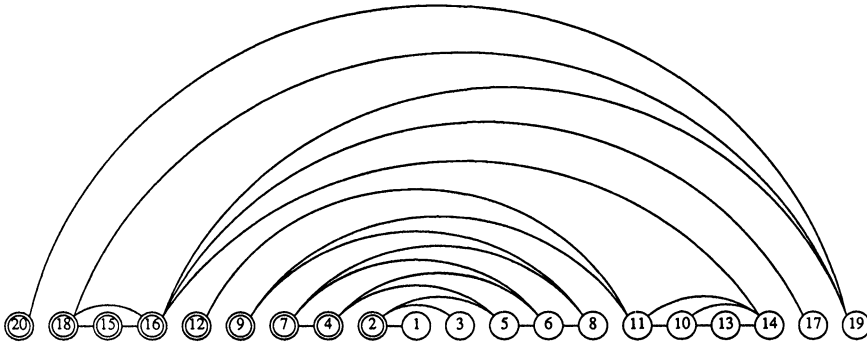


FIG. 7. Applying Algorithm $\mathcal{A}$ to the graph of Fig. 4(a) (unfolding the ultimate road of Fig. 6).

We next consider the complexity of the algorithm. The road p can be found in $O(|V|)$ time, by Lemmas 1–2. Assume that at any given time within the execution, the end-points of the spine are recorded. The overall time required for this operation is $O(|V|)$. The embedding of each subgraph $A(v) = (V_A, E_A)$ consists of (1) finding this subgraph, (2) finding canonical orderings of its vertices, and (3) placing the vertices of these subgraphs, using the stored information on the end-points of the spine. Each such operation requires $O(|V_A|)$ time (see the proof of Proposition 1). Since the subgraphs $A(v)$ are vertex-disjoint, the overall embedding process (Steps 2–8) requires $O(|V|)$ time. Thus, the time complexity of the algorithm is $O(|V|)$. Figure 7 depicts the resulting b/w embedding.

Finally, we note that the algorithm can be extended such that if (G, U) does not satisfy the one-page conditions, then one of the forbidden b/w subgraphs is found in Lemma 1 or 2, without increasing the complexity. $\square$

5. Summary. This paper investigated a generalization of the VLSI-related book embedding problem, namely, the *b/w book embedding problem*. In the generalized problem, a vertex-neighborhood constraint is imposed on the ordering of the vertices. The main result presented in this paper is a characterization of the b/w graphs that admit a one-page b/w embedding. The characterization is given in terms of the *one-page conditions*, which take the form of a set of forbidden b/w subgraphs.

The sufficiency part of the result was established by means of a linear time algorithm that produces a one-page b/w embedding of a b/w graph (G, U) , provided (G, U) satisfies the one-page conditions. The algorithm can be easily extended to discover one of the forbidden subgraphs in b/w graphs not satisfying the one-page conditions. Our algorithm utilizes a technique called *b/w unfolding*, which, we believe, is a feature of independent interest. Another application of the unfolding technique is demonstrated in the companion paper [MW]. There we present an algorithm for b/w embedding of any b/w tree in a two-page book.

The main direction for further research in the area of b/w embedding seems to be that of deriving lower and upper bounds on the width of the embedding. Of particular interest are the possible trade-offs between width and number of pages in b/w embeddings.

REFERENCES

- [BK1] F. BERNHART AND P. C. KAINEN, *The book thickness of a graph*, J. Combin. Theory Ser. B, 27 (1979), pp. 320–331.

- [BK2] R. P. BRENT AND H. T. KUNG, *On the area of binary tree layout*, Inform. Processing Lett., 11 (1980), pp. 44–46.
- [CLR] F. R. K. CHUNG, F. T. LEIGHTON, AND A. L. ROSENBERG, *Embedding graphs in books: a layout problem with applications to VLSI layout*, SIAM J. Algebraic Discrete Methods, 8 (1987), pp. 33–58.
- [E] S. EVEN, *Graph Algorithms*, Computer Science Press, San Francisco, CA, 1979.
- [ET] K. P. ESWARAN AND R. E. TARJAN, *Augmentation problems*, SIAM J. Comput., 5 (1976), pp. 653–665.
- [GJMP] M. R. GAREY, D. S. JOHNSON, G. L. MILLER, AND C. H. PAPADIMITRIOU, *The complexity of coloring circular arcs and chords*, SIAM J. Algebraic Discrete Methods, 1 (1980), pp. 216–218.
- [H] L. S. HEATH, *Embedding outerplanar graphs in small books*, SIAM J. Algebraic Discrete Methods, 8 (1987), pp. 198–227.
- [MW] S. MORAN AND Y. WOLFSTAHL, *Two-page book embedding of trees under vertex-neighborhood constraints*, submitted.
- [M] S. L. MITCHEL, *Linear algorithms to recognize outerplanar graphs and maximal outerplanar graphs*, Inform. Processing Lett., 9 (1979), pp. 229–232.
- [PRS] M. S. PATERSON, W. L. RUZZO, AND L. SNYDER, *Bounds on minimax edge length for complete binary trees*, Proc. of the 13th Ann. ACM Symp. on Theory of Computer Science, 1981, pp. 293–299.
- [R] A. L. ROSENBERG, *The DIOGENES approach to testable fault-tolerant arrays of processors*, IEEE Trans. Comput., C-32 (1983), pp. 902–910.
- [RS] R. ROBERTSON AND P. D. SEYMOUR, *Graph minors—a survey*, in Surveys in Combinatorics, I. Anderson, ed., Cambridge University Press, Cambridge, 1985, pp. 153–171.
- [S1] M. M. SYSLO, *Characterizations of outerplanar graphs*, Discrete Math., 26 (1979), pp. 47–53.
- [W] A. WIGDERSON, *The complexity of the Hamiltonian circuit problem for planar graphs*, Technical Report 298, Princeton University Electrical Engineering and Computer Science Dept., Princeton, NJ, 1982.
- [Y1] M. YANNAKAKIS, *Four pages are necessary and sufficient for planar graphs*, Proc. of the 18th Ann. ACM Symp. on Theory of Computing, 1986, pp. 104–108.
- [Y2] A. C. YAO, *The entropic limitations on VLSI computations*, Proc. of the 13th Ann. ACM Symp. on Theory of Computing, 1981, pp. 308–311.

ON THE 1.1 EDGE-COLORING OF MULTIGRAPHS*

TAKAO NISHIZEKI† AND KENICHI KASHIWAGI‡

Abstract. A new upper bound is proved for the chromatic index $q^*(G)$ of multigraphs $G = (V, E)$. Let $d(G)$ be the maximum degree of G , and let

$$p(G) = \text{MAX} \{ \lceil 2 |E(X)| / (|X| - 1) \rceil : X \subset V, |X| \neq 1 \text{ and odd} \}$$

where $E(X)$ is the set of edges in the subgraph of G induced by X . The upper bound is expressed in terms of the two trivial lower bounds $d(G)$ and $p(G)$ as follows:

$$q^*(G) \leq \text{MAX} \{ p(G), \lfloor 1.1 d(G) + 0.8 \rfloor \}.$$

The proof yields an algorithm which edge-colors any given multigraph G with at most $\lfloor 1.1 q^*(G) + 0.8 \rfloor$ colors. The running time is $O(|E|(d(G) + |V|))$ and the storage space is $O(|E|)$.

Key words. alternating path, approximation algorithm, chromatic index, critical path, edge-coloring, NP-complete, performance ratio, polynomial-time, recoloring, Goldberg and Seymour's conjecture

1. Introduction. The edge-coloring problem arises in many applications, including various scheduling and partitioning problems [FW]. The problem is simply stated: color the edges of a given multigraph G using as few colors as possible so that no two adjacent edges receive the same color. This minimum number of colors is called the *chromatic index* of G and denoted by $q^*(G)$.

Let $d(G)$ denote the maximum degree of a vertex. By a well-known result due to Vizing, the chromatic index of a simple graph (that is, one without multiple edges) is at most $d(G) + 1$ [Vi]. The situation for graphs that are not simple, often called *multigraphs*, is more complicated, and it is this case that will be considered here.

Two trivial lower bounds are known for the chromatic index $q^*(G)$ of multigraphs G . One is the maximum degree $d(G)$. The other is

$$p(G) = \text{MAX} \{ \lceil 2 |E(X)| / (|X| - 1) \rceil : X \subset V, |X| \neq 1 \text{ and odd} \}$$

where $E(X)$ is the set of edges in the subgraph of G induced by X . Note that no more than $(|X| - 1)/2$ of the edges in $E(X)$ can have the same color if $|X|$ is odd. Thus

$$q^*(G) \geq \text{MAX} \{ p(G), d(G) \}.$$

Very few counter-examples to the equality above are known, and Goldberg [G1] and Seymour [Se] conjectured that the following upper bound holds:

$$q^*(G) \leq \text{MAX} \{ p(G), d(G) + 1 \}.$$

To the best of our knowledge this conjecture has neither been proved nor disproved so far. The following Goldberg's bound [Go2] has been the best known one in this direction:

$$q^*(G) \leq \text{MAX} \{ p(G), \lfloor (9 d(G) + 6) / 8 \rfloor \}.$$

This paper puts their conjecture one step forward, proving

$$q^*(G) \leq \text{MAX} \{ p(G), \lfloor 1.1 d(G) + 0.8 \rfloor \}.$$

* Received by the editors November 24, 1986; accepted for publication (in revised form) September 20, 1989.

† Department of Information Engineering, Faculty of Engineering, Tohoku University, Sendai 980, Japan.

‡ Mitsubishi Electric Corporation, The Consumer Products Development Laboratory, Ohfunata 247, Japan.

Our bound improves all the previous bounds [Sh], [Go1], [Go2], [An] including Goldberg's bound above and Shannon's: $q^* \leq \lfloor 3d(G)/2 \rfloor$.

In view of the potential applications, it would be useful to have an efficient algorithm capable of coloring any multigraph G with the minimum number $q^*(G)$ of colors. Unfortunately no such efficient algorithm is currently known. Moreover, Holyer has shown that the edge-coloring problem is NP-hard [Ho], and therefore it seems unlikely that any such polynomial-time algorithm exists [AHU]. He actually proved that deciding whether a simple graph has chromatic index of 3 or less is NP-complete. This implies that unless $P = NP$, there does not exist a polynomial-time approximation algorithm for finding an edge-coloring of a multigraph G that uses at most $(4/3 - \varepsilon)q^*(G)$ for any ε [NS].

The proof of our bound yields a polynomial-time algorithm for finding an edge-coloring of G that uses at most $(4/3)q^*(G)$ colors; moreover it uses at most $\lfloor 1.1q^*(G) + 0.8 \rfloor$ colors, that is, the number of used colors exceeds $q^*(G)$ no more than one plus 10 percent of $q^*(G)$. This result improves on the following sequence of algorithms: Nishizeki and Sato's [NS], which achieves the $(4/3)q^*$ bound as well, but uses as many as $\lfloor (5q^* + 2)/4 \rfloor$ colors; and Hochbaum, Nishizeki, and Shmoys' [HNS] and Goldberg's [Go3] which use at most $\lfloor (9q^* + 6)/8 \rfloor$ colors. In the terminology of [GJ], the "absolute performance ratio" of our algorithm is $4/3$ while the "asymptotic performance ratio" is 1.1 . The running time of the algorithm is $O(|E|(d(G) + |V|))$ and the storage space is $O(|E|)$.

2. Preliminaries. This section introduces terminology. Throughout the paper, $G = (V, E)$ denotes a given *multigraph* of vertex set V and edge set E , which may contain multiple edges but no self-loops. Since we deal with only a multigraph, we simply call G a *graph*. An edge-coloring of G with at most q colors is called a q -*coloring*. Denote by uv an edge joining vertices u and v . Assume that a graph G is edge-colored with a set of q colors. Color c in the set is a *missing color* of vertex $v \in V$ (or v *misses* c) if color c is assigned to none of the edges incident to v . Denote by $M(v)$ the set of all the missing colors of v . Denote by $C(u, v)$ the set of colors assigned to the multiple edges joining vertices u and v . An edge colored c is called a c -*edge*. For two colors a and b , the spanning subgraph of G induced by all the edges colored a or b is called an ab -*subgraph*, and is denoted by $G[a, b]$. Each connected component of $G[a, b]$ is either a path or a cycle, in which edges are colored alternately a or b . Such a path (or cycle) is called an ab -*alternating path (cycle)* or simply ab -*path (or cycle)*. We denote by $G[a, b; x]$ the connected component of $G[a, b]$ containing vertex x . A vertex $x \in V$ is an end of an ab -alternating path if and only if a or $b \in M(x)$. Interchanging the colors a and b of the edges in an ab -alternating path or cycle yields another q -coloring of G . Interchanging the colors of an alternating path or cycle is often called simply *recoloring* a path or cycle. If $a \in M(x)$ for a vertex $x \in V$, then $G[a, b; x]$ is an ab -alternating path starting from x , and denoted by $\text{Apath}(a, b; x)$. If all the edges of G , except an edge $e = xy$, are colored with q colors, and if $a \in M(x)$ and $b \in M(y)$, then the ab -alternating path between x and y , if any, is called an ab -*critical path* in particular. Note that every critical path contains an odd number of vertices. The ab -critical path is denoted by $P(a, b)$, and the number of its vertices is denoted by $|P(a, b)|$. If there is no ab -critical path, $|P(a, b)|$ is defined to be infinite. If S is a subgraph of G , then the edge set of S is denoted by $E(S)$ and the vertex set by $V(S)$ or simply S .

3. Upper bound. Let $q(G) = \text{MAX} \{p(G), \lfloor 1.1d(G) + 0.8 \rfloor\}$. The following theorem is the main result of this paper.

THEOREM 1. *Every multigraph $G = (V, E)$ satisfies $q^*(G) \leq q(G)$.*

Suppose to the contrary that $q^*(G) > q(G)$ for a graph G , and let G be such a graph with a minimum number of edges. Let $e = xy$ be an arbitrary edge of G , and let $G' = G - e$ be the graph obtained from G by deleting edge e . Then by the assumption G' can be edge colored with $q(G)$ colors. We will show that a q -coloring of G' can be extended to a q -coloring of G , that is, a contradiction occurs. The remainder of this section is devoted to the preliminary of the proof.

The following facts can be easily derived.

Fact 1. $d(G) \geq 3$.

Proof. If $d(G) \leq 2$, then clearly $q^*(G) \leq q(G)$. $\square$

The following fact is an immediate consequence of Fact 1.

Fact 2. $q(G) \geq \lfloor 1.1 d(G) + 0.8 \rfloor \geq d(G) + 1$.

Fact 3. In a q -coloring of G' every vertex has at least one missing color; in particular, the ends x and y of the uncolored edge have at least two.

Fact 4. The ends x and y of the uncolored edge e have no common missing color.

Proof. If x and y have a common missing color c , then coloring xy c yields a q -coloring of whole G . $\square$

Fact 5. G' has an ab -critical path for any $a \in M(x)$ and $b \in M(y)$.

Proof. Otherwise, interchange the colors a and b of $A_{\text{path}}(a, b; x)$. Then x and y have a common missing color b , contradicting Fact 4. $\square$

The following lemma has been proved in [Go2, Lemma 4].

LEMMA 1. *No two vertices on any critical path have a common missing color. (In order to make the paper self-contained, we will give a proof in Appendix 1.)*

We obtain the following lemma from the definition of $q(G)$.

LEMMA 2. *Let S be a set of vertices no two of which have a common missing color. Then we have*

(a) *If $x, y \in S$ then $|S| \leq 10$; and*

(b) *If $x \in S$ then $|S| \leq 11$.*

Proof. (a) Clearly $|M(x)|, |M(y)| \geq q - d + 1$, and $|M(v)| \geq q - d$ for every vertex v other than x and y . Since $\sum_{v \in S} |M(v)| \leq q$, we have

$$2(q - d + 1) + (|S| - 2)(q - d) \leq q,$$

which immediately yields

$$|S| \leq (q - 2)/(q - d).$$

Noting that $d \geq 3$ and $q \geq (1.1d + 0.8) - 0.9$, we easily have $|S| \leq 10$.

(b) Similar to (a). $\square$

LEMMA 3. *Let $P(a, b)$ be the longest critical path among all the q -coloring of G' . Then $P(a, b)$ has at most nine vertices.*

Proof. By Lemma 1 no two vertices on $P(a, b)$ have a common missing color. Therefore $|P(a, b)| \leq 10$ by Lemma 2(a). Since clearly $|P(a, b)|$ is odd, we have $|P(a, b)| \leq 9$. $\square$

Thus we shall consider the case in which the longest critical path has 9, 7, 5, or 3 vertices. Before presenting the detail of the proof, we give two lemmas. One of them, Lemma 4, has been proved in [Go2, Lemma 5].

LEMMA 4. *Any two critical paths $P(a, b)$ and $P(f, g)$ contain no two vertices u and v of a common missing color. (A proof will appear in Appendix 2.)*

LEMMA 5. *All the critical paths together contain at most 10 vertices.*

Proof. The proof follows immediately from Lemmas 1, 2, and 4. $\square$

LEMMA 6. *Let $P(a, b)$ be an ab -critical path in a q -coloring of G' , and let $c \in M(x)$. Recolor an ac - or bc -alternating path or cycle containing neither x nor y if it exists. Let*

$P'(a, b)$ be the new ab -critical path in the resulting q -coloring of G' . Then no two vertices in $V(P(a, b)) \cup V(P'(a, b))$ have a common missing color, and

$$|V(P(a, b)) \cup V(P'(a, b))| \leq 10.$$

Proof. We may assume that $P'(a, b)$ is obtained by recoloring an ac -path. Consider the original q -coloring of G' . By Lemma 1 no two vertices on $P(a, b)$ have a common missing color. We now prove that no two vertices on $P'(a, b)$ have a common missing color. Suppose that two vertices u, u' on $P'(a, b)$ have a common missing color f . If $f \neq a, c$, then u and u' on $P'(a, b)$ would have a common missing color f after recoloring the ac -path, contrary to Lemma 1. If $f = c$ and u or u' is on the ac -path, then, after recoloring the ac -path, u or u' would miss color a and there would not exist an ab -critical path $P'(a, b)$, a contradiction. Therefore, if $f = c$, then neither u nor u' is on the ac -path, and consequently u and u' would have a common missing color c after recoloring the ac -path, contrary to Lemma 1. Similarly a contradiction occurs if $f = a$.

We shall next prove that no two vertices $v \in V(P(a, b))$ and $v' \in V(P'(a, b))$ have a common missing color in the original q -coloring of G' . Suppose that v and v' miss a color $f \neq a, b, c$. We may assume that $v \in V(P(a, b) - P'(a, b))$, and $v' \in V(P'(a, b) - P(a, b))$.

Let g be a missing color of y other than b . Clearly $g \neq a, b, c, f$, and $\text{Apath}(g, f; y)$ must end at either v or v' . Recolor $\text{Apath}(g, f; y)$, then $f \in M(y)$ and either $f \in M(v)$ or $f \in M(v')$. In the first case, $P(a, b)$ contains two vertices v and y of a common missing color f , contradicting Lemma 1. In the latter case, $P'(a, b)$ obtained by recoloring the ac -path contains two vertices v' and y of a common missing color f , contrary to Lemma 1.

Thus we have proved that no two vertices in $V(P(a, b)) \cup V(P'(a, b))$ have a common missing color. Lemma 2(a) immediately implies $|V(P(a, b)) \cup V(P'(a, b))| \leq 10$. $\square$

An edge of G is said to *leave a subgraph H at vertex z* if it joins a vertex z in H and a vertex not in H . From now on we assume that $P(a, b)$ is the *longest* critical path between x and y and that H is the subgraph of G induced by the (odd number of) vertices of $P(a, b)$.

The definition of $q(G)$ implies $q \geq \lceil 2|E(H)|/(|V(H)| - 1) \rceil$. Clearly each of the q colors is assigned to at most $(|V(H)| - 1)/2$ edges of H . However H has an uncolored edge xy . Therefore there must be a color c assigned to at most $(|V(H)| - 3)/2$ edges of H . Thus H contains three or more vertices to which no c -edge of H is incident. Since at most one of them misses color c , G' contains two or more c -edges leaving H . (In particular there exist three or more c -edges leaving H if no vertex in H misses c .)

The following lemma is implicitly mentioned in [Go2].

LEMMA 7. *We may assume that an edge colored with a missing color of x or y leaves H .*

Proof. By the discussion above there are two or more edges leaving H . We first show that G has an edge leaving H and colored with a missing color of a vertex of H , not necessarily x or y . Suppose otherwise. Then G has three or more c -edges leaving H and no vertex in H misses c . Let uu' be one of these c -edges leaving H where $u \in H$ and $u' \notin H$. Let f be any missing color of u , where $f \neq a, b$. Let vv' be the final c -edge of $\text{Apath}(f, c; u)$ leaving H where $v \in H$ and $v' \notin H$. If $u = v$ then interchange the colors of $\text{Apath}(f, c; u)$. Now $c \in M(u)$, and there is a c -edge leaving H . Thus $u \neq v$. Let g be any missing color of v , where $g \neq a, b, c, f$. Since no f - or g -edge leaves H , $\text{Apath}(g, f; v)$ does not leave H and must end at u . Recolor $\text{Apath}(g, f; v)$, now $f \in M(v)$. Recolor $\text{Apath}(f, c; v)$. Now $c \in M(v)$, and there is a c -edge leaving H . Thus G' has an edge leaving H and colored with a missing color of a vertex of H .

Define newly colored c and vertices $u, v \in H$ so that $c \in M(v)$ and a c -edge uu' leaves H . One may assume, without loss of generality, that $v \neq x, y$ and that vertices x, v, u and y appear on $P(a, b)$ in this order; otherwise interchange the roles of x and y . Let w be the preceding vertex of v on $P(a, b)$. Erase the color of edge wv of $P(a, b)$. Recolor the subpath of $P(a, b)$ between x and w . Color xy b . In the resulting coloring, wv is the uncolored edge, $c \in M(v)$ and there exists a c -edge uu' leaving H . $\square$

LEMMA 8. *Let $c(\neq a, b)$ be an arbitrary color, and let uu' be a c -edge leaving H , where $u \in V(H)$ and $u' \notin V(H)$. Unless $G[a, c; u]$ (or $G[b, c; u]$) is a path terminating in x or y , $G[a, c; u]$ (or $G[b, c; u]$) have two or more a - (or b -) edges on $P(a, b)$.*

Proof. Suppose that $G[a, c; u]$ is not a path terminating in x or y but contains exactly one a -edge of $P(a, b)$. Recolor $G[a, c; u]$, then $a \in M(x)$, $b \in M(y)$ and either there could be no ab -critical path or a new ab -critical path would be longer than $P(a, b)$. $\square$

LEMMA 9. *The longest critical path $P(a, b)$ contains at least five vertices.*

Proof. Suppose that $P(a, b)$ contains exactly three vertices. Then by Lemma 7 we may assume that $c \in M(x)$ and two c -edges leave H at y and the intermediate vertex of $P(a, b)$. Then $G[a, c; y]$ is not a path terminating in x or y , but contains exactly one a -edge on $P(a, b)$, contradicting Lemma 8. $\square$

By Lemmas 3 and 9 we shall consider three cases, in which the longest critical path $P(a, b)$ has 9, 7, or 5 vertices, respectively. The proof of the cases will be given in the succeeding three sections. Our goal is to show that there are no two edges leaving the subgraph induced by $V(P(a, b))$ and colored the same color. For each case, we obtain a contradiction by showing that either the graph contains a longer critical path, or it contains more than 10 vertices without a common missing color.

4. Case 1. The longest critical path $P(a, b)$ contains nine vertices. Let $P(a, b) = xz_1z_2 \cdots z_7y$. By Lemma 7 we may assume that $c \in M(x)$ and two or more c -edges leave H . No vertex in $P(a, b)$ except x misses color c . By Fact 5 there must be a cb -critical path $P(c, b)$. Path $P(c, b)$ does not leave H , that is, $V(P(c, b)) \subset V(P(a, b))$, for otherwise there would exist at least 11 vertices in $P(a, b) \cup P(c, b)$, contradicting Lemma 5. By Lemma 3 $P(c, b)$ contains at most 9 vertices. If $|P(c, b)| = 9$, then there could not exist a c -edge leaving H . If $|P(c, b)| = 7$, then the component of $G[b, c]$ which contains the c -edges leaving H would have exactly one b -edge on $P(a, b)$, contradicting Lemma 8. Thus $|P(c, b)| = 5$ or 3. We shall consider the following two cases.

Case 1.1. $|P(c, b)| = 3$. In this case path $P(c, b)$ is xz_1y . We can easily know that no c -edge leaves H at z_2 ; otherwise, the new ab -critical path $P'(a, b)$ produced by recoloring $G[b, c; z_2]$ satisfies $|V(P(a, b) \cup P'(a, b))| \geq 11$, contradicting Lemma 6. Similarly no c -edge leaves H at z_7 . Therefore two or more c -edges must leave H only at z_3, z_4, z_5 , or z_6 . Consider the following two cases.

Case 1.1.1. $c \in C(z_2, z_7)$. Since $z_1z_2z_7yz_1$ is an ac -alternating cycle, neither $G[a, c; z_3]$ nor $G[a, c; z_6]$ is a path terminating in x or y . Therefore by Lemma 8 $c \notin C(z_3, z_4)$, $c \notin C(z_5, z_6)$, and $G[a, c; z_3]$ contains two or more a -edges on $P(a, b)$. Therefore $G[a, c; z_3]$ must contain the two a -edges z_3z_4 and z_5z_6 together with all the c -edges leaving H . Consequently $G[a, c; z_3] = G[a, c; z_6]$.

We first show that a c -edge $z_3z'_3$ leaves H . Otherwise, a c -edge joins z_3 with either z_5 or z_6 since $c \notin M(z_3)$. Suppose first that $c \in C(z_3, z_5)$, then recoloring $G[a, c; z_3]$ would produce a new ab -critical path having 11 or more vertices, contradicting Lemma 3. Suppose next that $c \in C(z_3, z_6)$, then $G[b, c; z_4]$ would contain exactly one b -edge on $P(a, b)$, contrary to Lemma 8.

By symmetry a c -edge $z_6z'_6$ leaves H at z_6 . (Note that erasing color b from xz_1 and assigning color b to the uncolored edge xy produces a q -coloring of G' , in which the direction of $P(a, b)$ is reversed.)

Let $P'(a, b)$ be the new ab -critical path produced by recoloring $G[a, c; z_3]$. Then $P'(a, b)$ contains $x, z_1, z_2, z_3, z'_3, \dots, z'_6, z_6, z_7$ and y , and consequently $|V(P(a, b) \cup P'(a, b))| \geq 11$, contradicting Lemma 6.

Case 1.1.2. $c \notin C(z_2, z_7)$. Since two or more c -edges leave H only at z_3, z_4, z_5 or z_6 , by symmetry we may assume that a c -edge leaves H at either z_3 or z_4 .

Suppose that a c -edge $z_3z'_3$ leaves H at z_3 . Then a c -edge joins z_2 with either z_4, z_5 or z_6 . If $c \in C(z_2, z_4)$, then the new ab -critical path $P'(a, b)$ produced by recoloring $G[b, c; z_3]$ contains z'_3 , and consequently $|V(P(a, b) \cup P'(a, b))| \geq 11$, contradicting Lemma 6. If $c \in C(z_2, z_5)$, then either $c \in C(z_7, z_6)$ and a c -edge leaves H at z_4 or $c \in C(z_7, z_4)$ and a c -edge leaves H at z_6 . In the former case, $G[a, c; z_3]$ contains exactly one a -edge z_3z_4 , contradicting Lemma 8. In the latter case, the new ab -critical path $P'(a, b)$ produced by recoloring $G[b, c; z_2]$ satisfies $|P'(a, b)| \geq 11$, contrary to Lemma 3. Finally, if $c \in C(z_2, z_6)$, then either $c \in C(z_4, z_7)$ or $c \in C(z_5, z_7)$. In the former case, $|P'(a, b)| \geq 11$, contrary to Lemma 3. In the latter case, $G[a, c; z_3]$ would contain exactly one a -edge on $P(a, b)$, contrary to Lemma 8.

Thus we have shown that no c -edge leaves H at z_3 . By symmetry no c -edge leaves H at z_6 . We have already shown that no c -edge leaves H at z_2 or z_7 . Therefore two c -edges must leave H at z_4 and z_5 , and either $z_2z_3z_7z_6z_2$ is a bc -cycle or $z_2z_3z_2$ and $z_6z_7z_6$ are bc -cycles. In either case $G[b, c; z_4]$ contains exactly one b -edge on $P(a, b)$, contrary to Lemma 8.

Case 1.2. $|P(c, b)| = 5$. Exactly two of the four b -edges on $P(a, b)$ are contained in $P(c, b)$. One of them is xz_1 . The other is either z_2z_3, z_4z_5 or z_6z_7 . Thus we shall consider the following three cases.

Case 1.2.1. $P(c, b)$ contains z_2z_3 . In this case $P(c, b)$ is either $xz_1z_2z_3y$ or $xz_1z_3z_2y$. In either case the leaving c -edges are incident to z_4, z_5, z_6 or z_7 . Similarly as in Case 1.1.1 we can easily show that there exist two c -edges $z_4z'_4$ and $z_7z'_7$ leaving H , and that $G[b, c; z_4]$ contains z_4, z'_4, z_5, z_6, z_7 , and z'_7 . Let $P'(a, b)$ be the new ab -critical path produced by recoloring $G[b, c; z_4]$. Then $|P(a, b) \cup P'(a, b)| \geq 11$, contradicting Lemma 6.

Case 1.2.2. $P(c, b)$ contains z_6z_7 . A contradiction can be derived similarly as in Case 1.1.1.

Case 1.2.3. $P(c, b)$ contains z_4z_5 . In this case either $P(c, b) = xz_1z_4z_5y$ or $P(b, c) = xz_1z_5z_4y$.

Case 1.2.3.1. $P(c, b) = xz_1z_4z_5y$. We can easily know that no c -edge leaves H at z_2 ; otherwise, recoloring $G[b, c; z_3]$ would produce a new ab -critical path $P'(a, b)$ satisfying $|V(P(a, b) \cup P'(a, b))| \geq 11$. Thus either $c \in C(z_2, z_3), c \in C(z_2, z_6)$ or $c \in C(z_2, z_7)$. However if $c \in C(z_2, z_3)$ then $G[b, c; z_6]$ would contain exactly one b -edge on $P(a, b)$, contradicting Lemma 8. If $c \in C(z_2, z_6)$ then recoloring $G[b, c; z_2]$ would produce an ab -critical path having more than 9 vertices, contradicting Lemma 3. Thus we have shown that $c \in C(z_2, z_7)$. Then two c -edges leave H at z_3 and z_6 .

Recolor $G[b, c; z_3]$, and subsequently recolor the ac -cycle $G[a, c; y]$. Then the new ab -critical path, if any, has 11 or more vertices, a contradiction.

Case 1.2.3.2. $P(c, b) = xz_1z_5z_4y$. As in Case 1.2.3.1 above we can show that $c \in C(z_2, z_7)$. Then there are two c -edges $z_3z'_3$ and $z_6z'_6$ leaving H , where $z'_3, z'_6 \notin V(H)$. (See Fig. 1.) Let f be a missing color of y other than b . There must exist af - and cf -critical paths. By Lemma 5 $P(a, f)$ must contain only the vertices on $P(a, b)$, while $P(c, f)$ contains at most one vertex not on $P(a, b)$. By Lemma 8 every component of $G[a, f]$ (or $G[b, f]$) containing a leaving f -edge must contain two or more a -edges (respectively, b -edges) in H . Since $P(a, f)$ cannot leave H , an f -edge joins x with one of $z_1, z_2, \dots, z_7$. It is rather easy to verify that $f \notin C(x, z_4) \cup C(x, z_7)$, as follows.

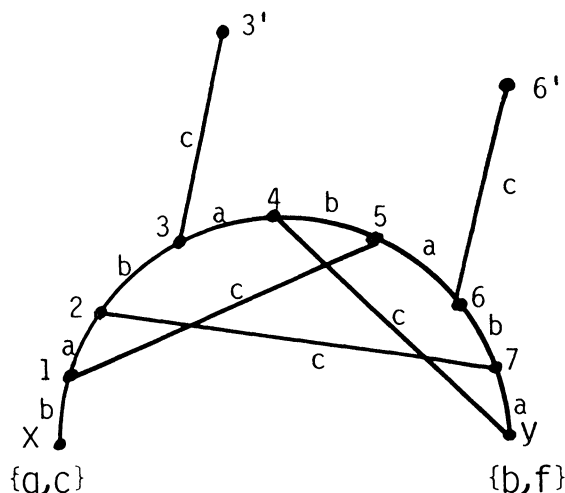


FIG. 1. Illustration for Case 1.2.3.2.

First suppose that $f \in C(x, z_4)$. Then $P(c, f) = xz_4z_7$, and so $P(c, f)$ and $G[c, f; z_3]$ are disjoint. Recolor $G[c, f; z_3]$, then $P(a, f) = xz_4z_3z'_3 \cdots z_7y$ leaves H while $P(a, b)$ remains unchanged. Therefore $|P(a, f) \cup P(a, b)| \geq 11$, contradicting Lemma 5.

Next suppose that $f \in C(x, z_7)$. Then $P(a, f) = xz_7y$. Recolor all the components of $G[a, f]$ that contain a -edges z_1z_2 , z_3z_4 and z_5z_6 . Now

$$P(c, f) = xz_7z_2z_1z_5z_6z'_6 \cdots z'_3z_3z_4y,$$

so $|P(c, f)| \geq 11$, contradicting Lemma 3.

Thus we have shown that an f -edge joins x with either z_1 , z_2 , z_3 , z_5 or z_6 . In either case, an appropriate recoloring would produce 11 or more vertices on critical paths, contrary to Lemma 5. The easy but lengthy case study will be given in Appendix 3 for the patient reader.

5. Case 2. The longest critical path $P(a, b)$ contains seven vertices. Before presenting a proof for Case 2, we give two lemmas, which generalize Lemma 7 in some sense.

LEMMA 10. Assume that the following conditions hold:

(a) $\{a_1, a_2, \dots, a_k\} \subset M(x)$;

(b) $b \in M(y)$;

(c) there is a subgraph S induced by nine vertices including x and y such that no two vertices in S have a common missing color;

(d) none of the edges leaving S is colored $a_1, \dots, a_k$ or b ; and

(e) an edge leaving S is colored with a missing color f of a vertex $w \neq y$ in S .

Then, applying a sequence of recolorings, we can alter the q -coloring of G' , without changing the subgraph of G induced by the edges colored $a_1, \dots, a_k$ and b , so that an edge leaving S is colored with a missing color of y .

Proof. By condition (d) $f \neq a_1, \dots, a_k, b$. Since no two vertices in S have a common missing color, two or more f -edges leave S . Therefore eleven or more vertices are in S or ends of the leaving f -edges. By Lemma 2(a) two of them, say u and v , have a common missing color g . We may assume that one of them, say u , is an end of a leaving f -edge, not in S . Therefore $g \neq f$. Let h be a missing color of y other than b ,

where $h \neq a_1, \dots, a_k, b$. We may assume that $h \neq f$ and no h -edge leaves S ; otherwise the proof is completed. Note that possibly $h = g$.

We may assume that the common missing color g of two vertices is different from $a_1, \dots, a_k, b$. Suppose otherwise that $g = a_1, \dots, a_k$, or b . Since there is no leaving edge colored with $a_1, \dots, a_k, b$ or h , $\text{Apath}(g, h; u)$ is disjoint with S . Recolor $\text{Apath}(g, h; u)$, then the common missing color h of two vertices u and y is different from $a_1, \dots, a_k, b$, and the subgraph of G induced by the edges colored $a_1, \dots, a_k$ and b remains unchanged.

Recolor $\text{Apath}(h, g; y)$ if $g \notin M(y)$. Then $g \in M(y)$. If $v \in V(S) - \{y\}$, then $\text{Apath}(h, g; y)$ had to end at v (otherwise, two vertices v and y in S would have a common missing color g , contrary to condition (c)), and consequently color g remains missing at u . On the other hand, if $v \notin S$, then v is also an end of a leaving f -edge and either $g \in M(u)$ or $g \in M(v)$ after recoloring $\text{Apath}(h, g; y)$. Thus no matter whether or not $v \in S$, there remains a leaving f -edge having an end which is not in S and misses color $g \in M(y)$. Now $\text{Apath}(f, g; w)$ must end at y ; otherwise, recoloring it would make two vertices w and y in S have a common missing color g , contrary to condition (c). The leaving f -edge above is not contained in $\text{Apath}(f, g; w)$ since its end not in S misses g . Recolor $\text{Apath}(f, g; w)$, then $f \in M(y)$ and the f -edge remains to leave S . $\square$

LEMMA 11. Assume that the following conditions hold:

(a) $\{a_1, a_2, \dots, a_k\} \subset M(x)$;

(b) $b \in M(y)$;

(c') there is a subgraph S induced by vertices including x and y such that no two vertices in S have a common missing color;

(d) none of the edges leaving S is colored $a_1, \dots, a_k$ or b ; and

(e') none of the edges leaving S is colored with a missing color of a vertex in S , but three or more of them are colored with the same color, say f .

Let uu' be an f -edge leaving S such that $u \in S - \{x\}$ and $u' \notin S$. Let g be a missing color of u such that $g \neq a_1, \dots, a_k, b, f$. Let vv' be the last f -edge leaving S and contained in $\text{Apath}(g, f; u)$, where $v \in S$ and $v' \in S$. Then

(i) $v = x$;

(ii) every $\text{Apath}(a_i, f; x)$ contains no vertex in S except x ;

(iii) if $|S| = 9$ and $P'(f, b)$ is the new fb -critical path obtained by recoloring $\text{Apath}(a_i, f; x)$, then $P'(f, b)$ contains no f -edge leaving S .

Proof. By condition (e') no g -edge leaves S , while by condition (c') no vertex in S except u misses color g . If $v = u$, then recoloring $\text{Apath}(g, f; u)$ would make $f \in M(u)$ and an f -edge remain to leave S , contrary to condition (e'). Thus $v \neq u$.

(i) Suppose that $v \neq x$. Then v has a missing color $h (\neq a_1, \dots, a_k, b, f, g)$. (If $v = x$, then v may have no missing color other than $a_1, \dots, a_k$.) By condition (c') no vertex in S except v misses color h , and by condition (e') no h -edge leaves S . Recolor $\text{Apath}(h, g; v)$, then $g \in M(v)$. Furthermore recolor $\text{Apath}(g, f; v)$, then $f \in M(v)$ and an f -edge remains to leave S , contrary to condition (e'). Thus we have shown that $v = x$.

(ii) Suppose that for some $i, i = 1, 2, \dots, k$, $\text{Apath}(a_i, f; x)$ contains a vertex in S other than x . We may assume, without loss of generality, that $i = 1$. Let ww' be the last f -edge leaving S in $\text{Apath}(a_1, f; x)$, where $w \in S - \{x\}$ and $w' \notin S$. Let $h \in M(w)$, where $h \neq a_1, \dots, a_k, f$. There is no a_1 - or h -edge leaving S . Interchange the colors a_1 and h only in the subgraph S , then $a_1 \in M(w)$ and $\{h, a_2, \dots, a_k\} \subset M(x)$, and hence all the conditions (a), (b), (c'), (d), and (e') hold if the roles of a_1 and h are interchanged. Recolor $\text{Apath}(a_1, f; w)$, then $f \in M(w)$ and an f -edge remains to leave S , contradicting condition (e').

(iii) Suppose that $P'(f, b)$ contains a leaving f -edge. Then 11 or more vertices are contained in $S \cup P'(f, b)$ since $|S| = 9$ in (iii). By Lemma 2(a) two vertices t and t' in $S \cup P'(f, b)$ have a common missing color h . We may assume that $t \in S - P'(f, b)$ and $t' \in P'(f, b) - S$ is an end of a leaving f -edge. Let j be a missing color of y other than b . If $\text{Apath}(j, h; y)$ ended at t' , then recoloring $\text{Apath}(j, h; y)$ would make the two vertices t and y in S have a common missing color h , a contradiction. Thus $\text{Apath}(j, h; y)$ must end at t . In this case recolor $\text{Apath}(a_i, f; x)$ and $\text{Apath}(j, h; y)$, then two vertices y and t' on the new fb -critical path $P'(f, b)$ would have a common missing color h , a contradiction. $\square$

We are now ready to present a proof for Case 2. Let $P(a, b) = xz_1z_2z_3z_4z_5y$. By Lemma 7 we may assume that $c \in M(x)$ and two or more c -edges leave the subgraph H induced by $V(P(a, b))$. None of vertices $z_1, \dots, z_5$ and y misses color c , and by Fact 5 there must exist a cb -critical path $P(c, b)$ such that $|P(c, b)| \leq 7$. By Lemma 5 $|P(a, b) \cup P(c, b)| \leq 9$. Moreover by Lemma 8 a component of $G[b, c]$ (or $G[a, c]$) containing a leaving c -edge must include two or more b -edges (or a -edges) on $P(a, b)$ if it is not a path terminating in x or y . There are three cases to consider.

Case 2.1. $P(c, b)$ contains 7 vertices. Clearly $P(c, b)$ contains the b -edge xz_1 . Since there is a c -edge leaving H , $V(P(a, b)) \neq V(P(c, b))$. Therefore $P(c, b)$ contains a leaving c -edge. Since $|P(a, b) \cup P(c, b)| \leq 9$, $P(c, b)$ contains exactly two leaving c -edges and contains exactly one of the two b -edges z_2z_3 and z_4z_5 . Denote by w_2 and w_3 the ends of the two leaving c -edges not on $P(a, b)$. Then $b \in C(w_2, w_3)$, and $a \notin M(w_2) \cup M(w_3)$. If $a \in C(w_2, w_3)$, then recoloring $G[a, b; w_2]$ would make $|P(c, b)| \geq 9$, contradicting the assumption that the longest critical path contains 7 vertices. Thus $a \in C(w_2, w_3)$. Furthermore we can verify the following lemma.

LEMMA 12. *We may assume that $P(c, b)$ takes one of the following three forms (see Fig. 2):*

- (a) $xz_1w_2w_3z_4z_5y$;
- (b) $xz_1w_2w_3z_5z_4y$;
- (c) $xz_1w_2w_3z_3z_2y$.

Furthermore the two vertices that are on $P(a, b)$ but not on $P(c, b)$ are joined by a c -edge.

Proof. If $P(c, b)$ is none of (a)–(c), then $P(c, b)$ must be

- (d) $xz_1w_2w_3z_2z_3y$;
- (e) $xz_1z_2z_3w_2w_3y$;
- (f) $xz_1z_3z_2w_2w_3y$;
- (g) $xz_1z_5z_4w_2w_3y$; or
- (h) $xz_1z_4z_5w_2w_3y$.

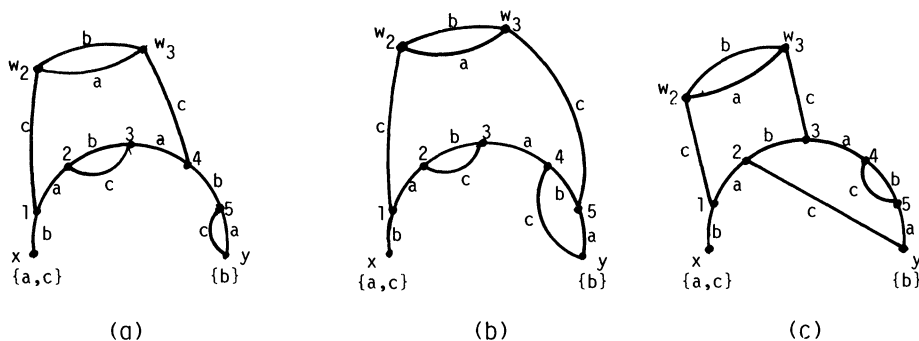


FIG. 2. Illustration for Lemma 10.

In either of cases (a)–(h) the two vertices on $P(a, b)$, not on $P(c, b)$, must be joined by a c -edge because, otherwise, the component of $G[b, c]$ containing these vertices would contain exactly one b -edge on $P(a, b)$ and contain c -edges leaving H , contrary to Lemma 8. In case (d), since $yz_3z_4z_5y$ is an ac -cycle, $G[a, c; z_1]$ contains exactly one a -edge on $P(a, b)$, contradicting Lemma 8. Thus case (d) is impossible. Similarly case (h) is also impossible. In cases (e)–(g), erase the color b from xz_1 , color the uncolored edge xy with b , and interchange the roles of y and z_1 . Then cases (e)–(g) can be reduced to cases (a)–(c), respectively. $\square$

Let S be the subgraph of G induced by the nine vertices in $P(a, b) \cup \{w_2, w_3\}$. By Lemma 12 no c -edge leaves S . Since $q \geq \lceil 2|E(S)|/(|V(S)| - 1) \rceil$, there must exist an edge leaving S . Assume that the edge is colored f , then $f \neq a, b, c$. Next we have the following lemma.

LEMMA 13. $f \in M(y)$.

Proof. If there is a vertex w in S with $f \in M(w)$, then S satisfies the conditions (a)–(e) of Lemma 10 with $a_1 = a$ and $a_2 = c$, and consequently we can make $f \in M(y)$. Thus we may assume that no vertex in S misses color f . Define u, u', v, v' , and g as in Lemma 11, then $v = x$ and neither $\text{Apath}(a, f; x)$ nor $\text{Apath}(c, f; x)$ contains any vertex in S except x . Furthermore $P'(f, b)$ contains no f -edge leaving S .

We then show that $|P'(f, b)| = 3$, that is, $f \in C(z_1, y)$. Clearly $|P'(f, b)| \leq 7$. Suppose $|P'(f, b)| = 5$ or 7 . Then $P'(f, b)$ contains one or two of the three b -edges z_2z_3, z_4z_5 and w_2w_3 . Therefore there exists a component of $G[f, b]$ which contains leaving f -edges and exactly one b -edge on $P(a, b)$ or $P(c, b)$, contradicting Lemma 8.

By Lemma 12 we shall consider the following three cases; note that $f \in C(z_1, y)$ although the f -edge is not drawn in Fig. 2.

Case (a). S takes the form shown in Fig. 2(a). We first claim that no f -edge leaves S at z_2, w_2 or z_5 . Suppose that to the contrary there is an f -edge leaving S at z_2, w_2 , or z_5 . Let $P'(a, b)$ be the new ab -critical path obtained by successively recoloring $G[c, f; x]$ and all the components of $G[b, f]$ containing leaving f -edges. Let $P'(c, b)$ be the new cb -critical path obtained by successively recoloring $G[a, f; x]$ and all the components of $G[b, f]$ containing leaving f -edges. Then either $|S \cup P'(a, b)| \geq 11$ or $|S \cup P'(c, b)| \geq 11$. Therefore, considering a missing color of y other than b , we can derive a contradiction similarly as in the proof of Lemma 11 (iii).

The claim above implies that exactly two f -edges leave S at z_3, z_4 , and w_3 . We can easily know that $f \notin C(w_2, w_3) \cup C(z_2, z_3) \cup C(z_4, z_5)$; for otherwise either $|P'(a, b)|$ or $|P'(c, b)|$ would exceed 7.

Suppose that the f -edges leave S either at z_3 and z_4 or at w_3 and z_4 . We may assume the former. Then either $f \in C(w_2, z_5) \cap C(z_2, w_3)$ or $f \in C(w_2, z_2) \cap C(z_5, w_3)$. In either case $G[a, f; z_3]$ contains exactly one a -edge on $P(a, b)$, contradicting Lemma 8.

Thus the f -edges leave S at z_3 and w_3 . Since $f \notin C(z_4, z_5), f \notin C(w_2, z_2)$. Therefore either $f \in C(w_2, z_5) \cap C(z_2, z_4)$ or $f \in C(w_2, z_4) \cap C(z_2, z_5)$. Then we would have $|P'(a, b)| \geq 9$ or $|P'(c, b)| \geq 9$, respectively.

Case (b). S takes the form shown in Fig. 2(b). As in case (a) above, we can easily show that the two f -edges leave S at w_3 and z_3 . Furthermore $f \notin C(w_2, z_2)$, and either $f \in C(w_2, z_4) \cap C(z_2, z_5)$ or $f \in C(w_2, z_5) \cap C(z_2, z_4)$. In the former case $G[f, c; w_3]$ contains exactly one c -edge on $P(c, b)$, contrary to Lemma 8. In the latter case we would have $|P'(c, b)| \geq 9$.

Case (c). S takes the form shown in Fig. 2(c). Similarly as in Case (a) above, we can show that exactly two f -edges leave S at w_3 and $z_4, f \notin C(w_2, z_2)$, and $f \in C(w_2, z_3) \cap C(z_2, z_5)$. However in this case we would have $|P'(c, b)| \geq 9$. $\square$

Next we have the following lemma.

LEMMA 14. If $f \in M(y)$, then $|P(a, f)| = |P(c, f)| = 3$.

Proof. Since $|P(a, b) \cup P(c, b)| = 9$, both $P(a, f)$ and $P(c, f)$ contain no f -edges leaving S . In Case 2 we are assuming that the longest critical path contains 7 vertices. Therefore $|P(a, f)|, |P(c, f)| \leq 7$. Suppose $|P(a, f)| = 7$. Then $P(a, f)$ contains three a -edges including z_5y . Since $P(a, f)$ does not leave S , exactly two of the three a -edges z_1z_2 , z_3z_4 and w_2w_3 are contained in $P(a, f)$, and consequently two f -edges leave S at the ends of the remaining a -edge. If that a -edge is either z_1z_2 or z_3z_4 , then the component of $G[a, f]$ containing the two f -edges leaving H (and S) includes exactly one a -edge on $P(a, b)$, contradicting Lemma 8. If that a -edge is w_2w_3 , then the successive recolorings of $G[a, f; w_2]$ and $G[a, b; w_2]$ would make $|P(c, b)| \geq 9$. Thus we have shown that $|P(a, f)| \leq 5$. Similarly as above we have $|P(c, f)| \leq 5$.

Suppose next that $|P(a, f)| = 5$. Then $P(a, f)$ contains the a -edge z_5y and exactly one of the three a -edges z_1z_2 , z_3z_4 and w_2w_3 . Suppose first that either z_1z_2 or z_3z_4 , say z_3z_4 , is contained in $P(a, f)$. Then $f \in C(z_1, z_2)$; otherwise $G[a, f; z_1]$ would violate Lemma 8. Thus two f -edges must leave S at w_2 and w_3 . The successive recolorings of $G[a, f; w_2]$ and $G[a, b; w_2]$ would make $|P(b, c)| \geq 9$, a contradiction. Suppose next that w_2w_3 is contained in $P(a, f)$. Then $P(a, f)$ is either $xw_2w_3z_5y$ or $xw_3w_2z_5y$. By Lemma 12 there are three cases to consider.

Case (a). S takes the form in Fig. 2(a). In this case $|P(c, f)| \geq 7$, a contradiction. Thus we have $|P(a, f)| = 3$. Since in this case the coloring of S is symmetric with respect to colors a and c , we also have $|P(c, f)| = 3$.

Case (b): S takes the form in Fig. 2(b). Since $|P(c, f)| \leq 5$, $P(a, f)$ must be $xw_2w_3z_5y$. Then $P(c, f) = xw_2z_1z_4y$, and two f -edges leave S at z_2 and z_3 . The component $G[b, f; z_3]$ violates Lemma 8 since $xz_1z_4z_5w_3w_2x$ is a bf -cycle. Thus we have $|P(a, f)| = 3$. By symmetry we also have $|P(c, f)| = 3$. (Consequently this case is impossible.)

Case (c). S takes the form in Fig. 2(c). If $P(a, f) = xw_2w_3z_5y$, then $P(c, f) = xw_2z_1z_2y$ and two f -edges leave S at z_3 and z_4 , and hence $G[a, f; z_3]$ violates Lemma 8. If $P(a, f) = xw_3w_2z_5y$, then $P(c, f) = xw_3z_3z_2y$ and two f -edges leave S at z_1 and z_4 , and hence recoloring $G[a, f; z_3]$ would make $|P(c, f)| \geq 9$. Thus we have shown that $|P(a, f)| = 3$. Therefore $f \in C(x, z_5)$, and $P(c, f) = xz_5z_4z_2y$. Clearly $f \notin C(z_1, w_2) \cup C(z_3, w_3) \cup C(z_1, z_3)$. Moreover no two f -edges leave S at z_1 and z_3 ; otherwise, recoloring $G[a, f; z_1]$ would make $|P(a, b)| \geq 9$. Therefore $f \notin C(w_2, w_3)$. Thus either $f \in C(z_1, w_3)$ or $f \in C(z_3, w_2)$. In either case recoloring $G[f, c; w_2]$ would make $|P(c, b)| \geq 9$. Hence this case is also impossible. $\square$

By Lemmas 13 and 14 we may assume that $f \in M(y)$, $P(c, b) = xz_1w_2w_3z_4z_5y$, and an f -edge must join x and z_5 , as shown in Fig. 3. If $f \in C(z_1, z_4)$, then $xz_1z_4z_5x$ is a bf -cycle and consequently either $G[b, f; z_2]$ or $G[b, f; w_2]$ contains exactly one b -edge on $P(a, b)$ or $P(c, b)$, contrary to Lemma 8. Thus $f \notin C(z_1, z_4)$.

We next show that two f -edges leave S at z_1 and z_4 . Suppose that no f -edge leaves S at z_1 . Then Lemma 8 implies that $f \notin C(z_1, z_2) \cup C(z_1, w_2)$ and $f \notin C(z_4, z_3) \cup C(z_4, w_3)$. Therefore either $f \in C(z_1, z_3)$ or $f \in C(z_1, w_3)$. If $f \in C(z_1, z_3)$, then $f \in C(z_2, z_4)$ (otherwise recoloring $G[a, f; z_2]$ would make $|P(a, b)| \geq 9$), and $G[b, f; w_2]$ violates Lemma 8. If $f \in C(z_1, w_3)$, then $f \in C(w_2, z_4)$ and $G[b, f; z_2]$ violates Lemma 8. Thus we have shown that an f -edge leaves S at z_1 . Similarly we can show that an f -edge leaves S at z_4 .

Let z_1v_2 and z_4v_3 be these two f -edges leaving S at z_1 and z_4 . $G[a, f; z_1]$ must contain two a -edges z_1z_2 and z_3z_4 on $P(a, b)$, and consequently contains the two f -edges. Since recoloring $G[a, f; z_1]$ does not make $|P(a, b)| > 7$, we have $b \in C(v_2, v_3)$. Note that $xz_1v_2v_3z_4z_5x$ is now a bf -alternating cycle. Furthermore $f \in C(z_2, z_3)$ and $f \in$

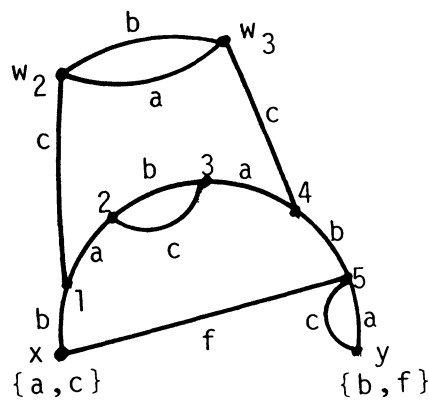


FIG. 3. Illustration for Case 2.1.

$C(w_2, w_3)$; otherwise, $G[b, f; z_2]$ or $G[b, f; w_2]$ would violate Lemma 8. Recoloring $G[a, f; z_1]$ or $G[c, f; z_1]$ can make $P(a, b)$ or $P(c, b)$ pass through v_2 and v_3 . Therefore $a, c \notin M(v_2) \cup M(v_3)$. Moreover $a, c \in C(v_2, v_3)$: otherwise, recoloring $G[a, b; v_2]$ or $G[b, c; v_2]$ and subsequently recoloring $G[a, f; z_1]$ or $G[c, f; z_1]$ would make $|P(a, b)| \geq 9$ or $|P(c, b)| \geq 9$. Thus we have shown that the coloring is as in Fig. 4. Since $|S \cup \{v_2, v_3\}| \geq 11$, there are two vertices of a common missing color. Exactly one of them must be either v_2 or v_3 . We may assume that the other is either w_2 or w_3 . Recolor the cycle $G[a, f; z_1]$, then these two vertices are now on critical paths, a contradiction.

Case 2.2. $P(c, b)$ contains 5 vertices. Neither the b -edge z_2z_3 nor z_4z_5 is contained in $P(c, b)$; otherwise, the component of $G[b, c]$ containing a leaving f -edge would contain exactly one b -edge on $P(a, b)$. Thus $P(c, b) = xz_1w_2w_3y$, where $w_2, w_3 \notin P(a, b)$. Clearly $a \notin M(w_2) \cup M(w_3)$. Furthermore $a \in C(w_2, w_3)$: otherwise, recoloring $G[a, b; w_2]$ would make $|P(c, b)| \geq 7$.

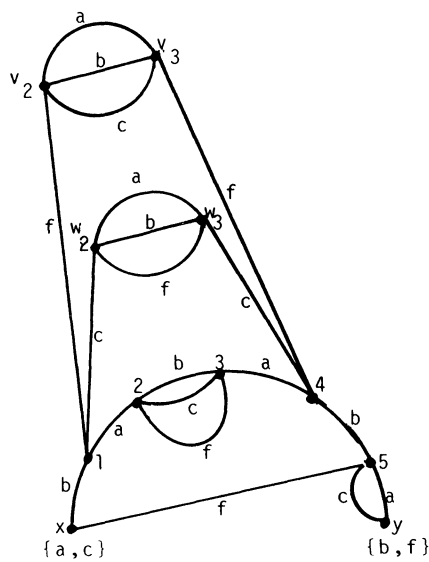


FIG. 4. Further illustration for Case 2.1.

We can show that no c -edge leaves H at z_2, z_3, z_4 or z_5 . If a c -edge leaves H at z_2 , then the new ab -critical path $P'(a, b)$ obtained by recoloring $G[b, c; z_3]$ satisfies

$$|P'(a, b) \cup P(a, b) \cup P(b, c)| \geq 11,$$

and hence a contradiction can easily be derived by considering a missing color of y other than b . Thus no c -edge leaves H at z_2 . Similarly no c -edge leaves H at z_5 , either. If two c -edges leave H at z_3 and z_4 , then $G[a, c; z_3]$ violates Lemma 8. Thus we have shown that no c -edge leaves H at z_2, z_3, z_4 or z_5 . Therefore either

- (i) $c \in C(z_5, z_2) \cap C(z_3, z_4)$,
- (ii) $c \in C(z_5, z_3) \cap C(z_2, z_4)$, or
- (iii) $c \in C(z_5, z_4) \cap C(z_2, z_3)$.

Let S be the subgraph of G induced by the nine vertices $P(a, b) \cup \{w_2, w_3\}$. There exists an f -edge leaving S , where $f \neq a, b, c$. Similarly as in Lemma 13, we can show that $f \in M(y)$. Suppose that $f \notin M(y)$. Then by Lemma 10 we may assume that no vertex in S misses color f . Define u, u', v, v' and g as in Lemma 11, then $v = x$ and neither $\text{Apath}(a, f; x)$ nor $\text{Apath}(c, f; x)$ contains any vertex in S except x . Furthermore $P'(f, b)$ contains no f -edges leaving S . Similarly as in the proof of Lemma 13, we can easily know that $P'(f, b) = 3$, that is, $f \in C(z_1, y)$, and that no f -edge leaves S at z_2, z_5, w_2 , or w_3 . Therefore exactly two f -edges leave S at z_3 and z_4 . Then $G[a, f; z_3]$ violates Lemma 8.

Similarly as in the proof of Lemma 14, we can easily show that $|P(a, f)|, |P(c, f)| \leq 5$. Suppose that $|P(a, f)| = 5$, then $P(a, f)$ must be either $xw_2w_3z_5y$ or $xw_3w_2z_5y$. In the former case $|P(c, f)| \geq 7$, a contradiction. In the latter case recoloring $G[c, f; w_2]$ makes $|P(c, b)| \geq 7$, a contradiction. Thus we have shown that $|P(a, f)| = 3$ and $P(a, f) = xz_5y$. Then $P(c, f)$ is either $xz_5z_2w_3y, xz_5z_3w_3y$ or $xz_5z_4w_3y$, according to cases (i)–(iii). Furthermore $f \in C(z_1, w_2)$. $G[a, f; z_3]$ in case (i) and $G[b, f; z_3]$ in case (iii) violate Lemma 8. In case (ii) the successive recoloring of $G[c, f; z_2]$ and $G[b, c; z_2]$ makes $|P(a, b)| \geq 9$, a contradiction.

Case 2.3. $P(c, b)$ contains 3 vertices. $P(c, b) = xz_1y$. We can easily show that two c -edges z_2w_2 and z_5w_3 leave H . Since recoloring the components of $G[b, c]$ containing z_2z_3 and z_4z_5 makes $P(a, b)$ pass through w_2 and w_3 , we have $a, b \in C(w_2, w_3)$. Since $z_1z_2w_2w_3z_5y z_1$ is an ac -cycle, $a, c \in C(z_3, z_4)$. Let S be the subgraph of G induced by the vertices in $P(a, b) \cup \{w_2, w_3\}$. Recoloring the bc -cycle $G[b, c; z_2]$ can make $P(a, b)$ pass through w_2 and w_3 . Now no two vertices in S have a common missing color; for, otherwise, considering a missing color of y other than b , we could derive a contradiction. Therefore there exists an f -edge leaving S , where $f \neq a, b, c$.

Similarly as in Lemma 13, we can show that $f \in M(y)$. Suppose that $f \notin M(y)$. Then by Lemma 10 we may assume that no vertex in S misses color f . Define u, u', v, v' and g as in Lemma 11, then $v = x$ and neither $\text{Apath}(a, f; x)$ nor $\text{Apath}(c, f; x)$ contains any vertex in S except x . Furthermore $|P'(f, b)| \leq 5$, and $P'(f, b)$ contains no f -edges leaving S . If $|P'(f, b)| = 5$, $P'(f, b)$ must be either $xz_1w_2w_3y$ or $xz_1w_3w_2y$; otherwise, the component of $G[b, f]$ containing the leaving f -edges violates Lemma 8. In either case the new $P'(a, b)$ obtained by recoloring the components of $G[b, f]$ containing the leaving f -edges satisfies $|S \cup P'(a, b)| \geq 11$, and we can derive a contradiction by considering a missing color of y other than b . Thus we have shown that $|P'(f, b)| = 3$, and $f \in C(z_1, y)$. Then, similarly as above, we can show that no f -edge leaves S at z_2 or z_5 . Therefore an f -edge joins z_2 with either (i) z_3 , (ii) z_4 , (iii) z_5 , (iv) w_2 , or (v) w_3 . Recoloring components of $G[b, f]$ in cases (i) and (ii), $G[a, f]$ in cases (iii) and (v), or $G[c, f]$ in case (iv), respectively, produces a contradiction. Thus we have proved that $f \in M(y)$.

Similarly as in the proof of Lemma 14, we can show that $|P(a, f)|, |P(c, f)| \leq 5$. If $|P(a, f)| = 5$, then $P(a, f)$ is either $xw_3w_2z_5y$ or $xw_2w_3z_5y$. In the former case $|P(c, f)| \geq 7$, a contradiction. In the latter case $G[a, f; z_3]$ contradicts Lemma 8. Thus we have shown that $|P(a, f)| = 3$. Therefore $P(a, f) = xz_5y$, $P(c, f) = xz_5w_3z_1y$, and consequently $f \in C(z_2, w_2)$. However then $G[a, f; z_3]$ violates Lemma 8.

6. Case 3. The longest critical path $P(a, b)$ contains five vertices. Let $P(a, b) = xz_1z_2z_3y$. By Lemma 7 one may assume that a c -edge leaves H where $c \in M(x)$. $P(c, b)$ must contain exactly five vertices; otherwise $|P(c, b)| = 3$, and $G[b, c; z_2]$ would contain exactly one b -edge on $P(a, b)$. Thus two c -edges z_1w_2 and yw_3 leave H . Furthermore $a, b \in C(w_2, w_3)$ and $b, c \in C(z_2, z_3)$. Denote by S the subgraph of G induced by the seven vertices in $P(a, b) \cup P(c, b)$. The definition of $q(G)$ implies that an edge leaves S . We consider the following four cases.

Case 3.1. An edge colored with a missing color f of y leaves S . Clearly $3 \leq |P(a, f)|, |P(c, f)| \leq 5$. If $|P(a, f)| = 3$, then $P(c, f) = xz_3z_2w_3y$, and two f -edges leave S at z_1 and w_2 and consequently $G[c, f; w_2]$ contains exactly one c -edge on $P(c, b)$, contrary to Lemma 8. Thus $|P(a, f)| = 5$. Therefore $P(a, f)$ contains two a -edges including z_3y . If $P(a, f)$ contains none of the f -edges leaving S , then $P(a, f)$ would contain either the a -edge z_1z_2 or w_2w_3 , and consequently either $G[b, f; w_2]$ or $G[a, f; z_1]$ would contain exactly one b - or a -edge on $P(c, b)$ or $P(a, b)$, contrary to Lemma 8. Thus $P(a, f)$ must contain an f -edge leaving S . By symmetry, $|P(c, f)| = 5$ and $P(c, f)$ must contain an f -edge leaving S .

Since $|S \cup P(a, f) \cup P(c, f)| \leq 10$, the coloring of G must be as shown in Fig. 5. Since $f \notin M(z_1) \cup M(z_2) \cup M(w_2)$, at least one f -edge leaves S at z_1, z_2 , or w_2 . The component of $G[a, f]$ or $G[c, f]$ containing the leaving f -edge includes exactly one a - or c -edge on $P(a, b)$ or $P(c, b)$, contrary to Lemma 8.

Case 3.2. An edge colored with a missing color of z_1 leaves S . We can reduce this case to Case 3.1 as follows. Erase the color b from xz_1 , and color the uncolored edge xy with b . Then the situation is the same as in Case 3.1.

Case 3.3. An edge colored with a missing color f of x leaves S . $P(f, b)$ must contain 5 vertices; otherwise, a component of $G[b, f]$ containing a leaving f -edge would include exactly one b -edge on $P(a, b)$ or $P(c, b)$.

Furthermore $P(f, b)$ must contain a leaving f -edge; otherwise, a component of $G[b, f]$ containing a leaving f -edge would include exactly one b -edge on $P(a, b)$ or $P(c, b)$.

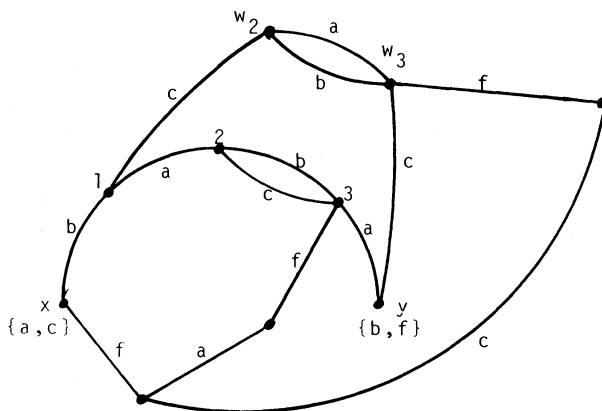


FIG. 5. Illustration for Case 3.1.

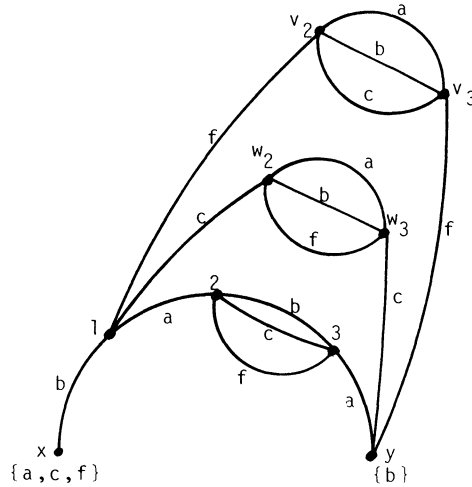


FIG. 6. Illustration for Case 3.2.

Thus $P(f, b)$ must be $xz_1v_2v_3y$, where $v_2, v_3 \notin V(S)$. We can easily show that $a, b, c \in C(v_2, v_3)$, $a, b, f \in C(w_2, w_3)$ and $b, c, f \in C(z_2, z_3)$, as shown in Fig. 6. Denote by T the subgraph of G induced by the nine vertices in $P(a, b) \cup \{w_2, w_3, v_2, v_3\}$. Since no two vertices in T have a common missing color, an edge must leave T by the definition of $q(G)$. Using Lemmas 10 and 11 similarly as in Lemma 13, we can rather easily show that a g -edge leaves T and $g \in M(y)$. Clearly

$$3 \leq |P(a, g)|, |P(c, g)|, |P(f, g)| \leq 5,$$

and none of $P(a, g)$, $P(c, g)$ and $P(f, g)$ leave T . If $|P(a, g)| = 3$, then $|P(c, g)|, |P(f, g)| \leq 5$ implies $g \in C(z_2, w_3) \cap C(z_2, v_3)$, a contradiction. Thus $|P(a, g)| = 5$. Similarly $|P(c, g)| = |P(f, g)| = 5$. Therefore $V(P(a, b)) = V(P(a, g))$, $V(P(c, b)) = V(P(c, g))$, and $V(P(f, b)) = V(P(f, g))$. Then no g -edge can leave T , a contradiction.

Case 3.4. An edge colored with a missing color f of a vertex $\neq x, z_1, y$ leaves S . In this case $f \notin M(x) \cup M(z_1) \cup M(y)$. By symmetry we may assume that an f -edge leaves S and $f \in M(w_3)$. Apath $(f, b; w_3)$ must end at y , otherwise, erase color c from edge yw_3 and color the uncolored edge xy with c ; then there would be no f -critical path between w_3 and y . Furthermore Apath $(f, b; w_3)$ must contain either three or five vertices. If Apath $(f, b; w_3)$ contained exactly three vertices, then f -edges would leave S at x, z_1, z_2 , or z_3 , and a component of $G[b, f]$ having a leaving f -edge would include exactly one b -edge in $P(a, b)$ or $P(c, b)$. Thus Apath $(f, b; w_3)$ ends at y and has five vertices.

Apath $(f, b; w_3)$ contains two b -edges including w_3w_2 . If Apath $(f, b; w_3)$ contained no f -edge leaving S , then exactly one of the two b -edges xz_1 and z_2z_3 would not be contained in Apath $(f, b; w_3)$; and two f -edges would leave S at its two ends; and consequently a component of $G[b, f]$ containing a leaving f -edge would include exactly one b -edge in $P(a, b)$, violating Lemma 8. Thus Apath $(f, b; w_3)$ contains two f -edges w_2v_2 and yv_3 leaving S , and $b \in C(v_2, v_3)$. Now no f -edge leaves S at x, z_1, z_2 , or z_3 ; otherwise, a component of $G[b, f]$ containing a leaving f -edge would include exactly one b -edge on $P(a, b)$ or $P(c, b)$. Erasing c from yw_3 and coloring xy with c makes cb - and fb -critical paths have five vertices. That is, the roles of x and w_3 can be interchanged. Therefore we can easily know that $a, c \in C(v_2, v_3)$. Moreover $f \in C(x, z_1)$; otherwise, $G[b, f; z_1]$ would violate Lemma 8 with respect to the cb -critical path between w_3 and y . Similarly

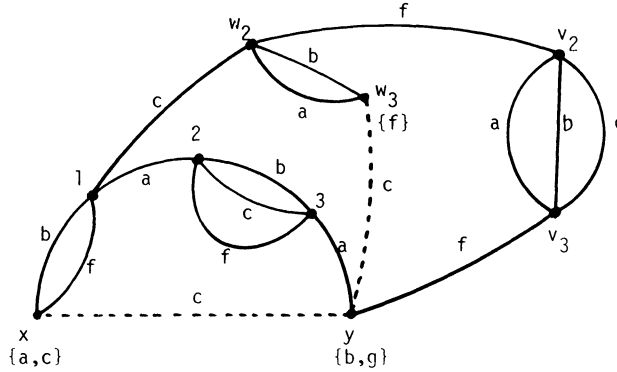


FIG. 7. Illustration for Case 3.4.

we have $f \in C(z_2, z_3)$. See Fig. 7. Furthermore, considering a missing color of y other than b , we can know that no two of the nine vertices in $P(a, b) \cup \{w_2, w_3, v_2, v_3\}$ have a common missing color. Denote by T the subgraph of G induced by the nine vertices. Since no two vertices in T have a common missing color, an edge, say a g -edge, leaves T by the definition of $q(G)$.

We next show that $g \in M(y)$. If g is missing at a vertex of T , then similarly as in Lemma 10 we can easily show that $g \in M(y)$. Thus suppose that there is no leaving edge whose color is missing at a vertex in T . Then we have the following lemma.

LEMMA 15. *Let uu' be a g -edge leaving T such that $u \in T - \{x, w_3\}$ and $u' \notin T$. Let h be a missing color of u such that $h \neq a, b, c, f, g$. Let vv' be the last g -edge leaving T and contained in $\text{Apath}(h, g; u)$, where $v \in T$ and $v' \notin T$. Then*

- (i) $v = x$ or w_3 ;
- (ii) if $v = x$, then neither $\text{Apath}(a, g; x)$ nor $\text{Apath}(c, g; x)$ contains a vertex in T except x ; and if $v = w_3$, then after erasing color c from w_3y and coloring xy c neither $\text{Apath}(f, g; w_3)$ nor $\text{Apath}(c, g; w_3)$ contains a vertex in T except w_3 ; and
- (iii) the new gb -critical path $P'(g, b)$ obtained by recoloring $\text{Apath}(a, g; x)$, $\text{Apath}(c, g; x)$, $\text{Apath}(f, g; w_3)$ or $\text{Apath}(c, g; w_3)$ contains no g -edge leaving T .

Proof. The proof is similar to Lemma 11. $\square$

Clearly $|P'(g, b)|$ must be either 3 or 5. In either case there would exist a component of $G[g, b]$ which contains leaving g -edges and exactly one b -edge on $P(a, b)$, $P(c, b)$ or $P(f, b)$, contradicting Lemma 8. Thus we have proved that $g \in M(y)$.

No g -edge leaves S , and $|P(a, g)|, |P(c, g)| \leq 5$. Therefore either (i) $P(a, g) = xz_3y$, $P(c, g) = xz_3z_2w_3y$, and $g \in C(z_1, w_2)$, (ii) $P(a, g) = xz_1z_2z_3y$ and $P(c, g) = xz_1w_2w_3y$, or (iii) $P(a, g) = xw_3w_2z_3y$, $P(c, g) = xw_3y$, and $g \in C(z_1, z_2)$. In either case $G[b, g; v_2]$ contains exactly one b -edge on $\text{Apath}(f, b; w_3)$, a contradiction.

Case 3.5. *An f -edge leaves S and no vertex in S misses color f .* We can reduce this case to one of Cases 3.1–3.4. Let uu' be an f -edge leaving S such that $u \in V(S) - \{x, y\}$ and $u' \notin V(S)$. Let g be any missing color of u . Then by Lemma 11 (i) and (ii) the last f -edge leaving S in $\text{Apath}(g, f; u)$ is xx' , and both $\text{Apath}(a, f; x)$ and $\text{Apath}(c, f; x)$ contain no vertex in S except x . Furthermore recoloring $\text{Apath}(a, f; x)$ or $\text{Apath}(c, f; x)$ does not make $|P'(f, b)| \geq 7$. It suffices to consider only the case where exactly three f -edges xx' , yv_3 , and z_1v_2 leave S at x, y and z_1 ; otherwise a component of $G[a, f]$, $G[b, f]$, or $G[c, f]$ containing a leaving f -edge would include exactly one a -, b -, or c -edge on $P(a, b)$ or $P(c, b)$. Clearly $b, c, f \in C(z_2, z_3)$, $a, b, f \in C(w_2, w_3)$, and $a, b, c \in C(v_2, v_3)$. Therefore the coloring must be as shown in Fig. 8. Denote by

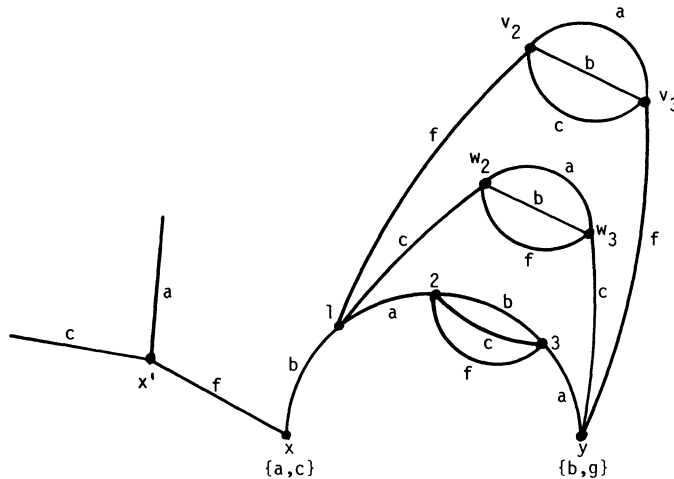


FIG. 8. Illustration for Case 3.5.

T the subgraph of G induced by these nine vertices. Since no two vertices in T have a common missing color, an edge must leave T by the definition of $q(G)$. Let g be the color of the edge. Recoloring cycle $G[a, c; z_1]$ (or $G[a, f; z_1]$) makes $P(a, b)$ pass through w_2 and w_3 (or v_2 and v_3). Using an argument similarly in Lemmas 10 and 11, we can show that a g -edge leaves T and $g \in M(y)$. Since no g -edge leaves S and $|P(a, g)|$, $|P(c, g)| \leq 5$, either (i) $P(a, g) = xz_3y$, $P(c, g) = xz_3z_2w_3y$, and $g \in C(z_1, w_2)$, (ii) $P(a, g) = xz_1z_2z_3y$ and $P(c, g) = xz_1w_2w_3y$, or (iii) $P(a, g) = xw_3w_2z_3y$, $P(c, g) = xw_3y$, and $g \in C(z_1, z_2)$. In either case two g -edges leave T at v_2 and v_3 , and $G[b, g; v_2]$ contains exactly one b -edge on $P(f, b)$ when recoloring Apath $(a, f; x)$, a contradiction.

7. Algorithmic remark. The proof given in the preceding sections yields a polynomial-time algorithm which edge-colors any given multigraph $G = (V, E)$ with at most $q(G)$ colors, where

$$q(G) = \text{MAX} \{p(G), \lfloor 1.1 d(G) + 0.8 \rfloor\}.$$

In this section we give some remarks on the algorithm.

The algorithm is iterative in a sense that it colors the edges of G one by one. As in [HNS] it is not necessary to compute $q(G)$ explicitly. (Probably $q(G)$ itself may not be computable in polynomial time.) Initially $q = \lfloor 1.1 d(G) + 0.8 \rfloor$ colors are available, and a new color is introduced only if $q(G)$ turns out to exceed the number q of the currently available colors. The algorithm colors an edge xy in five steps, each corresponding to the cases $|P(a, b)| = 3, 5, 7, 9$ or the case $|P(a, b)| \geq 11$. Each of the first four steps increases $|P(a, b)|$ two or more, and reduces to one of the latter cases. We can easily know that each of these steps repeats the recoloring of an alternating path or cycle at most constant times, say ten times. When we eventually have $|P(a, b)| \geq 11$, the last step makes x and y have a common missing color c . The step requires at most 12 recolorings of alternating paths or cycles, since by Lemma 2(b) $P(a, b)$ contains two vertices having a common missing color among the first 12 vertices starting from x , and each recoloring used in the proof of Lemma 1 decreases the distance between the two vertices at least one. Hence the algorithm colors one edge by repeating the recoloring of an alternating path or cycle at most constant times, say $4 \times 10 + 12 = 52$ times by rough estimate.

Using data structures mentioned in [NS] and [HNS] we can easily recolor an alternating path or cycle in $O(|V| + d(G))$ time. Therefore the algorithm runs in $O(|E|(d(G) + |V|))$. Furthermore the storage space is $O(|E|)$. Clearly the algorithm uses at most $\lfloor 1.1q^*(G) + 0.8 \rfloor$ colors. Therefore the number of used colors exceeds the chromatic index $q^*(G)$ no more than one plus 10 percent of $q^*(G)$. Thus the asymptotic performance ratio of the algorithm is 1.1. Similarly as in [NS] one can easily show that the number $q(G)$ of used colors is at most $(4/3)q^*(G)$. Therefore the “absolute performance ratio” is $4/3$.

Finally we remark on the possibility of generalizing our bound. Hakimi and Kariv have introduced a new type of edge colorings [Ha], [HK]. Goldberg’s bound mentioned in the introduction has been generalized to their case [NNS]. Our bound is also expected to be generalized to their case.

Appendix 1.

Proof of Lemma 1. Suppose to the contrary that $P(a, b)$ contains two vertices having a common missing color c . Let u and v be two vertices such that the distance between them on $P(a, b)$ is minimum; assume that x, u, v, y appear on $P(a, b)$ in this order. If u is followed by v on path $P(a, b)$, then we can break the critical path $P(a, b)$ simply by recoloring edge uv with color c . There exists no more ab -critical path in the resulting q -coloring of G' , contradicting Fact 5. Thus u is not followed by v on path $P(a, b)$. Then the vertex u' next to u on $P(a, b)$ is different from v . Let c' be any missing color of u' . Interchange the colors of the alternating path $\text{Apath}(c', c; u')$. If the path does not end at u then two successive vertices u and u' have a common missing color c , a contradiction. Otherwise, u' and v have a common missing color c and the distance between them is less than that between u and v . $\square$

Appendix 2.

Proof of Lemma 4. Suppose otherwise. Let $P(a, b)$ be an ab -critical path, where $a \in M(x)$ and $b \in M(y)$. Let $P(f, g)$ be an fg -critical path, where $f \in M(x)$ and $g \in M(y)$. Since $a \neq f$ or $b \neq g$, we may assume that $a \neq f$. By Lemma 1 neither $P(a, b)$ nor $P(f, g)$ contains two vertices of a common missing color. Therefore we may assume that $u \in P(a, b)$ and $v \in P(f, g)$. Let $c \in M(u) \cap M(v)$. Let u' be the vertex in $P(a, b)$ next to x ; let v' be the vertex in $P(f, g)$ next to x . Let c' and c'' be arbitrary missing colors of u' and v' , respectively. If $u \neq u'$ then $\text{Apath}(c', c; u')$ must end at u , because, otherwise, recoloring $\text{Apath}(c', c; u')$ would make two vertices u' and u on $P(a, b)$ have a common missing color c , contrary to Lemma 1. Similarly if $v \neq v'$ then $\text{Apath}(c'', c; v')$ must end at v . We may assume that $u' \neq v'$; for, otherwise, recoloring $\text{Apath}(c', c; u')$ would produce two vertices having a common missing color on $P(a, b)$ or $P(f, g)$. If $u \neq u'$ then interchange the colors of $\text{Apath}(c', c; u')$. If $v \neq v'$ then recolor $\text{Apath}(c'', c; v')$. Now we have $c \in M(u') \cap M(v')$. First suppose that $\text{Apath}(a, c; x)$ ends at v' . Then recolor $\text{Apath}(a, c; x)$. Now $c \in M(x)$ and $a \in M(v')$. Assign color c to the edge xu' of $P(a, b)$, and assign color b to the uncolored edge xy . The result is a q -coloring of G . Next suppose that $\text{Apath}(a, c; x)$ does not end at v' . Then recolor $\text{Apath}(a, c; x)$. Since $a, c \neq f, g$, $P(f, g)$ remains a critical path. However $P(f, g)$ contains two vertices x and v' of a common missing color c , a contradiction. $\square$

Appendix 3.

The remaining proof of Case 1.2.3.2. An f -edge joins x with either z_1, z_2, z_3, z_5 , or z_6 . In either case an appropriate recoloring would produce 11 or more vertices on critical paths, contrary to Lemma 5. The following are the recolorings for these cases; the detail is omitted.

Case 1.2.3.2.1. $f \in C(x, z_1)$. Noting that $P(c, f)$ cannot contain two leaving c -edges, we can easily show that $P(a, f)$ takes one of the following five forms:

- (a) $xz_1z_2z_3z_4z_5z_6z_7y$;
- (b) $xz_1z_2z_4z_3z_6z_5z_7y$;
- (c) $xz_1z_2z_5z_6z_3z_4z_7y$;
- (d) $xz_1z_2z_6z_5z_4z_3z_7y$; and
- (e) $xz_1z_2z_7y$.

In cases (a) and (d) recolor $G[b, c; z_3]$ and $G[a, b; z_3]$ successively. In cases (b) and (c) recolor $G[c, f; z_3]$. Finally in cases (e) $P(c, f)$ is either $xz_1z_5z_3z'_3z_4y$, $xz_1z_5z_4y$, $xz_1z_5z_6z'_6z_4y$, $xz_1z_5z'_3z_3z_4y$, or $xz_1z_5z'_3z_3z_4y$. If $P(c, f) = xz_1z_5z'_3z_3z_4y$, then recolor $G[a, f; z_6]$. Otherwise recolor $G[a, f; z_3]$.

Case 1.2.3.2.2. $f \in C(x, z_2)$. $P(a, f)$ takes one of the following five forms.

- (a) $xz_2z_1z_3z_4z_7y$;
- (b) $xz_2z_1z_4z_3z_6z_5z_7y$;
- (c) $xz_2z_1z_5z_6z_3z_4z_7y$;
- (d) $xz_2z_1z_6z_5z_3z_4z_7y$; and
- (e) $xz_2z_1z_7y$.

In cases (a)–(d) recolor $G[c, f; z_3]$. In case (e) recolor $G[a, f; z_3]$.

Case 1.2.3.2.3. $f \in C(x, z_3)$. $P(c, f)$ contains exactly one vertex z'_3 not on $P(a, b)$, and a leaving f -edge joins z'_3 with a vertex in H . The two a -edges z_3z_4 and z_7y are contained in $P(a, f)$. If both two a -edges z_1z_2 and z_5z_6 were also contained in $P(a, f)$, then there would exist no f -edge leaving H , a contradiction. If one of them was contained in $P(a, f)$, then two f -edges would leave H at the ends of the other a -edge, and consequently the component of $G[a, f]$ containing the a -edge violates Lemma 8. Thus we have shown that $P(a, f) = xz_3z_4z_7y$. Furthermore we can easily know that $f \notin C(z_1, z_2) \cup C(z_5, z_6)$, and that $P(c, f)$ is either $xz_3z'_3z_1z_5z_2z_7z_4y$ or $xz_3z'_3z_2z_7z_4y$. In the first case recolor $G[b, c; z_3]$. In the latter case recolor the components of $G[a, f]$ that contain the a -edges z_1z_2 and z_5z_6 . Then $|P(c, f)| \geq 11$.

Case 1.2.3.2.4. $f \in C(x, z_5)$. $P(a, f)$ takes one of the following three forms:

- (a) $xz_5z_6z_2z_1z_4z_3z_7y$;
- (b) $xz_5z_6z_3z_4 \cdots y$;
- (c) $xz_5z_6z_7y$.

In cases (a) and (b) recolor $G[c, f; z_3]$. If $xz_5 \notin G[c, f; z_6]$ in case (c), then recolor $G[c, f; z_6]$. Otherwise recolor $G[b, c; z_3]$ and $G[a, b; z_3]$ successively.

Case 1.2.3.2.5. $f \in C(x, z_6)$. Since an f -edge joins z'_6 with a vertex in H , $|P(a, f)| \leq 5$ or 7. If $|P(a, f)| = 7$, then two f -edges would leave H at the two ends of the a -edge that is on $P(a, b)$ but not on $P(a, f)$, contradicting Lemma 8. Thus $|P(a, f)| = 5$ and $P(a, f) = xz_6z_5z_7y$. Clearly an f -edge joins z'_6 with either z_1 , z_2 , or z_4 . If $f \in C(z'_6, z_2)$, then $f \in C(z_1, z_4)$, and an f -edge leaves H at z_3 , and consequently two f -edges leave H at the ends of the b -edge z_2z_3 , contradicting Lemma 8. Thus we can show that $P(c, f)$ is either

- (a) $xz_6z'_6z_1z_5z_7z_2z_4y$ or
- (b) $xz_6z'_6z_4y$.

In either case recolor $G[b, c; z_3]$ and $G[a, b; z_3]$ successively.

REFERENCES

- [AHU] A. V. AHO, J. E. HOPCROFT, AND J. D. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.
- [An] L. D. ANDERSEN, *On edge-colouring of graphs*, Math. Scand., 40 (1974), pp. 161–175.

- [FW] S. FIORINI AND R. J. WILSON, *Edge-Colourings of Graphs*, Pitman, London, 1977.
- [GJ] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability*, W. H. Freeman, San Francisco, 1979.
- [Go1] M. K. GOLDBERG, *On multigraphs with almost maximal chromatic class*, Discret Analiz, 23 (1973), pp. 3–7. (In Russian.)
- [Go2] ———, *Edge-coloring of multigraphs: Recoloring technique*, J. Graph Theory, 8 (1984), pp. 123–137.
- [Go3] ———, *An approximation algorithm for the edge-coloring problem*, Congressus Numerantium, 43 (1984), pp. 317–319.
- [HNS] D. S. HOCHBAUM, T. NISHIZEKI, AND D. S. SHMOYS, *A better than “best possible” algorithm to edge color multigraphs*, J. Algorithms, 7 (1986), pp. 79–104.
- [Ho] I. J. HOLYER, *The NP-completeness of edge-colourings*, SIAM J. Comput., 10 (1980), pp. 718–720.
- [Ha] S. LOUIS HAKIMI, *Further results on a generalization of edge-coloring*, in Graph Theory with Applications to Algorithms and Computer Science, Y. Alavi, G. Chartrand, L. Lesniak, D. Lick, and C. Wall, eds., John Wiley, New York, 1985, pp. 371–389.
- [HK] S. LOUIS HAKIMI AND ODED KARIV, *On a generalization of edge-coloring in graphs*, J. Graph Theory, 10 (1986), pp. 139–154.
- [NNS] S. NAKANO, T. NISHIZEKI, AND N. SAITO, *On the f -coloring of graphs*, IEEE Trans. Circuits Systems, 35 (1988), pp. 345–353.
- [NS] T. NISHIZEKI AND M. SATO, *An algorithm for edge-coloring multigraphs*, Trans. Inst. of Electronics and Communication Eng., J67-D, 4 (1984), pp. 466–471. (In Japanese.)
- [Se] P. D. SEYMOUR, *On multi-colourings of cubic graphs, and conjectures of Fulkerson and Tutte*, Proc. London Math. Soc., 3 (1979), pp. 423–460.
- [Sh] C. E. SHANNON, *A theorem on colouring lines of a network*, J. Math. Phys., 28 (1949), pp. 148–151.
- [Vi] V. G. VIZING, *On an estimate of the chromatic class of a p -graph*, Discret Analiz, 3 (1964), pp. 25–30. (In Russian.)

A HIERARCHY OF RELAXATIONS BETWEEN THE CONTINUOUS AND CONVEX HULL REPRESENTATIONS FOR ZERO-ONE PROGRAMMING PROBLEMS*

HANIF D. SHERALI† AND WARREN P. ADAMS‡

Abstract. In this paper a reformulation technique is presented that takes a given linear zero-one programming problem, converts it into a zero-one polynomial programming problem, and then relinearizes it into an extended linear program. It is shown that the strength of the resulting reformulation depends on the degree of the terms used to produce the polynomial program at the intermediate step of this method. In fact, as this degree varies from one up to the number of variables in the problem, a hierarchy of sharper representations is obtained with the final relaxation representing the convex hull of feasible solutions. The reformulation technique readily extends to produce a similar hierarchy of linear relaxations for zero-one polynomial programming problems. A characterization of the convex hull in the original variable space is also available through a projection process. The structure of this convex hull characterization (or its other relaxations) can be exploited to generate strong or facetial valid inequalities through appropriate surrogates in a computational framework. The surrogation process can also be used to study various classes of facets for different combinatorial optimization problems. Some examples are given to illustrate this point.

Key words. convex hull, zero-one programs, tight relaxations, facets

AMS(MOS) subject classifications. 90C10, 90C27

1. Introduction. This paper describes a technique for generating a hierarchy of polyhedral representations for linear and polynomial zero-one programming problems. In the linear case, the technique first multiplies the constraints in the problem, including the zero-one interval bounds on the variables, by a select set of d -degree polynomial terms or factors formed using the n zero-one variables, where $1 \leq d \leq n$. The resulting system is then linearized by defining new nonnegative variables for each existing cross-product term. It is shown that for each $d = 1, \dots, n$, by enforcing the binary restrictions on the original x -variables, we obtain an equivalent reformulation of the problem which has at least as tight a linear programming relaxation as that obtained by using factors of degree $d - 1$. (Here, the 0-degree factor is taken as unity.) Moreover, when $d = n$, we show that the resulting linear system represents a polytope whose extreme points are precisely the zero-one solutions feasible to the original problem, and hence characterizes the convex hull of feasible solutions. A projection of this convex hull characterization onto the space of the original variables with an identification of its facets is also evident. A similar approach produces a hierarchy of polyhedral representations extending up to the convex hull of feasible solutions for zero-one polynomial programming problems.

The foregoing approach is a generalization (in the pure zero-one case) of the linearization strategy suggested by Adams and Sherali (1987) for mixed-integer zero-one programming problems using $d = 1$. This technique was theoretically shown to generate tighter linear programming relaxations than alternative methods such as those due to Petersen (1971), Glover and Woolsey (1974), and Glover (1975). Computational strategies for exploiting the structure of this reformulation were presented for zero-one quadratic programming problems in Adams and Sherali (1986). The analysis in the present

* Received by the editors December 12, 1988; accepted for publication (in revised form) September 20, 1989. This material is based upon work supported by National Science Foundation grant ECS-8807090. The United States Government has certain rights in this material.

† Department of Industrial Engineering and Operations Research, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0118.

‡ Department of Math Sciences, Clemson University, Clemson, South Carolina 29634-1907.

paper demonstrates that while maintaining the basic structure of the reformulation, we can strengthen it as desired to the point that we obtain the convex hull of feasible solutions to the original zero-one problem.

Balas (1983) has proposed an alternative technique for obtaining a hierarchy of relaxations for mixed-integer zero-one programs via disjunctive programming methods. Here, the basic construct used from disjunctive programming is a description of the closure convex hull of a union of polyhedral sets in terms of a certain higher dimensional polyhedron. This construct is used as follows. Given a representation of the feasible region F as $F = \bigcap_{j \in T} S_j$, where S_j is the union of certain polyhedra for each j in an index set T , a relaxation of the closure convex hull of F is obtained by taking the intersection of the closure convex hull of each S_j , $j \in T$. This process is said to determine a hull-relaxation of F . If F is written as the intersection of its continuous relaxation constraints, along with the disjunctions that $x_j = 0$ or $x_j = 1$ for each binary variable x_j , this process yields the usual linear programming relaxation. On the other end of the spectrum, if F is represented as a union of polyhedra ($|T| = 1$) by essentially enumerating all possible feasible binary combinations, we obtain the closure convex hull of F via this process. The in-between relaxations are obtained by converting in a stage-wise fashion the intersections of each of various pairs of sets S_j to an equivalent union of polyhedra before taking the hull-relaxation. Evidently, there is a considerable degree of flexibility in designing these in-between relaxations. While our technique coincides with Balas' method at the two ends of this spectrum of relaxations, it differs in the sequence of intermediate relaxations.

Observe also that our technique is in the spirit of Chvatal's (1973) result which shows that any *given* facetial inequality for a linear, bounded, pure integer programming problem can be derived through a sequence of constraint surrogations followed by a rounding up of coefficients, where the new constraints can themselves be used in the surrogates. Our technique can similarly be viewed as an appropriate sequential process of constraint multiplication by single degree factors followed by variable redefinitions. However, unlike Chvatal, we provide an *explicit* characterization of the convex hull. Its structure enables us to not only verify the validity of facetial inequalities through a surrogation process, but can also be used constructively to generate facets for different classes of combinatorial zero-one problems, such as for the zero-one knapsack problem (Balas (1975, 1976), Balas and Zemel (1977), Hammer, Johnson, and Peled (1975), and Wolsey (1975)), the travelling salesman problem (Crowder and Padberg (1980), Grotschel and Padberg (1979), and Lawler et al. (1985)), the multiple choice problem (Johnson (1981)), set covering and set packing problems (Padberg (1973), (1979), (1980)), positive zero-one programs (Balas and Zemel (1984)), general zero-one problems (Crowder, Johnson, and Padberg (1983), Wolsey (1976), and Zemel (1978)), and certain special problems as in Balas (1985), among several others.

Furthermore, our characterization can be used computationally to provide sharper or tighter polyhedral representations for linear and polynomial zero-one programming problems (including mixed-integer cases), beyond that described in Adams and Sherali (1987), and to generate strong valid inequalities for such problems. From an algorithmic viewpoint, the importance of having tight linear programming relaxations which provide at least a partial convex hull representation has long been recognized. This has led to some crucial and critical research on model construction and problem formulation as in Jeroslow (1980), (1983), (1984a), (1984b), (1985), Jeroslow and Lowe (1984), (1985), Meyer (1975), (1976), (1981), Meyer, Thakkar, and Hallman (1980), and Williams (1974), (1985). Cutting plane generation schemes striving to provide tighter relaxations have also been proposed by Martin and Schrage (1982), (1983), Padberg

(1973), (1979), (1980), Van Roy and Wolsey (1983) and Wolsey (1975), (1976). Automatic reformulation procedures utilizing such constraint generation schemes within a branch and bound framework are presented in Crowder, Johnson, and Padberg (1983), Hoffman and Padberg (1985), Johnson and Suhl (1980), Johnson, Kostreva, and Suhl (1985), Oley and Sjouquist (1982), Spielberg and Suhl (1980), and Van Roy and Wolsey (1984a), (1984b). For some specially structured problems, a variable redefinition technique has also been proposed by Martin (1984), and Eppen and Martin (1985), where a linear transformation on variables is used to yield an equivalent formulation that tightens the continuous relaxation by constructing a partial convex hull of a subset of constraints. Our work is intended to lead to research in the same spirit as the foregoing papers.

Our presentation is organized as follows. Section 2 presents the hierarchy of tighter relaxations for linear zero-one problems. Section 3 establishes the convex hull property of the final relaxation, providing a characterization of the facets of the convex hull of X . This is then generalized to zero-one polynomial programming problems in § 4. Some illustrative examples are provided in § 5, and § 6 provides some concluding remarks pertaining to our ongoing research.

2. A hierarchy of relaxations. Consider the feasible region X of a linear zero-one programming problem in n variables as defined below, where e is a vector of ones in R^n .

$$(2.1) \quad X = \left\{ x \in R^n: \sum_{j=1}^n \alpha_{rj} x_j \geq \beta_r, r = 1, \dots, R_1, \sum_{j=1}^n a_{rj} x_j = b_r, \right. \\ \left. r = 1, \dots, R_2, 0 \leq x \leq e, x \text{ integer} \right\}.$$

We will now define a series of relaxations of X by using various d -degree polynomial factors to operate on the constraints of X . For $d = 0$, we will assume that these factors are unity so that we obtain the linear programming relaxation X_0 of X as given by (2.1) without the integrality restriction. For $d = 1, \dots, n$, let us define the polynomial factor $F_d(J_1, J_2)$ as

$$(2.2) \quad F_d(J_1, J_2) = \left(\prod_{j \in J_1} x_j \right) \left(\prod_{j \in J_2} (1 - x_j) \right) \quad \text{for } J_1, J_2 \subseteq N \equiv \{1, \dots, n\}, \\ J_1 \cap J_2 = \phi, \quad |J_1 \cup J_2| = d.$$

Clearly, for any $d = 1, \dots, n$ and any J_1, J_2 satisfying the above restrictions, $F_d(J_1, J_2) \geq 0$ for all binary x . As an abbreviating terminology, we will refer to any (J_1, J_2) satisfying the conditions in (2.2) as being of *order* d . Note that for each $d = 1, \dots, n$, there are $\binom{n}{d} \cdot 2^d$ such factors, and these factors are comprised using a total of $\sum_{i=1}^d \binom{n}{i}$ distinct variable terms. In particular, for $d = 1$ there are $2n$ such factors given by x_j and $(1 - x_j)$ for $j = 1, \dots, n$, comprised of the n terms $x_1, \dots, x_n$, while for $d = n$, there are 2^n such factors, and $2^n - 1$ terms. Now, for any $d \in \{1, \dots, n\}$, construct a relaxation X_d of X via the following two steps.

Step 1. Multiply each of the $R_1 + R_2$ constraints in X_0 (i.e., excluding the bounding restrictions $0 \leq x \leq e$) by each of the (nonnegative) factors defined in (2.2), using the relationship $x_j^2 = x_j$, and so $x_j(1 - x_j) = 0$, for each $j = 1, \dots, n$. Include in the resulting set of polynomial relationships the constraints representing nonnegativities on all possible factors of degree $D = \min \{d + 1, n\}$; that is, $F_D(J_1, J_2) \geq 0$ for each (J_1, J_2) of order D .

Step 2. Linearize the resulting polynomial constraint expressions by substituting w_J for the nonlinear term $\pi_{j \in J} x_j$ for each set J with $|J| \geq 2$, where the elements of the

set J are arranged in increasing order. Denoting in general the linear expression for any factor $F_d(J_1, J_2)$ obtained upon making such a substitution in terms of the variables (x, w) as $f_d(J_1, J_2)$, we obtain the resulting polyhedral set X_d given in (2.3) below.

$$(2.3a) \quad X_d = \left\{ (x, w) : \text{for each } (J_1, J_2) \text{ of order } d \text{ we have} \right. \\ \left. \left(\sum_{j \in J_1} \alpha_{rj} - \beta_r \right) f_d(J_1, J_2) + \sum_{j \in N - (J_1 \cup J_2)} \alpha_{rj} f_{d+1}(J_1 + j, J_2) \geq 0 \right. \\ \left. \text{for } r = 1, \dots, R_1, \right.$$

$$(2.3b) \quad \text{and} \left(\sum_{j \in J_1} a_{rj} - b_r \right) f_d(J_1, J_2) + \sum_{j \in N - (J_1 \cup J_2)} a_{rj} f_{d+1}(J_1 + j, J_2) = 0 \\ \left. \text{for } r = 1, \dots, R_2, \right.$$

and for each (J_1, J_2) of order D , where $D = \min \{d + 1, n\}$, we have

$$(2.3c) \quad f_D(J_1, J_2) \geq 0 \Big\}.$$

Observe that in (2.3) if we put $d = 0$ and define $(J_1, J_2) = (\phi, \phi)$ to be the only pair of order zero, and let $f_0(\phi, \phi) \equiv 1$, we obtain precisely the linear programming relaxation X_0 . Further observe that for any $d = 1, \dots, n$, the operations conducted on the set X in Steps 1 and 2 to obtain the set X_d do not eliminate any binary x solutions feasible to X . Also, note that at Step 1 above, if we had symmetrically multiplied each of the bounding constraints $0 \leq x_j \leq 1, j = 1, \dots, n$, by each of the factors in (2.2), we would have obtained relationships enforcing nonnegativities on all factors of degree d and all factors of degree $D = \min \{d + 1, n\}$. However, when $d < n$, Lemma 1 below shows that it is sufficient to include only the nonnegativities on the factors of degree $(d + 1)$ as in (2.3c) above.

Finally, note that we could have more generally included the inequalities $(\sum_{j=1}^n \alpha_{rj} x_j - \beta_r) \geq 0, r = 1, \dots, R_1$, along with the inequalities $x_j \geq 0$ and $(1 - x_j) \geq 0, j = 1, \dots, n$, in various combinations to define the factors of the type (2.2) to be used in the above reformulation scheme. Although the new constraints thus generated might help tighten the representations obtained for $d < n$, and so may be algorithmically advantageous, it follows from our development in the sequel that these constraints are all implied by the constraints of X_n . (We remark, however, that for any $d \leq n$, the constraints generated via the products of such new factors with the *equality* constraints in X are already implied by the constraints present in X_d above.)

LEMMA 1. *For any $1 \leq d < n$, the constraints $f_d(J_1, J_2) \geq 0$ for all (J_1, J_2) of order d are implied by the constraints $f_{d+1}(J_1, J_2) \geq 0$ for all (J_1, J_2) of order $(d + 1)$.*

Proof. Consider any (J_1, J_2) of order $1 \leq d < n$. Let $k \in N - (J_1 \cup J_2)$. Then we have

$$(2.4a) \quad F_{d+1}(J_1 + k, J_2) + F_{d+1}(J_1, J_2 + k) = [x_k + (1 - x_k)] F_d(J_1, J_2) = F_d(J_1, J_2).$$

Hence, we have,

$$(2.4b) \quad f_d(J_1, J_2) \equiv f_{d+1}(J_1 + k, J_2) + f_{d+1}(J_1, J_2 + k) \geq 0$$

and the proof therefore follows. $\square$

Next, in order to establish the equivalence of X_d to X under binary restrictions on x , and to establish the necessary machinery for our convex hull characterization, consider the following results.

LEMMA 2. For any $d \in \{1, \dots, n\}$, consider the constraint set

$$(2.5) \quad Z_d = \{(x, w) : f_d(J_1, J_2) \geq 0 \text{ for all } (J_1, J_2) \text{ of order } d\}.$$

Let $\hat{x}$ be any binary vector. Then $(\hat{x}, \hat{w}) \in Z_d$ if and only if $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J such that $|J| = 1, \dots, d$, where $\hat{w}_J \equiv \hat{x}_J$ when $|J| = 1$.

Proof. If $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J such that $|J| = 1, \dots, d$, we have that the value at $(\hat{x}, \hat{w})$ of $f_d(J_1, J_2)$ equals that of $F_d(J_1, J_2)$ for all (J_1, J_2) , and so $(\hat{x}, \hat{w}) \in Z_d$. Hence, let us now prove the converse by induction. Note that it is obviously true for $d = 1$, and so assume that it is true for $Z_1, \dots, Z_{d-1}$ and consider Z_d , $2 \leq d \leq n$. We are given $(\hat{x}, \hat{w}) \in Z_d$ and we need to show that $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J such that $|J| = 1, \dots, d$. Note from (2.4b) that $(\hat{x}, \hat{w})$ satisfies $f_{d-1}(J_1, J_2) \geq 0$ for all (J_1, J_2) of order $(d-1)$, and so by the induction hypothesis, we have $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all J with $|J| = 1, \dots, d-1$. Hence, consider any J with $|J| = d \geq 2$, and let $\{k, l\} \subseteq J$. Since $F_d(J-k, k) = (1 - x_k)(\pi_{j \in J-k} x_j)$, the constraint $f_d(J-k, k) \geq 0$ is

$$(2.6) \quad w_J \leq w_{J-k} \text{ for any } k \in J.$$

Furthermore, since $F_d(J-k-l, \{k, l\}) = \pi_{j \in J-k-l} x_j (1 - x_k - x_l + x_k x_l)$, the constraint $f_d(J-k-l, \{k, l\}) \geq 0$ is

$$(2.7) \quad w_J \geq w_{J-l} + w_{J-k} - w_{J-k-l} \text{ for any } \{k, l\} \in J,$$

where $w_{J-k-l} \equiv 1$ if $J = \{k, l\}$ itself. Now, if $\pi_{j \in J} \hat{x}_j = 0$, then $\hat{w}_{J-k} = \pi_{j \in J-k} \hat{x}_j = 0$ for some $k \in J$, and since $\hat{w}_J \geq 0$ from the constraint $f_d(J, \phi) \geq 0$, we have from (2.6) that $\hat{w}_J = 0$. On the other hand, if $\pi_{j \in J} \hat{x}_j = 1$, then by the induction hypothesis, we have $\hat{w}_{J-l} = \hat{w}_{J-k} = \hat{w}_{J-k-l} = 1$ and so from (2.6) and (2.7) we obtain $\hat{w}_J = 1$. Therefore $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all J with $|J| = d$ as well, and the proof is complete. $\square$

We are now ready to establish the hierarchy in the relaxations X_d of X , for $d = 1, \dots, n$. This is addressed in the following result.

THEOREM 1. Let X_0 be the linear programming relaxation of X , and let $\text{conv}(X)$ denote the convex hull of X . Define

$$(2.8) \quad X_{Pd} = \{x : (x, w) \in X_d\}$$

to be the projection of X_d onto the x -space for $d = 1, \dots, n$. Then we have,

$$(2.9) \quad X_{P0} \supseteq X_{P1} \supseteq X_{P2} \supseteq \dots \supseteq X_{Pn} \supseteq \text{conv}(X)$$

where $X_0 \equiv X_{P0}$.

Proof. Consider any $d \in \{1, \dots, n\}$, and let $(\tilde{x}, \tilde{w}) \in X_d$, so that $\tilde{x} \in X_{Pd}$. We will first show that the same values for $\tilde{x}$ and the components of $\tilde{w}$ defining X_{d-1} satisfy the constraints in X_{d-1} , hence implying that $\tilde{x} \in X_{P(d-1)}$, and consequently that $X_{Pd} \subseteq X_{P(d-1)}$. Toward this end, consider any inequality $\sum_{j=1}^n \alpha_{rj} x_j \geq \beta_r$, $r \in \{1, \dots, R_1\}$. Let (J_1, J_2) be any pair of order $(d-1)$, and denote $J_3 = N - (J_1 \cup J_2)$. (J_1 and J_2 are both empty if $d = 1$.) Then by (2.3a) of X_d , we have for any $k \in J_3$ that

$$\left(\sum_{j \in J_1} \alpha_{rj} - \beta_r \right) f_d(J_1, J_2 + k) + \sum_{j \in J_3 - k} \alpha_{rj} f_{d+1}(J_1 + j, J_2 + k) \geq 0$$

and

$$\left(\sum_{j \in J_1} \alpha_{rj} - \beta_r \right) f_d(J_1 + k, J_2) + \alpha_{rk} f_d(J_1 + k, J_2) + \sum_{j \in J_3 - k} \alpha_{rj} f_{d+1}(J_1 + k + j, J_2) \geq 0.$$

Summing these last two inequalities and using (2.4b), we get

$$(2.10) \quad \left(\sum_{j \in J_1} \alpha_{rj} - \beta_r \right) f_{d-1}(J_1, J_2) + \sum_{j \in J_3} \alpha_{rj} f_d(J_1 + j, J_2) \geq 0$$

where $f_0(\phi, \phi) \equiv 1$. However, note that (2.10) is precisely the constraint $\sum_{j=1}^n \alpha_{rj} x_j \geq \beta_r$ if $d = 1$ (whence $J_1 = J_2 = \phi$), and is the constraint (2.3a) of the set X_{d-1} if $d \geq 2$. Hence, $(\tilde{x}, \tilde{w})$ satisfies the constraints (2.3a) in X_{d-1} . In an identical manner, we get that $(\tilde{x}, \tilde{w})$ satisfies the constraints (2.3b) in X_{d-1} . Furthermore, if $d \in \{1, \dots, n-1\}$, then by Lemma 1, $(\tilde{x}, \tilde{w})$ satisfies the constraints (2.3c) of X_{d-1} , and if $d = n$, then (2.3c) for X_d and X_{d-1} are identical. Hence, we have that $(\tilde{x}, \tilde{w})$ satisfies all the constraints in X_{d-1} , and so $\tilde{x} \in X_{P(d-1)}$. This establishes that $X_{Pd} \subseteq X_{P(d-1)}$ for $d = 1, \dots, n$.

To complete the proof, we need to show that $\text{conv}(X) \subseteq X_{Pn}$. If $\text{conv}(X) = \phi$, the result follows trivially. Otherwise, let $\hat{x} \in X$, and define $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all the $2^n - (n+1)$ sets J such that $|J| = 2, \dots, n$. Then by construction, $(\hat{x}, \hat{w}) \in X_n$ and so $\hat{x} \in X_{Pn}$. Hence $X \subseteq X_{Pn}$, and so $\text{conv}(X) \subseteq X_{Pn}$, and the proof is complete. $\square$

COROLLARY 1. *Let $d \in \{1, \dots, n\}$. Then $(\hat{x}, \hat{w})$ with $\hat{x}$ binary is a feasible solution to X_d if and only if $\hat{x} \in X$ and $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J such that $|J| = 2, \dots, D$, where $D = \min\{d+1, n\}$.*

Proof. If $(\hat{x}, \hat{w})$ with $\hat{x}$ binary is a feasible solution to X_d , then since $X = \{x \in X_0: x \text{ binary}\}$ and since $\hat{x} \in X_0$ from (2.9) of Theorem 1, we have that $\hat{x} \in X$. Moreover, since $(\hat{x}, \hat{w}) \in X_D$ by (2.3c), we have from Lemma 2 that $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J with $|J| = 2, \dots, D$. The converse of the corollary follows directly from the construction of X_d , and the proof is complete. $\square$

3. Convex hull characterizations. We will now show that in the hierarchy described in Theorem 1, the final relaxation X_{Pn} is indeed the convex hull of X . We begin by first demonstrating that the system Z_n defined in (2.5) describes a polytope with all binary valued vertices. As we will see later, this forms a fundamental system which characterizes the convex hull of feasible solutions for any linear or polynomial zero-one programming problem. Toward this end, consider the following result which describes a useful linear transformation.

LEMMA 3. *Consider the linear transformation*

$$(3.1a) \quad y_{J_1} = f_n(J_1, \bar{J}_1) \text{ for all } J_1 \subseteq N \equiv \{1, \dots, n\}, J_1 \neq \phi, \bar{J}_1 = N - J_1.$$

Then this describes a nonsingular linear transformation with an inverse

$$(3.1b) \quad w_{J_1} = \sum_{J \subseteq \bar{J}_1} y_{J_1 \cup J} \text{ for all } J_1 \subseteq N, J_1 \neq \phi$$

where $w_J \equiv x_J$ when $|J| = 1$.

Proof. Note that writing (3.1a) as $y = Bw$, (3.1b) asserts that $Ty = TBw = w$, thus implying that the square matrices T and B are nonsingular. Hence, to prove the result, let us show that (3.1b) holds under (3.1a).

Consider any $J_1 \subseteq N$, $J_1 \neq \phi$, and let y_J for all $J \subseteq N$ be as defined in (3.1a). Observing that for each $J \subseteq N$, $f_n(J, \bar{J}) = \sum_{L \subseteq J} (-1)^{|L|} w_{J \cup L}$, we get by direct algebraic manipulation that

$$\begin{aligned}
 \sum_{J \subseteq \bar{J}_1} y_{J_1 \cup J} &= \sum_{J \subseteq \bar{J}_1} f_n(J_1 \cup J, \overline{J_1 \cup J}) = \sum_{J \subseteq \bar{J}_1} \sum_{L \subseteq J_1 \cup J} (-1)^{|L|} w_{J_1 \cup J \cup L} \\
 (3.2) \quad &= \sum_{M \subseteq \bar{J}_1} w_{J_1 \cup M} \left[\sum_{\substack{J \subseteq M \\ L = M - J}} (-1)^{|L|} \right] = \sum_{M \subseteq \bar{J}_1} w_{J_1 \cup M} \left[\sum_{L \subseteq M} (-1)^{|L|} \right] \\
 &= w_{J_1} + \sum_{\substack{M \subseteq \bar{J}_1 \\ M \neq \phi}} w_{J_1 \cup M} (1 - 1)^{|M|} = w_{J_1}.
 \end{aligned}$$

This completes the proof. $\square$

Example 1. To illustrate, note that for $n = 3$, putting

$$\begin{aligned}
 y_{123} &= w_{123}, y_{23} = (w_{23} - w_{123}), y_{13} = (w_{13} - w_{123}), y_{12} = (w_{12} - w_{123}), \\
 (3.3) \quad y_3 &= (x_3 - w_{13} - w_{23} + w_{123}), y_2 = (x_2 - w_{12} - w_{23} + w_{123}), \text{ and} \\
 y_1 &= (x_1 - w_{12} - w_{13} + w_{123})
 \end{aligned}$$

where the expressions for $f_n(J_1, \bar{J}_1)$ in (3.3) may be readily verified, we get

$$\begin{aligned}
 w_{123} &= y_{123}, w_{23} = (y_{23} + y_{123}), w_{13} = (y_{13} + y_{123}), w_{12} = (y_{12} + y_{123}), \\
 (3.4) \quad x_3 &= (y_3 + y_{13} + y_{23} + y_{123}), x_2 = (y_2 + y_{12} + y_{23} + y_{123}), \text{ and} \\
 x_1 &= (y_1 + y_{12} + y_{13} + y_{123}).
 \end{aligned}$$

Next, consider the following (well known) result.

LEMMA 4. Consider the set $S_x = \{x: Gx \geq g\} \neq \phi$. Under a nonsingular linear transformation $y = H^{-1}x$, i.e., $x = Hy$, let the set S_x be transformed into the set $S_y = \{y: GHy \geq g\}$. Then S_x is bounded if and only if S_y is bounded, $\hat{x}$ is a vertex of S_x if and only if $\hat{y} = H^{-1}\hat{x}$ is a vertex of S_y , and the row $G_r x \geq g_r$ defines a facet of S_x if and only if the corresponding row $G_r Hy \geq g_r$ defines a facet of S_y .

Proof. The proof follows directly from the definitions of extreme points, extreme directions, and facets.

THEOREM 2. Consider the set

$$Z_n = \{(x, w): f_n(J, \bar{J}) \geq 0 \text{ for all } J \subseteq N, \bar{J} = N - J\}.$$

Then Z_n is bounded and $(\hat{x}, \hat{w})$ is an extreme point of Z_n if and only if $(\hat{x}, \hat{w})$ is binary with $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J such that $|J| = 2, \dots, n$. Moreover, every defining inequality of Z_n is a facetial inequality.

Proof. Note that the sum of the expressions for all the factors $F_n(J, \bar{J})$ over $J \subseteq N$ is unity, and so,

$$F_n(\phi, N) = 1 - \sum_{\substack{J \subseteq N \\ J \neq \phi}} F_n(J, \bar{J}).$$

Consequently, we may write Z_n as

$$(3.5a) \quad Z_n = \left\{ (x, w): f_n(J, \bar{J}) \geq 0 \text{ for all } J \subseteq N, J \neq \phi, \text{ and } \sum_{\substack{J \subseteq N \\ J \neq \phi}} f_n(J, \bar{J}) \leq 1 \right\}.$$

Now, using the nonsingular linear transformation (3.1), the set Z_n gets transformed into the y -space as

$$(3.5b) \quad S_n = \left\{ y: \sum_{\substack{J \subseteq N \\ J \neq \phi}} y_J \leq 1, \text{ and } y_J \geq 0 \text{ for all } J \subseteq N, J \neq \phi \right\}.$$

By Lemma 4, since S_n is bounded, so is Z_n , and since all the inequalities of S_n are facetial, so are the inequalities defining Z_n . Furthermore, the vertices of S_n are either $y = 0$ or y equal to a unit vector (one component unity and the others zero). Hence by Lemma 4 and (3.1b), the extreme points of Z_n are binary. Therefore, if $(\hat{x}, \hat{w})$ is an extreme point of Z_n , it is binary and by Lemma 2, $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J such that $|J| = 2, \dots, n$. Conversely, if $(\hat{x}, \hat{w})$ is binary with $\hat{w}_J = \pi_{j \in J} \hat{x}_j$, for all J with $|J| = 2, \dots, n$ then $(\hat{x}, \hat{w}) \in Z_n$ by construction, and moreover, from (3.1a), the corresponding $\hat{y}_{J_1} \equiv f_n(J_1, \bar{J}_1) \equiv F_n(J_1, \bar{J}_1)$ for all $J_1 \subseteq N, J_1 \neq \phi$. Hence, $\hat{y}$ is either the zero vector or a unit vector. Consequently, $\hat{y}$ is an extreme point of S_n which implies that $(\hat{x}, \hat{w})$ is an extreme point of Z_n by Lemma 4. This completes the proof. $\square$

COROLLARY 2. *Let $(x, w) \in Z_n$. Then for any nonempty sets J_1, J'_1 such that $J'_1 \subseteq J_1 \subseteq N$, we have $w_{J'_1} \geq w_{J_1}$, where $w_J \equiv x_J$ when $|J| = 1$.*

Proof. Using the transformation (3.1) as in (3.5), we have from the nonnegativity of y , and from (3.1b), since $\bar{J}'_1 = (J_1 - J'_1) \cup \bar{J}_1$, where $(J_1 - J'_1)$ is well defined, that

$$w_{J'_1} = \sum_{J' \subseteq \bar{J}'_1} y_{J' \cup J'} \geq \sum_{\substack{J' = (J_1 - J'_1) \cup J \\ J \subseteq \bar{J}_1}} y_{J' \cup J'} = \sum_{J \subseteq \bar{J}_1} y_{J_1 \cup J} = w_{J_1}. \quad \square$$

The above corollary establishes a relationship between variable values which can be useful in deriving implied inequalities. Observe that it is a generalization of inequalities (2.6) in that by restricting J'_1 and J_1 to satisfy $|J'_1| + 1 = |J_1|$, (2.6) results. Let us now establish the main result of this section.

THEOREM 3. *Let X_n and X_{Pn} be as defined by (2.3) and (2.8), respectively. Then $(\hat{x}, \hat{w})$ is an extreme point of X_n if and only if $(\hat{x}, \hat{w})$ is a binary feasible solution to X_n and, moreover, any such solution satisfies $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J such that $|J| = 2, \dots, n$. Furthermore, $X_{Pn} = \text{conv}(X)$.*

Proof. Note that from (2.3), we may write X_n as

$$(3.6) \quad X_n = \left\{ (x, w): \text{for each } J \subseteq N, \bar{J} = N - J, \text{ we have,} \right. \\ \left(\sum_{j \in J} \alpha_{rj} - \beta_r \right) f_n(J, \bar{J}) \geq 0 \text{ for } r = 1, \dots, R_1, \\ \left(\sum_{j \in J} a_{rj} - b_r \right) f_n(J, \bar{J}) = 0 \text{ for } r = 1, \dots, R_2, \text{ and } f_n(J, \bar{J}) \geq 0 \left. \right\}.$$

Now, define

$$(3.7) \quad E = \left\{ J \subseteq N: \left(\sum_{j \in J} \alpha_{rj} - \beta_r \right) < 0 \text{ for some } r \in \{1, \dots, R_1\} \text{ or} \right. \\ \left. \left(\sum_{j \in J} a_{rj} - b_r \right) \neq 0 \text{ for some } r \in \{1, \dots, R_2\} \right\},$$

and let $\bar{E}$ denote the complement of E with respect to the power set of N . Observe that for any $J \subseteq N$, if $J \in E$ then the corresponding $(R_1 + R_2 + 1)$ constraints in (3.6) are equivalent to the constraint $f_n(J, \bar{J}) = 0$, while if $J \in \bar{E}$, these are equivalent to the constraint $f_n(J, \bar{J}) \geq 0$. Hence, we may rewrite X_n as

$$(3.8) \quad X_n = \{(x, w): f_n(J, \bar{J}) = 0 \text{ for } J \in E, f_n(J, \bar{J}) \geq 0 \text{ for } J \in \bar{E}\}.$$

But this means that X_n is a subface of Z_n , and so the extreme points of X_n are precisely those extreme points of Z_n which are feasible to X_n . Hence, if $(\hat{x}, \hat{w})$ is an extreme point of X_n , then from Theorem 2, $(\hat{x}, \hat{w})$ is binary with $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J with $|J| = 2, \dots, n$. Conversely, if $(\hat{x}, \hat{w})$ is a binary feasible solution to X_n , then $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J with $|J| = 2, \dots, n$ by Corollary 1, and moreover from Theorem 2, $(\hat{x}, \hat{w})$ is an extreme point of Z_n , and hence of X_n as well.

Finally, note that if $\hat{x}$ is an extreme point of X_{P_n} , then since X_{P_n} is a projection of X_n , there exists a $\hat{w}$ such that $(\hat{x}, \hat{w})$ is an extreme point of X_n . From above, $(\hat{x}, \hat{w})$ is therefore binary with $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J with $|J| = 2, \dots, n$, and so by Corollary 1, $\hat{x} \in X$. Hence, the extreme points of X_{P_n} belong to X , and since X_{P_n} is bounded by Theorem 2, we get $X_{P_n} \subseteq \text{conv}(X)$. From (2.9) of Theorem 1, this means that $X_{P_n} = \text{conv}(X)$ and the proof is complete. $\square$

Example 2. Consider the set $X = \{x = (x_1, x_2): 2x_1 + 2x_2 \geq 1, x \text{ binary}\}$, and note that $\text{conv}(X) = \{x: x_1 + x_2 \geq 1, x \leq e\}$. The set

$$X_0 = \{x: 2x_1 + 2x_2 \geq 1, 0 \leq x \leq e\}$$

strictly contains $\text{conv}(X)$. Now, in order to construct X_1 , we use the factors $x_1, x_2, (1 - x_1)$ and $(1 - x_2)$ of degree 1 to multiply the constraint $2x_1 + 2x_2 \geq 1$, and include the nonnegativities on the factors $x_1x_2, x_1(1 - x_2), (1 - x_1)x_2$ and $(1 - x_1) \times (1 - x_2)$ of degree 2. Linearizing these sets of constraints as in Step 2 of the reformulation procedure gives constraint sets (2.3a) and (2.3c), respectively, as below.

$$X_1 = \{(x, w): x_1 + 2w_{12} \geq 0, x_2 + 2w_{12} \geq 0, x_1 + 2x_2 - 2w_{12} \geq 1, 2x_1 + x_2 - 2w_{12} \geq 1, \\ w_{12} \geq 0, x_1 - w_{12} \geq 0, x_2 - w_{12} \geq 0, -x_1 - x_2 + w_{12} \geq -1\}.$$

Note that the constraints of X_0 are obtainable from X_1 by surrogating appropriate constraints. For example, summing constraints 1 and 3 gives $2x_1 + 2x_2 \geq 1$, summing constraints 5 and 6 gives $x_1 \geq 0$, and summing constraints 7 and 8 gives $x_1 \leq 1$. Moreover, note that the point $x = (\frac{1}{2}, 0) \in X_0$, but putting $x_2 = 0$ in X_1 implies that $w_{12} = 0$ from constraints 5 and 7, and so $x_1 = 1$ from constraints 3 and 8. Hence, $(\frac{1}{2}, 0)$ is infeasible to X_{P_1} , and so, X_0 strictly contains X_{P_1} . However, the solution $\{x = (\frac{1}{3}, \frac{1}{3}), w_{12} = 0\} \in X_1$ while $x = (\frac{1}{3}, \frac{1}{3})$ is infeasible to $\text{conv}(X)$. Therefore, X_{P_1} strictly contains $\text{conv}(X)$.

Finally, consider X_2 . Using the factors $x_1x_2, x_1(1 - x_2), (1 - x_1)x_2$ and $(1 - x_1) \times (1 - x_2)$ of degree 2 on the constraint $2x_1 + 2x_2 \geq 1$ gives $3x_1x_2 \geq 0, x_1(1 - x_2) \geq 0, x_2(1 - x_1) \geq 0$, and $(-1)(1 - x_1)(1 - x_2) \geq 0$, respectively. Hence, $E = \{\phi\}$ in (3.7), and so from (3.8), we get $X_2 = \{(x, w): w_{12} \geq 0, x_1 - w_{12} \geq 0, x_2 - w_{12} \geq 0, w_{12} = x_1 + x_2 - 1\}$. Substituting the fourth constraint for w_{12} into the first three constraints, we obtain $X_{P_2} = \text{conv}(X)$ as in Theorem 3. Hence for this example, $X_0 \supset X_{P_1} \supset X_{P_2} = \text{conv}(X)$.

We have established, via Theorems 1 and 3, that $X_0 \supseteq X_{P_1} \supseteq \dots \supseteq X_{P_n} = \text{conv}(X)$. Observe that one can (consider) the process of constructing this hierarchy of tighter relaxations to be in the spirit of Chvatal (1973) as a sequential process in which the factors x_j and $(1 - x_j)$ for $j = 1, \dots, n$ are used to multiply the constraints of X_{d-1} ,

which are then linearized using $w_J \cdot x_k = w_{J \cup k}$ for all $J \subseteq N$, in order to produce X_d for $d = 1, \dots, n$. (This sequential technique produces certain additional implied constraints in X_d , which can be avoided by prohibiting the use of the factor x_j or $(1 - x_j)$ on a constraint which has previously been treated by either of these factors.)

Another useful fact to note at this point is an alternative representation of Z_n and X_n , given by (3.5a) and (3.8), respectively, in the (x, y) -variable space under the transformation (3.1). Note that under (3.1) Z_n of (3.5a) gets transformed into S_n given by (3.5b) in the y -space which can be written equivalently in the (x, y) space using (3.1b) as

$$(3.9) \quad Z_n(x, y) = \left\{ (x, y) : x_j = \sum_{J \subseteq N-j} y_{J \cup J} \text{ for } j = 1, \dots, n \right. \\ \left. \sum_{\substack{J \subseteq N \\ J \neq \phi}} y_J + y_0 = 1, y_0 \geq 0 \text{ and } y_J \geq 0 \text{ for all } J \subseteq N, J \neq \phi \right\}$$

where we have introduced a slack variable y_0 in the final constraint of Z_n . Accordingly, X_n of (3.8) can be written as

$$(3.10a) \quad X_n(x, y) = \left\{ (x, y) : x_j = \sum_{J \subseteq N-j} y_{J \cup J} \text{ for } j = 1, \dots, n \right. \\ \sum_{\substack{J \subseteq N \\ J \neq \phi}} y_J + y_0 = 1 \\ y_0 \geq 0 \text{ with } y_0 = 0 \text{ if } \{\phi\} \in E \\ \left. y_J \geq 0 \text{ with } y_J = 0 \text{ if } J \in E \text{ for all } J \subseteq N, J \neq \phi \right\}.$$

This means that the projected set X_{Pn} can equivalently be written as

$$(3.10b) \quad X_{Pn} = \{x : (x, y) \in X_n(x, y)\}.$$

Observe that the representation of $X_{Pn} = \text{conv}(X)$ as given by (3.10) has $(n + 1)$ constraints in (at most) $n + 2^n$ nonzero variables, and is essentially in the spirit of writing x as a convex combination of all feasible binary solutions in X , where $y_0 = 1$ represents $x_j = 0$ for all $j = 1, \dots, n$, and $y_J = 1$ for a set $J \subseteq N, J \neq \phi$, represents $x_j = 1$ for all $j \in J$ and $x_j = 0$ for all $j \in \bar{J}$.

A third point to note is that the set $X_{Pn} \equiv \text{conv}(X)$ can be characterized in the x -space itself via its projection from the (x, w) or the (x, y) space. For example, consider its representation in the (x, y) space as given by (3.10) and, for the sake of convenience, let us write it in the following form with obvious notation:

$$(3.11) \quad X_{Pn} = \{x : \text{there exists a } y \geq 0 \text{ such that } Ay = x \text{ and } e \cdot y = 1\}$$

where the vector y is comprised of components corresponding to the set $\bar{E}$ (i.e., the zero components have been eliminated), and where e is an appropriate vector of ones. We assume here that $\bar{E} \neq \phi$, so that $X_{Pn} \neq \phi$. Now, $x \in X_{Pn}$ if and only if $0 = \min \{0 \cdot y : Ay = x, e \cdot y = 1, y \geq 0\}$, i.e., $0 = \max \{\pi x + \pi_0 : \pi A + e \pi_0 \leq 0\}$, where

$\pi = (\pi_1, \dots, \pi_n)$ and π_0 are Lagrange multipliers associated with the constraints $Ay = x$ and $e \cdot y = 1$, respectively. Define the set

$$(3.12) \quad P = \{(\pi, \pi_0) : \pi A + e\pi_0 \leq 0\},$$

and note that P is clearly unbounded. Now, consider the following result.

THEOREM 4. *Let X_{P_n} and P be defined by (3.11) and (3.12), respectively. Then X_{P_n} is full dimensional in R^n if and only if the origin is a vertex of P . In this case, P has extreme directions (π^k, π_0^k) , $k = 1, \dots, K$, where $\pi_0^k = 0, +1$, or -1 , and*

$$(3.13) \quad X_{P_n} = \{x : \pi^k x + \pi_0^k \leq 0 \text{ for } k = 1, \dots, K\}.$$

Moreover, each defining inequality in (3.13) is a facet of X_{P_n} .

Proof. First, note that the convex set X_{P_n} is not full dimensional if and only if there exists a (π, π_0) , $\pi \neq 0$, such that $\pi x + \pi_0 = 0$ for all $x \in X_{P_n}$. Now, since $X_{P_n} = \text{conv}(X)$ given by (3.11) is the set of convex combinations of the columns A^i of A which correspond to (binary) feasible solutions in X , it follows that X_{P_n} is not full dimensional if and only if $\pi A + e\pi_0 = 0$ for some $\pi \neq 0$. Consequently, X_{P_n} is full dimensional in R^n if and only if $\pi A + e\pi_0 = 0$ implies (π, π_0) is zero, which happens if and only if there exist $(n+1)$ linearly independent hyperplanes defining P , i.e., the origin is a vertex of P . Hence, in this case, P is a polyhedral cone with a single extreme point at the origin, and with extreme directions or generators (π^k, π_0^k) for $k = 1, \dots, K$, where, without loss of generality, we may scale π_0^k to be 0, +1, or -1. Furthermore, since $x \in X_{P_n}$ if and only if $0 = \max \{ \pi x + \pi_0 : (\pi, \pi_0) \in P \}$, i.e., $\pi^k x + \pi_0^k \leq 0$ for all $k = 1, \dots, K$, we get (3.13).

Finally, let us show that if X_{P_n} is full dimensional, then any defining inequality $\pi^h x + \pi_0^h \leq 0$ in (3.13), where $h \in \{1, \dots, K\}$, is a facet of X_{P_n} . Since (π^h, π_0^h) is an extreme direction of P , there exist n linearly independent constraints $\pi A^i + \pi_0 \leq 0$ of P , indexed by $i \in H$, which are binding at (π^h, π_0^h) . But this means that A^i , $i \in H$, are n affinely independent points lying on the hyperplane $\pi^h x + \pi_0^h = 0$, and since $A^i \in X$ for all $i \in H$, this implies that $\pi^h x + \pi_0^h \leq 0$ is a facetial inequality, and the proof is complete. $\square$

Note that if $X_{P_n} = \text{conv}(X)$ is not full dimensional, we can still assert that $x \in X_{P_n}$ if and only if $0 = \max \{ \pi x + \pi_0 : \pi A + e\pi_0 \leq 0 \}$, which holds if and only if $0 = \max \{ (\pi_1 - \pi_2)x + (\pi_{01} - \pi_{02}) : (\pi_1 - \pi_2)A + e(\pi_{01} - \pi_{02}) \leq 0, \pi_1 \geq 0, \pi_2 \geq 0, \pi_{01} \geq 0, \pi_{02} \geq 0 \}$. Hence, whether or not X_{P_n} is full dimensional, we have the following results.

COROLLARY 3. *The projection of the set $X_{P_n} \neq \emptyset$ of (3.11) onto the x -space is given in general by (3.13), where $(\pi_1^k, \pi_2^k, \pi_{01}^k, \pi_{02}^k)$, $k = 1, \dots, K$ are the extreme directions of the set $\{(\pi_1, \pi_2, \pi_{01}, \pi_{02}) \geq 0 : (\pi_1 - \pi_2)A + e(\pi_{01} - \pi_{02}) \leq 0\}$, and where $\pi^k = (\pi_1^k - \pi_2^k)$, $\pi_0^k = \pi_{01}^k - \pi_{02}^k$ for each $k = 1, \dots, K$.*

COROLLARY 4. *The inequality $\pi x + \pi_0 \leq 0$ is valid for X_{P_n} if and only if it is implied by a constraint obtained by surrogating the constraints $Ay = x$, $e \cdot y = 1$ in (3.11) using multipliers (π, π_0) feasible to the dual set P defined by (3.12).*

Example 3. Let $X = \{x \in R^3 : \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3 \geq 1, x \text{ binary}\}$, where $\alpha_1 \geq 1$, $0 \leq \alpha_2 < 1$, $0 \leq \alpha_3 < 1$, and $\alpha_2 + \alpha_3 \geq 1$. Hence, from (3.7), the set $E = \{\{\phi\}, \{2\}, \text{ and } \{3\}\}$, and so from (3.10), noting accordingly that $y_0 = y_2 = y_3 = 0$, we have $X_{P_n} = \{x : y_1 + y_{12} + y_{13} + y_{123} = x_1, y_{12} + y_{23} + y_{123} = x_2, y_{13} + y_{23} + y_{123} = x_3, y_1 + y_{12} + y_{13} + y_{23} + y_{123} = 1, \text{ and } y \geq 0\}$. Denoting π_1, π_2, π_3 and π_0 as the Lagrange multipliers

with respect to the four equality constraints in X_{P_n} , we obtain the set P in (3.12) as

$$P = \{(\pi, \pi_0) : \pi_1 + \pi_0 \leq 0, \pi_1 + \pi_2 + \pi_0 \leq 0, \pi_1 + \pi_3 + \pi_0 \leq 0, \pi_2 + \pi_3 + \pi_0 \leq 0, \\ \text{and } \pi_1 + \pi_2 + \pi_3 + \pi_0 \leq 0\}.$$

Note that the origin is a vertex of P and so X_{P_n} is full dimensional by Theorem 4. Hence, taking three linearly independent constraints of P to be binding at a time, and accordingly solving for three variables uniquely in terms of the fourth, and considering the resulting solutions with $\pi_0 = 0, 1$, or -1 which are feasible to the remaining constraints, we obtain the extreme directions of P as $(\pi_1, \pi_2, \pi_3, \pi_0) = (+1, 0, 0, -1), (0, +1, 0, -1), (0, 0, +1, -1), (-1, 0, -1, +1)$ and $(-1, -1, 0, +1)$. Hence, $K = 5$ and $X_{P_n} \equiv \{x : x_1 \leq 1, x_2 \leq 1, x_3 \leq 1, x_1 + x_3 \geq 1, \text{ and } x_1 + x_2 \geq 1\}$, all of which are facetial inequalities as stated in Theorem 4.

Before progressing into § 4 on extensions to zero-one polynomial programming problems, we mention here that the polyhedral set X_d for $0 < d < n$ defined by (2.3a), (2.3b), and (2.3c) can be equivalently represented using fewer equations when the original set X contains equality constraints, that is, when $R_2 > 0$ in (2.1). To accomplish this, observe that for any d with $0 \leq d \leq n$ and any (J_1, J_2) of order d , $f_d(J_1, J_2)$ can be expressed as the addition and subtraction of the terms $w_{J_1 \cup J} \equiv f_{|J_1 \cup J|}(J_1 \cup J, \phi)$ for $J \subseteq J_2$. More precisely, it can be easily verified that for any (J_1, J_2) of order d , we have

$$f_d(J_1, J_2) = \sum_{p=|J_1|}^d \sum_{\substack{J \subseteq J_2 \\ |J|=p-|J_1|}} (-1)^{|J|} f_p(J_1 \cup J, \phi),$$

where $f_0(\phi, \phi) = 1$. Consequently, for any fixed d with $0 \leq d \leq n$, and for each $r = 1, \dots, R_2$, the equations

$$(3.14) \quad \left(\sum_{j \in J_1} a_{rj} - b_r \right) f_d(J_1, J_2) + \sum_{j \in N - (J_1 \cup J_2)} a_{rj} f_{d+1}(J_1 \cup j, J_2) = 0$$

for all (J_1, J_2) of order d

of (2.3b) can be equivalently replaced in X_d by

$$(3.15) \quad \left(\sum_{j \in J} a_{rj} - b_r \right) f_p(J, \phi) + \sum_{j \in N - J} a_{rj} f_{p+1}(J \cup j, \phi) = 0$$

for all $J \subseteq N$ with $|J| = p$, for $p = 0, \dots, d$

as shown in the following lemma. Since (3.14) consists of $\binom{n}{d} 2^d$ equations and (3.15) consists of $\sum_{i=0}^d \binom{n}{i}$ equations, a net savings of $R_2[2^d \binom{n}{d} - \sum_{i=0}^d \binom{n}{i}]$ equations results.

LEMMA 5. *Consider the set X_d for any $0 \leq d \leq n$. Then for each $r = 1, \dots, R_2$, the equations (3.14) in X_d are implied by the (valid) equations (3.15).*

Proof. For any (J_1, J_2) of order d , we can write the left-hand side of (3.14) as

$$\sum_{p=|J_1|}^d \sum_{\substack{J \subseteq J_2 \\ |J|=p-|J_1|}} (-1)^{|J|} \left[\left(\sum_{j \in J_1} a_{rj} - b_r \right) f_p(J_1 \cup J, \phi) \right. \\ \left. + \sum_{j \in N - (J_1 \cup J_2)} a_{rj} f_{p+1}(J_1 \cup J \cup j, \phi) \right].$$

But the term in $[\cdot]$ above equals

$$\left[\left(\sum_{j \in J_1 \cup J} a_{rj} - b_r \right) f_p(J_1 \cup J, \phi) + \sum_{j \in N - (J_1 \cup J_2)} a_{rj} f_{p+1}(J_1 \cup J \cup j, \phi) \right] \\ - \left[\sum_{j \in J} a_{rj} f_p(J_1 \cup J, \phi) + \sum_{j \in J_2 - J} a_{rj} f_{p+1}(J_1 \cup J \cup j, \phi) \right].$$

The first of these two terms in $[\cdot]$ is zero by (3.15). Hence, the left-hand side of (3.14) equals

$$- \sum_{p=|J_1|}^d \sum_{\substack{J \subseteq J_2 \\ |J| = p - |J_1|}} (-1)^{|J|} \left[\sum_{j \in J} a_{rj} f_p(J_1 \cup J, \phi) + \sum_{j \in J_2 - J} a_{rj} f_{p+1}(J_1 \cup J \cup j, \phi) \right] \\ = - \sum_{j \in J_2} a_{rj} \left[\sum_{\substack{J \subseteq J_2 \\ j \in J}} w_{J_1 \cup J} (-1)^{|J|} + \sum_{J \subseteq J_2 - j} (-1)^{|J|} w_{J_1 \cup J \cup j} \right] \\ = - \sum_{j \in J_2} a_{rj} \left[\sum_{J \subseteq J_2 - j} w_{J_1 \cup J \cup j} \{ (-1)^{|J|+1} + (-1)^{|J|} \} \right] = 0.$$

This completes the proof. $\square$

4. Extensions to zero-one polynomial programming problems. Let us now demonstrate that the foregoing results are generally applicable to zero-one polynomial programming optimization problems of the following form:

$$\text{PP: minimize } \sum_{t \in T_0} \alpha_{0t} p(J_{1t}^0, J_{2t}^0) \\ \text{subject to } x \in X = \{x: \sum_{t \in T_r} \alpha_{rt} p(J'_{1t}, J'_{2t}) \geq \beta_r \text{ for } r = 1, \dots, R_1 \\ (4.1) \quad \sum_{t \in T_r} \alpha_{rt} p(J'_{1t}, J'_{2t}) = \beta_r \text{ for } r = R_1 + 1, \dots, R_1 + R_2 \\ 0 \leq x \leq e, x \text{ integer} \}$$

where $p(J_1, J_2) \equiv (\pi_{j \in J_1} x_j)(\pi_{j \in J_2} (1 - x_j))$ are polynomial terms for $(J_1, J_2) = (J'_{1t}, J'_{2t})$, $t \in T_r$, $r = 0, \dots, R_1 + R_2$. Note that this is a generalization of the linear zero-one program for which we have $T_r = \{1, \dots, n\}$, $J'_{1t} = \{t\}$ and $J'_{2t} = \{\phi\}$ for all $t \in T_r$, $r = 0, \dots, R_1 + R_2$. Now, let δ denote the maximum degree over all polynomial terms in PP (including those in the objective function). Observe that $\delta = 1$ for the linear case. The hierarchy of relaxations X_d , $d = 1, \dots, n$ for problem PP is obtained by constructing X_d as follows for any $d \in \{1, \dots, n\}$.

Step 1. Multiply each of the $R_1 + R_2$ constraints in (4.1) by each factor $F_d(J_1, J_2)$ of degree d , using the relationships $x_j^2 = x_j$ and $x_j(1 - x_j) = 0$ for $j = 1, \dots, n$. Include the constraints $F_D(J_1, J_2) \geq 0$ for all (J_1, J_2) of order $D = \min\{d + \delta, n\}$.

Step 2. Linearize the resulting polynomial constraint expressions by substituting w_J for $\pi_{j \in J} x_j$ for all sets J such that $|J| \geq 2$. Let this resulting polyhedron in the (x, w) space be denoted by X_d as before. (For uniformity, let X_0 be the set of x -variables such that (x, w) is feasible to the constraints obtained via the above linearization process applied using $d = 0$. Note that for the linear case, this is the usual linear programming relaxation as before.)

Now, let us generalize the results of §§ 2 and 3. (Throughout, $w_J \equiv x_J$ whenever $|J| = 1$.) First of all, Lemmas 1 and 2 remain unaffected. Next, consider Theorem 1 and define X_{Pd} as in (2.8) for $d = 1, \dots, n$. Then as in (2.9), we again obtain the hierarchy of relaxations $X_0 \supseteq X_{P1} \supseteq X_{P2} \supseteq \dots \supseteq X_{Pn} \supseteq \text{conv}(X)$, with the proof of Theorem 1 repeating the same arguments based on the following observation. Consider any $d \in \{1, \dots, n\}$, let (J_1, J_2) be of order $(d-1)$, denote $J_3 = N - J_1 \cup J_2$, and let $k \in J_3$. Note that by virtue of (2.4), if $p(J'_1, J'_2)$ is any polynomial term, then

$$p(J'_1, J'_2)[F_d(J_1, J_2 + k) + F_d(J_1 + k, J_2)] = p(J'_1, J'_2)[F_{d-1}(J_1, J_2)].$$

Hence, any constraint in X_{d-1} constructed by multiplying some constraint in PP by $F_{d-1}(J_1, J_2)$ can be obtained as before by summing the corresponding constraints in X_d constructed by multiplying this same constraint by $F_d(J_1, J_2 + k)$ and by $F_d(J_1 + k, J_2)$ for any $k \in J_3$. Therefore the proof of Theorem 1 applies here as well.

Corollary 1 also follows since if $(\hat{x}, \hat{w})$ is any binary solution feasible to X_d for $d \in \{1, \dots, n\}$, then by Theorem 1, $(\hat{x}, \hat{w})$ satisfies the constraints in X_0 , and since $\hat{w}_J = \pi_{j \in J} \hat{x}_j$ for all sets J such that $|J| = 1, \dots, D = \min\{d + \delta, n\}$ by Lemma 2, this means that $\hat{x} \in X$. The converse statement of Corollary 1 follows by construction.

Lemmas 3 and 4, Theorem 2, and Corollary 2 remain unaffected and hold as before.

Next, to establish Theorem 3, consider any constraint $\sum_{t \in T_r} \alpha_{rt} p(J'_{1t}, J'_{2t}) \geq \beta_r$ in problem PP for some $r \in \{1, \dots, R_1\}$, and suppose that this constraint is multiplied by a factor $F_n(J, \bar{J})$ in constructing X_n , where $J \subseteq N$ and $\bar{J} = N - J$. Then, for any polynomial term $p(J'_{1t}, J'_{2t})$, the term $p(J'_{1t}, J'_{2t}) \cdot F_n(J, \bar{J}) \equiv F_n(J, \bar{J})$ provided $J'_{1t} \subseteq J$ and $J'_{2t} \subseteq \bar{J}$, and is zero otherwise. Therefore, denoting $T_r(J) = \{t \in T_r: J'_{1t} \subseteq J \text{ and } J'_{2t} \subseteq \bar{J}\}$ for all $J \subseteq N$ and $r = 1, \dots, R_1 + R_2$, we get

$$\begin{aligned} X_n = & \left\{ (x, w): \text{for each } J \subseteq N, \bar{J} = N - J, \text{ we have,} \right. \\ & \left[\sum_{t \in T_r(J)} \alpha_{rt} - \beta_r \right] f_n(J, \bar{J}) \geq 0 \text{ for } r = 1, \dots, R_1 \\ & \left[\sum_{t \in T_r(J)} \alpha_{rt} - \beta_r \right] f_n(J, \bar{J}) = 0 \text{ for } r = R_1 + 1, \dots, R_1 + R_2 \\ & \left. f_n(J, \bar{J}) \geq 0 \right\}. \end{aligned} \quad (4.2)$$

Now, let us define

$$\begin{aligned} E = & \left\{ J \subseteq N: \sum_{t \in T_r(J)} \alpha_{rt} - \beta_r < 0 \text{ for some } r \in \{1, \dots, R_1\} \text{ or} \right. \\ & \left. \sum_{t \in T_r(J)} \alpha_{rt} - \beta_r \neq 0 \text{ for some } r \in \{R_1 + 1, \dots, R_1 + R_2\} \right\} \end{aligned} \quad (4.3)$$

and let $\bar{E}$ be the complement of E with respect to the power set of N . Then X_n given by (4.2) is precisely of the form in (3.8). Therefore, the proof of Theorem 3 now follows identically, and so does the ensuing discussion in the foregoing section, including Theorem 4 and Corollaries 3 and 4.

Hence, concerning the solution of the zero-one polynomial programming problem PP, we have by Lemma 2 and Corollary 1 that for any $d \in \{0, 1, \dots, n\}$, Problem PP

is equivalent to the linear mixed-integer zero-one programming problem over the feasible region $X_d \cap \{x: x \text{ binary}\}$, where the objective function is linearized by substituting $w_J = \pi_{j \in J} x_j$ for all sets J such that $|J| \geq 2$. Moreover, by Theorems 1 and 3, the corresponding linear programming relaxation over X_d provides a hierarchy of tighter relaxations for increasing values of $d = 0, 1, \dots, n$, with the final relaxation representing the convex hull of X .

5. Illustrations for deriving implied inequalities. In § 3, we characterized the convex hull of feasible solutions X_{Pn} to X via X_n given by (3.8) in the (x, w) space, or via $X_n(x, y)$ given by (3.10) in the (x, y) space, where E is defined in (3.7). We also saw using the representation (3.10), for example, that valid inequalities can be derived by surrogating the constraints of $X_n(x, y)$ using dual feasible solutions to this problem defined by the set P in (3.12) (see Corollary 4). We demonstrate in this section that although X_n has a combinatorial structure, this structure is special enough to permit us to use it implicitly in deriving valid inequalities. This is particularly true if the original combinatorial optimization problem possesses some special identifiable structure.

As a simple example, consider the problem: maximize $\{x_1: 2x_1 + 2x_2 + \dots + 2x_n = n, x \text{ binary}\}$, where $n \geq 3$ and odd, which was shown by Jeroslow (1974) to require an exponential effort via branch and bound. This problem is trivially infeasible, and so the convex hull of feasible solutions is empty. We readily recognize this fact since E in (3.7) contains all the subsets of N and so $y_0 = y_J = 0$ for all $J \subseteq N$, contradicting that the y -variables must sum to unity in (3.10a). We note that the set $X_1 \neq \emptyset$ (since $x_i = \frac{1}{2}$ for all i , and $w_{ij} = (n-2)/4(n-1)$ for all (i, j) , $i < j$, is feasible to X_1), but that $X_d = \emptyset$ for $d \geq (n+1)/2$. This follows since for any d such that $(n+1)/2 \leq d \leq n$, we obtain the same contradiction as above that $w_J = 0$ for all sets J such that $|J| \geq 2$, implying that $x_j = 0$ for all $j = 1, \dots, n$. To illustrate for $n = 3$, the problem over X_1 is given below, where we have used the abbreviated form of X_1 as explained at the end of § 3.

$$(5.1) \quad \begin{aligned} &\text{Maximize } \{x_1: 2x_1 + 2x_2 + 2x_3 = 3, x_1 = 2w_{12} + 2w_{13}, x_2 = 2w_{12} + 2w_{23}, \\ &\quad x_3 = 2w_{23} + 2w_{13}, \text{ and (2.3c) with } D = 2, \text{ i.e., } [w_{ij} \leq x_i, w_{ij} \leq x_j, \\ &\quad w_{ij} \geq x_i + x_j - 1, \text{ and } w_{ij} \geq 0] \text{ for } i, j = 1, 2, 3, i < j\}. \end{aligned}$$

Note that as mentioned above, the solution $x_1 = x_2 = x_3 = \frac{1}{2}$ and $w_{12} = w_{13} = w_{23} = \frac{1}{8}$ is feasible, and so $X_1 \neq \text{conv}(X)$. However, if x_1 is fixed at unity, then (2.3c) implies that $w_{12} = x_2$ and $w_{13} = x_3$. This, together with the third and fourth constraints in (5.1) and the nonnegativity of the variables, implies that $x_2 = x_3 = 0$, which is inconsistent with the first constraint. Hence, the program (5.1) does tighten the continuous relaxation of the problem over X_0 . However, if we multiply the original equality constraint by $x_1 x_2$, $x_1 x_3$ and $x_2 x_3$ also, and denote the new product term $x_1 x_2 x_3$ by w_{123} , we get $w_{12} + 2w_{123} = 0$, $w_{13} + 2w_{123} = 0$ and $w_{23} + 2w_{123} = 0$. Since $w \geq 0$, this gives $w_{12} = w_{13} = w_{23} = 0$ which from the second, third, and fourth constraints in (5.1), implies that $x_1 = x_2 = x_3 = 0$, an inconsistency with respect to the first constraint in (5.1). Hence, $X_2 = X_3 = \text{conv}(X) = \emptyset$.

As a somewhat more involved example, consider the edge covering problem defined on an undirected, connected graph $G(V, A)$, which seeks a minimum weighted set of edges such that for each vertex $v \in V$, at least one incident edge is chosen. Hence for this problem, defining $x_i = 1$ if edge $i \in A$ is chosen, and zero otherwise, we have

$$(5.2) \quad X = \left\{ x: \sum_{i \in I(v)} x_i \geq 1 \text{ for each } v \in V, 0 \leq x_i \leq 1, x_i \text{ integer} \right\}$$

where $I(v)$ is the set of edges in A incident at vertex v , for each $v \in V$. Edmonds (1965) has shown that for any subset $S \subseteq V$ such that $|S|$ is odd, the following inequality is valid, and in fact, the set of all such inequalities along with X_0 define $\text{conv}(X)$.

$$(5.3) \quad \sum_{i \in \{\cup I(v): v \in S\}} x_i \geq \frac{|S| + 1}{2}.$$

Observe that (5.3) may be derived through Chvatal's (1973) construction process by surrogating the constraints in (5.2) corresponding to $v \in S$ using weights of $\frac{1}{2}$, and then rounding up the resulting coefficients.

The inequality (5.3) can also be derived through our construction of X_n in (3.10) where E is defined by (3.7). To see this, consider any subset $S \subseteq V$ with $|S| = s$ odd, denote $I = \{\cup I(v): v \in S\}$ and $|I| = q$, and let $s^* = (s + 1)/2$. Since (5.3) coincides with the constraints in (5.2) for $s = 1$, assume that $s \geq 3$ so that $q \geq 2$.

Now, for any $v \in V$, the constraint $\sum_{i \in I(v)} x_i \geq 1$ in (5.2) asserts that

$$(5.4) \quad y_J = 0 \quad \text{for all } J \subseteq \overline{I(v)},$$

since any such J belongs to E , where $\overline{I(v)} = N - I(v)$ is the complement of $I(v)$ with respect to the index set N for the arcs in the graph. In particular $\{\phi\} \in E$ and so $y_0 = 0$.

Next, consider any $J \subseteq N$ such that $|J \cap I| \leq (s^* - 1) = (s - 1)/2$. Hence, the arcs in J can involve at most $2|J \cap I| = (s - 1)$ vertices from S , and so for some vertex $v \in S$, we must have $J \subseteq \overline{I(v)}$. Therefore from (5.4), we conclude that

$$(5.5) \quad y_J = 0 \quad \text{for all } J \text{ such that } |J \cap I| \leq (s^* - 1).$$

This means that the $e \cdot y = 1$ constraint in (3.10a) may be written as

$$(5.6) \quad \sum_{J: |J \cap I| \geq s^*} y_J = 1.$$

Now, from the first n constraints in (3.10a), we have,

$$(5.7) \quad \sum_{i \in I} x_i = \sum_{i \in I} \sum_{J \subseteq N - i} y_{i \cup J} = \sum_{J: |J \cap I| = 1} y_J + 2 \sum_{J: |J \cap I| = 2} y_J + \cdots + q \sum_{J: |J \cap I| = q} y_J.$$

Hence, using (5.5) and (5.6) in (5.7), we obtain

$$(5.8) \quad \sum_{i \in I} x_i = \sum_{k=s^*}^q k \left[\sum_{J: |J \cap I| = k} y_J \right] \geq s^* \sum_{k=s^*}^q \sum_{J: |J \cap I| = k} y_J = s^*$$

which gives the required inequality (5.3).

There are two points worth noting here. Firstly, observe that (5.8) is derived through surrogating the $(n + 1)$ equations in (3.10a) using multipliers $\pi_i = -1$ for $i \in I$, $\pi_i = 0$ for $i \in N - I$ and $\pi_0 = s^*$ (or equivalently, $\pi_i = -1/s^*$ for $i \in I$, $\pi_i = 0$ for $i \in N - I$ and $\pi_0 = 1$), where $(\pi, \pi_0) \in P$ of (3.12), as evident by (5.8) using (5.6) and (5.7). Secondly, and more importantly, note that (5.7) is a general expression applicable to any problem and may be used to derive appropriate valid inequalities with unit coefficients for any problem. This construct is a consequence of the structure of X_n , which may be exploited to derive other types of inequalities as well for a wide variety of problems.

The third and final example of this section demonstrates that in addition to using X_n to generate the facets of X_{P_n} as shown by Lemmas 3 and 4 and Theorem 4, we can also generate the facets of other polytopes obtained by projecting X_n into different lower

dimensional subspaces of the (x, w) space. Consider, for example, the unconstrained 0-1 quadratic programming problem in three variables of the form

$$(5.9) \quad \text{maximize } c_1x_1 + c_2x_2 + c_3x_3 + q_{12}x_1x_2 + q_{13}x_1x_3 + q_{23}x_2x_3.$$

Because of the presence of quadratic terms, it may be desired to project X_3 into the subspace defined by $(x, w_{II}) \equiv (x_1, x_2, x_3, w_{12}, w_{13}, w_{23})$. Noting that for this problem $X_3 = Z_3$ since $E = \phi$ in (4.3), we obtain

$$\begin{aligned} X_3 = \{ (x_1, x_2, x_3, w_{12}, w_{13}, w_{23}, w_{123}): & -x_1 - x_2 - x_3 + w_{12} + w_{13} + w_{23} - w_{123} \geq -1, \\ & x_1 - w_{12} - w_{13} + w_{123} \geq 0, x_2 - w_{12} - w_{23} + w_{123} \geq 0, x_3 - w_{13} - w_{23} + w_{123} \geq 0, \\ & w_{12} - w_{123} \geq 0, w_{13} - w_{123} \geq 0, w_{23} - w_{123} \geq 0, w_{123} \geq 0 \}. \end{aligned}$$

By Theorem 2, $w_{12} = x_1x_2$, $w_{13} = x_1x_3$, and $w_{23} = x_2x_3$ at all extreme point solutions to X_3 , and so (5.9) is equivalent to maximizing the linear objective function $c_1x_1 + c_2x_2 + c_3x_3 + q_{12}w_{12} + q_{13}w_{13} + q_{23}w_{23}$ over X_3 .

Now, the projected set $X_3(x, w_{II})$, say, is given by

$$(5.10a)$$

$$\begin{aligned} X_3(x, w_{II}) = \{ (x, w_{II}): & -w_{123} \geq -w_{12}, -w_{123} \geq -w_{13}, -w_{123} \geq -w_{23}, \\ & w_{123} \geq -x_1 + w_{12} + w_{13}, w_{123} \geq -x_2 + w_{12} + w_{23}, \\ & w_{123} \geq -x_3 + w_{13} + w_{23}, -w_{123} \geq -1 + x_1 + x_2 + x_3 - w_{12} - w_{13} - w_{23}, \\ & \text{for some } w_{123} \geq 0 \}. \end{aligned}$$

Following the derivation in (3.10)–(3.13), we may write $X_3(x, w_{II})$ in terms of its facet defining inequalities as

$$(5.10b)$$

$$\begin{aligned} X_3(x, w_{II}) = \{ (x, w_{II}): & \pi_1^k(-w_{12}) + \pi_2^k(-w_{13}) + \pi_3^k(-w_{23}) \\ & + \pi_4^k(-x_1 + w_{12} + w_{13}) + \pi_5^k(-x_2 + w_{12} + w_{23}) + \pi_6^k(-x_3 + w_{13} + w_{23}) \\ & + \pi_7^k(-1 + x_1 + x_2 + x_3 - w_{12} - w_{13} - w_{23}) \leq 0, \text{ for } k = 1, \dots, K \} \end{aligned}$$

where $\pi^k \equiv (\pi_1^k, \dots, \pi_7^k)$, $k = 1, \dots, K$ are the extreme directions of the set

$$(5.11) \quad \{ \pi: -\pi_1 - \pi_2 - \pi_3 + \pi_4 + \pi_5 + \pi_6 - \pi_7 \leq 0, \pi \geq 0 \}.$$

This set has 16 extreme directions which are readily obtained as follows, where only the nonzero components are given: $(\pi_1 = 1)$, $(\pi_2 = 1)$, $(\pi_3 = 1)$, $(\pi_7 = 1)$, $(\pi_1 = \pi_4 = 1)$, $(\pi_1 = \pi_5 = 1)$, $(\pi_1 = \pi_6 = 1)$, $(\pi_2 = \pi_4 = 1)$, $(\pi_2 = \pi_5 = 1)$, $(\pi_2 = \pi_6 = 1)$, $(\pi_3 = \pi_4 = 1)$, $(\pi_3 = \pi_5 = 1)$, $(\pi_3 = \pi_6 = 1)$, $(\pi_4 = \pi_7 = 1)$, $(\pi_5 = \pi_7 = 1)$, and $(\pi_6 = \pi_7 = 1)$. Substituting into (5.10b) gives

$$\begin{aligned} X_3(x, w_{II}) = \{ (x, w_{II}): & w_{12} \geq 0, w_{13} \geq 0, w_{23} \geq 0, x_1 + x_2 + x_3 - w_{12} - w_{13} - w_{23} \leq 1, \\ & w_{13} \leq x_1, w_{23} \leq x_2, w_{13} + w_{23} \leq x_3 + w_{12}, w_{12} \leq x_1, \\ (5.12) \quad & w_{12} + w_{23} \leq x_2 + w_{13}, w_{23} \leq x_3, w_{12} + w_{13} \leq x_1 + w_{23}, \\ & w_{12} \leq x_2, w_{13} \leq x_3, x_2 + x_3 - w_{23} \leq 1, x_1 + x_3 - w_{13} \leq 1, \text{ and} \\ & x_1 + x_2 - w_{12} \leq 1. \} \end{aligned}$$

These are precisely the sixteen facets obtained by Padberg (1988) for (5.9).

Alternatively, $X_3(x, w_{II})$ in (5.12) could have been obtained from X_3 more simply by eliminating w_{123} via a Fourier–Motzkin elimination technique, by combining the four inequalities of the type “ $-w_{123} \geq$ ” in (5.10a) with each of the four inequalities of the type “ $w_{123} \leq$ ”.

Note that we could also have followed the derivation in (3.10)–(3.13) more closely by writing $X_3(x, w_{II})$ in the spirit of (3.10) as

$$\begin{aligned} X_3(x, w_{II}) &= \{ (x, w_{II}) : x_1 = y_1 + y_{12} + y_{13} + y_{123}, x_2 = y_2 + y_{12} + y_{23} + y_{123}, \\ &\quad x_3 = y_3 + y_{13} + y_{23} + y_{123}, w_{12} = y_{12} + y_{123}, w_{13} = y_{13} + y_{123}, \\ &\quad w_{23} = y_{23} + y_{123}, y_1 + y_2 + y_3 + y_{12} + y_{13} + y_{23} + y_{123} + y_0 = 1, y \geq 0 \} \\ &= \{ (x, w_{II}) : \text{there exists a } y \geq 0 \text{ such that } Ay = (x, w_{II})^t, \text{ and } e \cdot y = 1 \}, \end{aligned}$$

which is of the form (3.11). Then defining P as in (3.12), we could have obtained a characterization of $X_3(x, w_{II})$ in terms of its facets as in (3.13) of Theorem 4. Note that in this case, the set P of (3.12) is given by

$$\begin{aligned} P &= \{ (\pi, \pi_0) : \pi_1 + \pi_0 \leq 0, \pi_2 + \pi_0 \leq 0, \pi_3 + \pi_0 \leq 0, \pi_1 + \pi_2 + \pi_4 + \pi_0 \leq 0, \\ (5.13) \quad &\pi_1 + \pi_3 + \pi_5 + \pi_0 \leq 0, \pi_2 + \pi_3 + \pi_6 + \pi_0 \leq 0, \\ &\pi_1 + \pi_2 + \pi_3 + \pi_4 + \pi_5 + \pi_6 + \pi_0 \leq 0, \text{ and } \pi_0 \leq 0 \}. \end{aligned}$$

Employing the nonsingular linear transformation $\pi'_4 = -(\pi_1 + \pi_0)$, $\pi'_5 = -(\pi_2 + \pi_0)$, $\pi'_6 = -(\pi_3 + \pi_0)$, $\pi'_1 = -(\pi_1 + \pi_2 + \pi_4 + \pi_0)$, $\pi'_2 = -(\pi_1 + \pi_3 + \pi_5 + \pi_0)$, $\pi'_3 = -(\pi_2 + \pi_3 + \pi_6 + \pi_0)$, and $\pi'_7 = -\pi_0$, which has as its inverse, $\pi_0 = -\pi'_7$, $\pi_1 = (\pi'_7 - \pi'_4)$, $\pi_2 = (\pi'_7 - \pi'_5)$, $\pi_3 = (\pi'_7 - \pi'_6)$, $\pi_4 = (\pi'_4 + \pi'_5) - (\pi'_1 + \pi'_7)$, $\pi_5 = (\pi'_4 + \pi'_6) - (\pi'_2 + \pi'_7)$, $\pi_6 = (\pi'_5 + \pi'_6) - (\pi'_3 + \pi'_7)$, transforms (5.13) into the set $\{ \pi' : -\pi'_1 - \pi'_2 - \pi'_3 + \pi'_4 + \pi'_5 + \pi'_6 - \pi'_7 \leq 0, \pi' \geq 0 \}$ which is of the form (5.11). Using Lemma 4, the extreme directions enumerated for (5.11) may be transformed as above to determine the extreme directions of P in (5.13), and hence characterize $X_3(x, w_{II})$ in terms of its facets as in (3.13). This leads to the same set (5.12) obtained above. Hence, this illustrates how the desired projection can be conducted using the original inequalities of X_n , or under the transformation of Lemma 3, as one finds convenient.

6. Concluding discussions. In this paper we have presented a constructive technique for deriving tight relaxations for zero-one linear (or polynomial) programming problems, which yield a hierarchy ranging from the usual linear programming relaxation to the convex hull of feasible solutions. From a computational viewpoint, even the first order relaxation X_1 can provide an algorithmic advantage beyond the continuous relaxation X_0 , as evident in the results of Adams and Sherali (1986) on zero-one quadratic programming problems. By including partial sets of constraints from tighter relaxations, or by developing methods for generating desirable valid inequalities based on appropriate surrogates of the tighter relaxations, we can enhance the computational performance still further. Note that, as shown in Adams and Sherali (1987) using first degree factors, this strategy for obtaining tighter relaxations is also applicable to linear mixed-integer zero-one programming problems, or to mixed-integer polynomial programming problems which are linear in the continuous variables for fixed values of the binary variables. The construction of a similar hierarchy for these problems as in the present paper, along with a characterization of facets, will be presented in a subsequent paper (see Sherali and Adams (1989)).

From a theoretical viewpoint, as illustrated in the foregoing section, we have the opportunity to exploit the structure of the constructive representation of the convex hull of feasible solutions given by (3.10) to derive valid inequalities. Although establishing that these are members of the set defining (3.13) is a different task altogether, any facial properties of classes of inequalities generated in this fashion may be established by demonstrating the existence of n affinely independent feasible solutions lying on the defining hyperplane, as usual. The principal use of the characterization (3.10) is then to provide a unifying framework for studying known classes of facets for various combinatorial optimization problems, and to provide a mechanism for possibly discovering new classes of facets. This will be explored in future research.

REFERENCES

- W. P. ADAMS AND H. D. SHERALI (1986), *A tight linearization and an algorithm for zero-one quadratic programming problems*, Management Sci., 32, pp. 1274–1290.
- (1987), *Linearization strategies for a class of zero-one mixed integer programming problems*, Oper. Res., to appear.
- E. BALAS (1975), *Facets of the knapsack polytope*, Math. Programming, 8, pp. 146–164.
- (1976), *Facets of one-dimensional and multi-dimensional knapsack polytopes*, Instituto Nazionale di Alta Matematica, Symposia Mathematica, XIX, pp. 11–34.
- (1983), *Disjunctive programming and a hierarchy of relaxations for discrete optimization problems*, MSRS-492, GSIA, Carnegie Mellon University, Pittsburgh, PA 15213. (Also see SIAM J. Algebraic Discrete Methods, 6 (1985), pp. 466–486.)
- (1985), *On the facial structure of scheduling polyhedra*, MSRR-496(R), GSIA, Carnegie Mellon University, Pittsburgh, PA.
- E. BALAS AND E. ZEMEL (1977), *Facets of the knapsack polytope from minimal covers*, SIAM J. Appl. Math., 34, pp. 119–148.
- (1984), *Lifting and complementing yields all the facets of positive zero-one programming polytopes*, in Mathematical Programming, R. W. Cottle, M. L. Kelmanson, and B. Korte, eds., North Holland, Amsterdam, pp. 13–24.
- V. CHVATAL (1973), *Edmonds' polytopes and a hierarchy of combinatorial problems*, Discrete Math., 4, pp. 305–337.
- H. CROWDER, E. L. JOHNSON, AND M. W. PADBERG (1983), *Solving large-scale zero-one linear programming problems*, Oper. Res., 31, pp. 803–834.
- H. CROWDER AND M. W. PADBERG (1980), *Solving large-scale symmetric travelling salesman problems to optimality*, Management Sci., 26, pp. 495–509.
- J. EDMONDS (1965), *Maximum matching and a polyhedron with 0-1 vertices*, Journal Res. Nat. Bur. Standards (B), 69, pp. 125–130.
- G. D. EPPEN AND R. K. MARTIN (1985), *Solving multi-item capacitated lot sizing problems using variable redefinition*, Graduate School of Business, University of Chicago.
- F. GLOVER (1975), *Improved linear integer programming formulations of nonlinear integer problems*, Management Sci., 22, pp. 455–460.
- F. GLOVER AND E. WOOLSEY (1973), *Further reduction of zero-one polynomial programming problems to zero-one linear programming problems*, Oper. Res., 21, pp. 156–161.
- (1974), *Converting the 0-1 polynomial programming problem to a 0-1 linear program*, Oper. Res., 22, pp. 180–182.
- M. GROTSCHTEL AND M. W. PADBERG (1979), *On the symmetric travelling salesman problem II: Lifting theorems and facets*, Math. Programming, 16, pp. 281–302.
- P. L. HAMMER, E. L. JOHNSON, AND U. N. PELED (1975), *Facets of regular 0-1 polytopes*, Math. Programming, 8, pp. 179–206.
- K. HOFFMAN AND M. PADBERG (1985/6), *LP-based combinatorial problem solving*, Annals of OR, 4, pp. 145–194.
- R. G. JEROSLOW (1974), *Trivial integer programs unsolvable by branch-and-bound*, Math. Programming, 6, pp. 105–109.
- (1980), *Representation of unbounded optimizations as integer programs*, J. Optim. Theory Appl., 30, pp. 339–351.

- R. G. JEROSLOW (1983), *Alternative formulations of mixed-integer programs*, College of Industrial Management, Georgia Institute of Technology, Atlanta, GA.
- (1984a), *Representability of functions*, College of Industrial Management, Georgia Institute of Technology, Atlanta, GA.
- (1984b), *Representability in mixed integer programming, I: Characterization results*, College of Industrial Management, Georgia Institute of Technology, Atlanta, GA.
- (1985), *Representability in mixed integer programming, II: A lattice of relaxations*, College of Industrial Management, Georgia Institute of Technology, Atlanta, GA.
- R. G. JEROSLOW AND J. K. LOWE (1984), *Modelling with integer variables*, Math. Programming Stud., 22, pp. 167–184.
- (1985), *Experimental results on the new techniques for integer programming formulations*, Journal of the Operational Research Society, 36, pp. 393–403.
- E. L. JOHNSON, M. M. KOSTREVA, AND U. H. SUHL (1985), *Solving 0-1 integer programming problems arising from large scale planning models*, Oper. Res., 33, pp. 803–819.
- E. L. JOHNSON AND U. H. SUHL (1980), *Experiments in integer programming*, Discrete Appl. Math., 2, pp. 39–55.
- E. L. LAWLER, J. K. LENSTRA, A. H. G. RINNOOY KAN, AND D. SHMOYS (1985), *The Travelling Salesman Problem: A Guided Tour of Combinatorial Optimization*, John Wiley, Chichester.
- K. R. MARTIN (1984), *Generating alternative mixed-integer linear programming models using variable redefinition*, Graduate School of Business, University of Chicago.
- K. MARTIN AND L. SCHRAGE (1982), *Subset coefficient reduction cuts for 0-1 mixed-integer programming*, Graduate School of Business, University of Chicago.
- (1983), *Constraint aggregation and coefficient reduction cuts for mixed-0/1 linear programming*, Graduate School of Business, University of Chicago.
- R. R. MEYER (1975), *Integer and mixed-integer programming models: general properties*, J. Optim. Theory Appl., 16, pp. 191–206.
- (1976), *Mixed-integer minimization models for piecewise-linear functions of a single variable*, Discrete Math., 16, pp. 163–171.
- (1981), *A theoretical and computational comparison of 'equivalent' mixed-integer formulations*, Naval Res. Logist. Quart., 28, pp. 115–131.
- R. R. MEYER, M. V. THAKKAR, AND W. P. HALLMAN (1980), *Rational mixed-integer and polyhedral union minimization models*, Math. Oper. Res., 5, pp. 135–146.
- L. A. OLEY AND R. J. SJOQUIST (1982), *Automatic reformulation of mixed and pure integer models to reduce solution time in Apex IV*, presented at the ORSA/TIMS Fall Meeting, San Diego, CA.
- M. W. PADBERG (1973), *On the facial structure of set packing polyhedra*, Math. Programming, 5, pp. 199–215.
- (1979), *Covering, packing and knapsack problems*, Ann. Discrete Math., 4, pp. 265–287.
- (1980), *(1, k)-configurations and facets for packing problems*, Math. Programming, 18, pp. 94–99.
- M. W. PADBERG (1988), *The Boolean quadric polytope: Some characteristics, facets and relatives*, Working Paper, New York University.
- C. PETERSEN (1971), *A note on transforming the product of variables to linear form in linear programs*, Working Paper, Purdue University, Hammond, IN.
- H. D. SHERALI AND W. P. ADAMS (1989), *A hierarchy of relaxations and convex hull characterizations for mixed-integer zero-one programming problems*, Working Paper, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.
- K. SPIELBERG AND U. SUHL (1980), *An experimental software system for large scale 0-1 problems with efficient data structures and access to MPSX/370*, IBM Research Report RC 8219, White Plains, NY.
- T. J. VAN ROY AND L. A. WOLSEY (1984a), *Solving mixed integer programs by automatic reformulation*, CORE Discussion Paper No. 8432, Center for Operations Research and Econometrics, Universite Catholique de Louvain, Belgium.
- (1983), *Valid inequalities for mixed 0-1 programs*, CORE Discussion Paper No. 8316, Center for Operations Research and Econometrics, Universite Catholique de Louvain, Belgium.
- (1984b), *MPSARX, a mathematical programming system with an automatic reformulation executor*, CORE Computing Report 84-B-01, Center for Operations Research and Econometrics, Universite Catholique de Louvain, Belgium.
- H. P. WILLIAMS (1974), *Experiments in the formulation of integer programming problems*, in Math. Programming Study 2, M. L. Balinski, ed., North Holland, Amsterdam, pp. 180–197.
- (1985), *Model Building in Mathematical Programming*, second ed., Wiley Interscience, New York.
- L. A. WOLSEY (1975), *Faces for a linear inequality in 0-1 variables*, Math. Programming, 8, pp. 165–178.
- (1976), *Facets and strong valid inequalities for integer programs*, Oper. Res., 24, pp. 367–373.
- E. ZEMEL (1978), *Lifting the facets of 0-1 polytopes*, Math. Programming, 15, pp. 268–277.

FINDING CRITICAL INDEPENDENT SETS AND CRITICAL VERTEX SUBSETS ARE POLYNOMIAL PROBLEMS*

CUN-QUAN ZHANG†

Abstract. An independent set J_c of a graph G is called *critical* if

$$|J_c| - |N(J_c)| = \max \{ |J| - |N(J)| : J \text{ is an independent set of } G \},$$

and a vertex subset U_c is called *critical* if

$$|U_c| - |N(U_c)| = \max \{ |U| - |N(U)| : U \text{ is a vertex subset of } G \}.$$

In this paper, it will be shown that finding a critical independent set and a critical vertex subset of a graph are solvable in polynomial time.

Key words. independent set, polynomial algorithm

AMS(MOS) subject classifications. 05C35, 68R10

1. Introduction. It has been proved by mathematicians that finding a maximum independent set in certain kinds of graphs is solvable in polynomial time (for example, line graphs, bipartite graphs, circle graphs, circular arc graphs, and claw free graphs (see [GJ])), but it is well known that it is an NP-complete problem for general graphs (see [GJS]). In this paper, we will investigate another problem—finding a certain kind of independent set in general graphs. An independent set J_c of a graph G is called *critical* if $|J_c| - |N(J_c)|$ is the maximum of $|J| - |N(J)|$ over all independent sets J of G , where $N(J)$ is the set of all vertices of G adjacent to some vertex of J . It will be proved in this paper that finding a critical independent set of a graph is solvable in polynomial time. Let

$$\alpha_c = \max \{ |J| - |N(J)| : J \text{ is an independent set of } G \},$$

which is a parameter of a graph G and is called the *critical independence number* of G . The critical independence number α_c of a graph plays the central role in the study of fractional independence functions and fractional matching functions of graphs [GZ]. (It is proved in [GZ] that the fractional independence number and the fractional matching number of a graph G are $(n - \alpha_c)/2$ and $(n + \alpha_c)/2$, respectively, where $n = |V(G)|$.)

Some related problems and parameters of graphs will also be investigated in this paper. A vertex subset U_c of a graph $G = (V, E)$ is called *critical* if $|U_c| - |N(U_c)|$ is the maximum of $|U| - |N(U)|$ over all vertex subsets U of G . Let

$$\mu_c = \max \{ |U| - |N(U)| : U \text{ is a vertex subset of } G \}$$

which is a parameter of a graph G . Some similar parameters of graphs have been studied by Woodall [WD] and Mohar [MB]. The binding number $b(G)$ [WD] and the isoperimetric number $i(G)$ [MB] of a graph G are defined as the following:

$$b(G) = \min \left\{ \frac{|N(U)|}{|U|} : U \subset V(G), U \neq \emptyset \text{ and } N(U) \neq V(G) \right\};$$

$$i(G) = \min \left\{ \frac{|\partial(U)|}{|U|} : U \subset V(G), U \neq \emptyset \text{ and } |U| \leq \frac{|V(G)|}{2} \right\}$$

* Received by the editors January 16, 1989; accepted for publication (in revised form) September 20, 1989.

† Department of Mathematics, West Virginia University, Morgantown, West Virginia 26506. This research was partially supported by Air Force Office of Scientific Research grant 89-0068.

(where $\partial(U)$ is the number of edges of G with one end vertex in U). Later in this paper, it will be proved that $\alpha_c = \mu_c$.

Since the empty set is an independent set, and the set of all vertices of a graph G is also a vertex subset of G , it is trivial that

$$\alpha_c \geq 0 \quad \text{and} \quad \mu_c \geq 0$$

for any graph G . Note that the empty set and the entire graph are critical vertex subsets of some connected graph G if $\mu_c(G) = 0$. If we are to avoid these two trivial vertex subsets $\emptyset$ and $V(G)$, we may consider the following parameter of a graph G :

$$\mu'_c = \max \{ |U| - |N(U)| : U \subset V(G), U \neq \emptyset \text{ and } U \neq V(G) \}.$$

But for any connected graph G and a vertex v of G , we have that $N(V(G) \setminus \{v\}) = V(G)$, and therefore the parameter $\mu'_c(G)$ still has a lower bound -1 . In order to get more information about graphs, we prefer to consider only those vertex subsets U of a graph G such that $U \neq \emptyset$ and $N(U) \neq V(G)$ which is similar to the definition of the binding number of graphs. A vertex subset U of G is called *proper* if $U \neq \emptyset$ and $N(U) \neq V(G)$. A proper vertex subset U_{pc} of G is called *critical* if $|U_{pc}| - |N(U_{pc})|$ is the maximum of $|U| - |N(U)|$ over all proper vertex subsets U of G . The parameter μ_{pc} of a graph G is defined as the following:

$$\mu_{pc} = \max \{ |U| - |N(U)| : U \subset V(G), U \neq \emptyset \text{ and } U \neq V(G) \}.$$

The problems that will be proved to be solvable in polynomial time are listed as the following.

INSTANCE. Let $G = (V, E)$ be a graph with the vertex set V and the edge set E and k be an integer.

PROBLEM 1. Is there an independent set J of G such that

$$|J| - |N(J)| \geq k?$$

PROBLEM 1*. Find a critical independent set J_c and the critical independence number α_c of G . (That is, to find

$$\alpha_c = |J_c| - |N(J_c)| = \max \{ |J| - |N(J)| : J \text{ is an independent set of } G \}.$$

PROBLEM 2. Is there a vertex set U of G such that

$$|U| - |N(U)| \geq k?$$

PROBLEM 2*. Find a critical vertex subset U_c and the parameter μ_c of G . (That is, to find

$$\mu_c = |U_c| - |N(U_c)| = \max \{ |U| - |N(U)| : U \text{ is a vertex subset of } G \}.$$

PROBLEM 3. Is there a proper vertex subset U of G such that

$$|U| - |N(U)| \geq k?$$

PROBLEM 3*. Find a critical proper vertex subset U_{pc} and the parameter μ_{pc} of G . (That is, to find

$$\mu_{pc} = |U_{pc}| - |N(U_{pc})| = \max \{ |U| - |N(U)| : U \subset V(G), U \neq \emptyset \text{ and } U \neq V(G) \}.$$

2. Main results.

THEOREM 1. Problems 3 and 3* are solvable in polynomial time.

Before we prove Theorem 1 we would like to consider the following problems first. Theorem 1 will be a corollary of Theorem 2.

INSTANCE. Let $G = (V, E)$ be a graph with the vertex set V and the edge set E , $\{u, v\}$ be an ordered pair of nonadjacent vertices of G , and k be an integer.

PROBLEM 4. Is there a vertex subset U of G such that

$$u \in U, v \notin N(U) \text{ and} \\ |U| - |N(U)| \geq k?$$

PROBLEM 4*. Find a vertex subset U_o of G such that

$$|U_o| - |N(U_o)| = \max \{ |U| - |N(U)| : U \subset V(G), u \in U \text{ and } v \notin N(U) \}.$$

The vertex subset U_o found in problem 4* is called (u, v) -critical subset of G .

THEOREM 2. Problems 4 and 4* are solvable in polynomial time.

The following lemmas will be used in the proof of Theorem 2.

LEMMA 3 (Hall's Theorem [HP]). Let $B = (V_1, V_2; E)$ be a bipartite graph. The graph B has a matching covering all vertices of V_2 if and only if $|U| \leq |N(U)|$ for any subset U of V_1 .

LEMMA 4. Let $B = (V_1, V_2; E)$ be a bipartite graph. Assume that there is no matching of B covering all vertices of V_2 . We will have the following conclusions:

- (i) There is a subset U of V_1 such that $|N(U)| < |U|$;
- (ii) Let U_0 be a subset of V_1 such that $|U_0| - |N(U_0)|$ is as great as possible, then there is a matching of the induced bipartite subgraph $(U_0, (N(U_0); E[U_0, N(U_0)]))$ covering all vertices of $N(U_0)$.

Note that if A and B are a pair of disjoint vertex subsets of a graph G , the set of all edges joining A and B is denoted by $E(A, B)$.

Proof. The conclusion of (i) is an immediate corollary of Hall's Theorem.

Let U be a subset of V_1 such that $|U| - |N(U)|$ is as great as possible. Let $B' = (U, N(U); E[U, N(U)])$ be the subgraph of B induced by $U \cup N(U)$. We claim that $|X| \leq |N(X) \cap U|$ for any subset X of $N(U)$. If not, let $X \subseteq N(U)$ such that

$$|X| > |N(X) \cap U|.$$

We will consider the subset $Y = U \setminus N(X)$. Note that

$$N(Y) = N(U \setminus N(X)) \subseteq N(U) \setminus X,$$

and

$$\begin{aligned} |Y| - |N(Y)| &\geq |U \setminus N(X)| - |N(U) \setminus X| = [|U| - |U \cap N(X)|] - [|N(U)| - |X|] \\ &= [|U| - |N(U)|] + [|X| - |U \cap N(X)|] \\ &> |U| - |N(U)|. \end{aligned}$$

This contradicts the choice of U that $|U| - |N(U)|$ is maximum. So by Hall's Theorem, there is a matching in B' which covers all vertices of $N(U)$. $\square$

Proof of Theorem 2. We only prove that Problem 4* is solvable in polynomial time. Let $G = (V, E)$ be a graph with the vertex set $V = \{1, 2, 3, \dots, n\}$ and the edge set E . We will consider the ordered pair of vertices $(1, 2)$ of G and find a $(1, 2)$ -critical vertex subset.

Define a bipartite graph $B = (X, Y; E_B)$ where

$$X = \{x_1, \dots, x_n\},$$

$$Y = \{y_1, \dots, y_n\}, \quad \text{and}$$

$$E_B = \{(x_i, y_j) : (i, j) \text{ is an edge of } G\}.$$

Let V' be a subset of $V(G)$, then the corresponding subsets in X and Y are denoted by $X(V')$ and $Y(V')$, respectively. For example, if $V' = \{i_1, \dots, i_t\}$, then

$$X(V') = X(\{i_1, \dots, i_t\}) = \{x_{i_1}, \dots, x_{i_t}\}$$

and $Y(V') = Y(\{i_1, \dots, i_t\}) = \{y_{i_1}, \dots, y_{i_t}\}$. (Here X and Y can be considered bijections mapping $\{1, 2, \dots, n\}$ onto $\{x_1, \dots, x_n\}$ and $\{y_1, \dots, y_n\}$). If $W = \{x_{i_1}, \dots, x_{i_t}\} \subset X$ (or $W = \{y_{i_1}, \dots, y_{i_t}\} \subset Y$), then the corresponding subset $\{i_1, \dots, i_t\}$ of $V(G)$ is denoted by $X^{-1}(W)$ (or $Y^{-1}(W)$, respectively). The set of all neighbors of a vertex u in B is denoted by $N_B(u)$. If i is a vertex of G , then

$$N_B(x_i) = \{y_j \in Y : (x_i, y_j) \in E_B\} = \{y_j \in Y : (i, j) \in E(G)\} = Y(N(i)).$$

A weight $w: X \cup Y \rightarrow [0, 2]$ is called a $(1, 2)$ -proper weight of B if

$$w(x_1) = 2,$$

$$w(y_2) = 1,$$

$$1 \leq w(x_i) \leq 2 \quad \text{for each vertex } x_i \in X,$$

$$0 \leq w(y_i) \leq 1 \quad \text{for each vertex } y_i \in Y$$

$$\text{and } 0 \leq w(x_i) + w(y_i) \leq 2 \quad \text{for each edge } (x_i, y_i) \in E_B.$$

The total weight $\sum_{u \in X \cup Y} w(u)$ of B is denoted by $w(B)$. A $(1, 2)$ -proper weight w_m of B is called optimum if

$$w_m(B) = \max \{w(B) : w \text{ is a } (1, 2)\text{-proper weight of } B\}.$$

It is obvious that finding an optimum $(1, 2)$ -proper weight of B is a linear programming problem. Hence it is solvable in polynomial time. The purpose of the investigation of an optimum $(1, 2)$ -proper weight w_m of B is to prove that the vertex subset $\{i \in V(G) : w_m(x_i) > 1\}$ is a $(1, 2)$ -critical subset of G .

I. Let w_m be an optimum $(1, 2)$ -proper weight of B and U_o be a $(1, 2)$ -critical vertex subset of G . We claim that

$$(1) \quad w_m(B) \geq 2n + |U_o| - |N(U_o)| = 2n + \alpha_c.$$

Consider the following weight w_1 of B :

$$w_1(x_i) = \begin{cases} 2 & \text{if } i \in U_o \\ 1 & \text{otherwise} \end{cases}$$

$$\text{and } w_1(y_j) = \begin{cases} 0 & \text{if } j \in N(U_o) \\ 1 & \text{otherwise.} \end{cases}$$

It is easy to see that w_1 is a $(1, 2)$ -proper weight of B and

$$w_1(B) = 2|U_o| + (n - |U_o|) + (n - |N(U_o)|) = 2n + |U_o| - |N(U_o)|.$$

By the choice of w_m , we have verified the inequality (1).

II. Let

$$X_b = \{x_i : w_m(x_i) > 1\},$$

$$Y_s = \{y_i : w_m(y_i) < 1\},$$

$$X_{b'} = \{x_i : 1 < w_m(x_i) < 2\}, \quad \text{and}$$

$$Y_{s'} = \{y_i : 0 < w_m(y_i) < 1\}.$$

By the definition of $(1, 2)$ -proper weight, it is obvious that $N_B(X_b) \subseteq Y_s$ and $N_B(Y_{s'}) \subseteq [X \setminus X_b] \cup X_{b'}$.

III. *Case 1.* Suppose there is a matching M in the induced subgraph $B(X_{b'} \cup Y_{s'})$ covering all vertices of $Y_{s'}$. We claim that X_b is a $(1, 2)$ -critical vertex subset in this case. We are to adjust the weight w_m so that the new weight of each vertex in $X_{b'}$ is two and the new weight of each vertex in $Y_{s'}$ is zero. If we can verify that this new weight is optimum, by the inequality (1), it can be shown that $X^{-1}(X_b)$ is a $(1, 2)$ -critical vertex set of G .

If (u, v) is an edge of M , then let $u = M(v)$ and $v = M(u)$. The sets of vertices of $X_{b'}$ covered and not covered by M , are denoted by $M(Y_{s'})$ and $X_{b'} \setminus M$, respectively. Thus $X_{b'} \cup Y_{s'} = Y_{s'} \cup M(Y_{s'}) \cup (X_{b'} \setminus M)$ since M covers all vertices of $Y_{s'}$.

Consider the following weight w_2 of B :

$$w_2(v) = \begin{cases} 2 & \text{if } v \in X_{b'} \\ 0 & \text{if } v \in Y_{s'} \\ w_m(v) & \text{otherwise.} \end{cases}$$

Since any vertex of Y adjacent to a vertex x_i of $X_{b'}$ must be in Y_s , in which the weight of each vertex is zero, w_2 is a $(1, 2)$ -proper weight of B . Note that w_m is optimum and the total weight $w_m(B)$ cannot be less than $w_2(B)$. We claim that w_2 is also an optimum $(1, 2)$ -proper weight of B by proving that $w_m(B) \leq w_2(B)$. Since

$$\{v \in X \cup Y : w_m(v) \neq w_2(v)\} = X_{b'} \cup Y_{s'},$$

we must have that

$$\begin{aligned} w_m(B) - w_2(B) &= \sum_{v \in X_{b'}} [w_m(v) - w_2(v)] + \sum_{v \in Y_{s'}} [w_m(v) - w_2(v)] \\ &= \sum_{v \in Y_{s'}} \{[w_m(v) - w_2(v)] + [w_m(M(v)) - w_2(M(v))]\} \\ &\quad + \sum_{v \in X_{b'} \setminus M} [w_m(v) - w_2(v)]. \end{aligned}$$

Here

$$\begin{aligned} &[w_m(v) - w_2(v)] + [w_m(M(v)) - w_2(M(v))] \\ &= [w_m(v) + w_m(M(v))] - [w_2(v) + w_2(M(v))] \leq 2 - (0 + 2) = 0 \end{aligned}$$

for any $v \in Y_{s'}$, and

$$w_m(v) - w_2(v) < 2 - 2 = 0$$

for any $v \in X_{b'} \setminus M$. This implies that $w_m(B) - w_2(B) \leq 0$. Therefore w_2 is also an optimum $(1, 2)$ -proper weight of B and $w_m(B) = w_2(B)$. The total weight of w_2 is

$$w_2(B) = 2|X_b| + |X \setminus X_b| + |Y \setminus Y_s|.$$

Since $N_B(X_b) \subseteq Y_s$, we must have that

$$\begin{aligned} w_m(B) = w_2(B) &\leq 2|X_b| + (n - |X_b|) + (n - |N(X_b)|) = 2n + |X_b| - |N(X_b)| \\ &= 2n + |X^{-1}(X_b)| - |N(X^{-1}(X_b))| \leq 2n + \alpha_c \text{ (by the definition of } \alpha_c \text{).} \end{aligned}$$

By (1), all equalities hold, and therefore $X^{-1}(X_b) = \{i \in V(G) : w_m(x_i) > 1\}$ is a $(1, 2)$ -critical vertex subset of G .

IV. *Case 2.* If there is no matching of $B(X_{b'} \cup Y_{s'})$ covering all vertices of $Y_{s'}$, by Lemma 4, there is a subset Y_o of $Y_{s'}$ such that: (i) $|Y_o| > |N_B(Y_o) \cap X_{b'}|$ and, (ii) there is a matching M' in the induced bipartite subgraph $B(Y_o \cup [N_B(Y_o) \cap X_{b'}])$ covering all vertices of $N_B(Y_o) \cap X_{b'}$. We are to adjust the weight w_m so that the new weight of each vertex in $Y_o \cup [N_B(Y_o) \cap X_{b'}]$ is 1. We will find that the new weight is greater than w_m . It will contradict that w_m is optimum.

Consider the following weight w_3 :

$$w_3(v) = \begin{cases} 1 & \text{if } v \in Y_o \cup [N_B(Y_o) \cap X_{b'}] \\ w_m(v) & \text{otherwise.} \end{cases}$$

The weight w_3 is $(1, 2)$ -proper since $x_1 \notin N(Y_{s'})$ and any vertex adjacent to a vertex of Y_o must be in $[X \setminus X_b] \cup [N_B(Y_o) \cap X_{b'}]$ in which the weight of each vertex is one. Note that w_m is an optimum $(1, 2)$ -proper weight of B , thus the total weight of w_m cannot be less than the total weight of w_3 . Since

$$\begin{aligned} \{v \in V(B) : w_m(v) \neq w_3(v)\} &\subseteq Y_o \cup [N_B(Y_o) \cap X_{b'}] \\ &= [N_B(Y_o) \cap X_{b'}] \cup M'[N_B(Y_o) \cap X_{b'}] \cup [Y_o \setminus M'], \end{aligned}$$

where $M'[N_B(Y_o) \cap X_{b'}]$ and $[Y_o \setminus M']$ are the sets of vertices of Y_o covered and not covered by M' , respectively, we must have that

$$\begin{aligned} w_m(B) - w_3(B) &= \sum_{v \in N_B(Y_o) \cap X_{b'}} [w_m(v) - w_3(v)] + \sum_{v \in Y_o} [w_m(v) - w_3(v)] \\ &= \sum_{v \in N_B(Y_o) \cap X_{b'}} \{[w_m(v) - w_3(v)] + [w_m(M(v)) - w_3(M(v))]\} \\ &\quad + \sum_{v \in Y_o \setminus M'} [w_m(v) - w_3(v)]. \end{aligned}$$

But

$$\begin{aligned} [w_m(v) - w_3(v)] + [w_m(M(v)) - w_3(M(v))] \\ = [w_m(v) + w_m(M(v))] - [w_3(v) + w_3(M(v))] \leq 2 - (1 + 1) = 0 \end{aligned}$$

for any $v \in N_B(Y_o) \cap X_{b'}$, and

$$w_m(v) - w_3(v) < 1 - 1 = 0$$

for any $v \in Y_o \setminus M'$. Since $|Y_o| > |N_B(Y_o) \cap X_{b'}|$, the set $Y_o \setminus M'$ is not empty and we have that $w_m(B) - w_3(B) < 0$. This contradicts that w_m is optimum and completes the proof of the theorem. $\square$

Proof of Theorem 1. For any ordered pair of nonadjacent vertices $\{u, v\}$ of G , by Theorem 2, we can find a (u, v) -critical subset in polynomial time. The (u, v) -critical subset is denoted by $U_{C(u,v)}$. Then choose a (u_o, v_o) -critical vertex subset $U_{C(u_o, v_o)}$ such that

$$|U_{C(u_o, v_o)}| - |N(U_{C(u_o, v_o)})| = \max \{|U_{C(u,v)}| - |N(U_{C(u,v)})| : (u, v) \text{ are an ordered pair of nonadjacent vertices of } G\}$$

which is a proper critical subset of G desired in Problem 3*. The total cost of finding a critical proper vertex subset is polynomial since the cost of finding a (u, v) -critical vertex subset is polynomial and the number of pairs of nonadjacent vertices in G is at most $\binom{n}{2}$. $\square$

THEOREM 5. *Problems 2 and 2* are solvable in polynomial time.*

Proof. Let $G = (V, E)$ be a graph. Consider a new graph G' by adding two isolated vertices x, y , to G . Let U_c be an (x, y) -critical subset of G' . Obviously, $x, y \in U_c$ and it is clear that $U_c \setminus \{x, y\}$ is a critical subset of G .

Alternating proof of the theorem. Define a bipartite graph $B = (X, Y; E_B)$ where $X = \{x_1, \dots, x_n\}$, $Y = \{y_1, \dots, y_n\}$ and $E_B = \{(x_i, y_j) : (i, j) \text{ is an edge of } G\}$. Assign a weight w to the vertex set of B such that $w: X \cup Y \rightarrow [0, 2]$ and

$$1 \leq w(x_i) \leq 2 \quad \text{for each vertex } x_i \in X,$$

$$0 \leq w(y_i) \leq 1 \quad \text{for each vertex } y_i \in Y$$

$$\text{and } 0 \leq w(x_i) + w(y_i) \leq 2 \quad \text{for each edge } (x_i, y_i) \in E_B.$$

The total weight $\sum_{u \in X \cup Y} w(u)$ is denoted by $w(B)$. Let

$$w_m(B) = \max \{ w(B) : w \text{ is a weight of } B \text{ satisfying the above definition} \}.$$

By an argument similar to the proof of Theorem 2, we can prove that the set $V_b = \{i \in V(G) : w_m(x_i) > 1\}$ is a critical vertex subset of G . $\square$

THEOREM 6. *Problems 1 and 1* are solvable in polynomial time.*

Before the proof of Theorem 6, we will prove a Theorem by which Theorem 6 is an immediate corollary.

THEOREM 7. *Let $G = (V, E)$ be a graph.*

(i) *Let U_c be a critical vertex subset of G and $T_1, \dots, T_h$ be all nontrivial components of the induced subgraph $G(U_c)$. Then $J = V(U_c) \setminus [V(T_1) \cup \dots \cup V(T_h)]$ is a critical independent set of G and $|J| - |N(J)| = |U_c| - |N(U_c)|$.*

(ii) $\max \{|J| - |N(J)| : J \text{ is an independent set of } G\} = \max \{|U| - |N(U)| : U \text{ is a vertex subset of } G\}$. *That is, any critical independent set is also a critical vertex subset of G and therefore $\alpha_c = \mu_c$.*

Proof. It is obvious that

$$(2) \quad 0 \leq \alpha_c = \max \{|J| - |N(J)| : J \text{ is an independent set of } G\}$$

$$\leq \max \{|U| - |N(U)| : U \text{ is a vertex subset of } G\} = \mu_c \dots$$

since any independent set is a vertex subset of G .

Let U_c be a critical vertex subset of G . The theorem is trivial if U_c is an empty set. Assume that U_c is a counterexample to the theorem containing a minimum number of vertices. By this assumption, U_c cannot be an independent set of G . Let T be a nontrivial component of the induced subgraph $G(U_c)$. It is clear that $V(T) \subseteq N(T)$ since T is not a singleton. Thus

$$|U_c \setminus T| - |N(U_c \setminus T)| \geq [|U_c| - |T|] - [|N(U_c)| - |T|] = |U_c| - |N(U_c)|.$$

It implies that $U_c \setminus T$ is also a critical vertex subset of G which contains less vertices than U_c . It contradicts the choice of U_c and therefore completes the proof of the theorem. $\square$

Proof of Theorem 6. Let U_c be a critical vertex subset of G . Let $T_1, \dots, T_t$ be all nontrivial components of the induced subgraph $G(U_c)$. Then by Theorem 7, $U_c \setminus \{T_1, \dots, T_t\}$ is a critical independent set of G . Since finding a critical vertex subset U_c and deleting all vertices of U_c incident with some edge of $G(U_c)$ need only polynomial cost, finding a critical independent set is solvable in polynomial time.

Alternating proof of the theorem (also see [GZ]). Consider a weight $w: V(G) \rightarrow [0, 1]$ such that

$$0 \leq w(v) \leq 1 \quad \text{for each vertex } v \text{ of } G$$

$$\text{and } w(u) + w(v) \leq 1 \quad \text{for each edge } (u, v) \text{ of } G.$$

The total weight $\sum_{u \in V(G)} w(u)$ is denoted by $w(G)$. Let

$$w_m(G) = \max \{ w(G) : w \text{ is a weight of } G \text{ satisfies the above definition} \}.$$

Obviously finding w_m is a linear programming problem. By an argument similar to the proof of Theorem 2, we can prove that the set $V_b = \{ v \in V(G) : w_m(v) > \frac{1}{2} \}$ is a critical independent set of G . $\square$

(Note that the weight w defined above is called a fractional independence function of a graph G which was introduced by Domke, Hedetniemi, and Laskar in [DHL] and by Grinstead and Slater in [GS2], and was studied by Grinstead and Slater in [GS1] and by Goldwasser and Zhang in [GZ].)

REFERENCES

- [DHL] G. S. DOMKE, S. T. HEDETNIEMI, AND R. C. LASKAR, *Fractional packings, coverings, and irredundance in graphs*, preprint.
- [GJ] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability*, W. H. Freeman and Company, New York, 1979.
- [GJS] M. R. GAREY, D. S. JOHNSON, AND L. STOCKMEYER, *Some simplified NP-complete graph problems*, Theoret. Comput. Sci., 1 (1976), pp. 237–267.
- [GS1] D. L. GRINSTEAD AND P. J. SLATER, *On fractional covering and fractional independence parameters*, Third Carbondale Combinatorics Conference, Southern Illinois Univ., Carbondale, IL, 1988.
- [GS2] ———, *Fractional domination and fractional packings in graphs*, preprint.
- [GZ] J. GOLDWASSER AND C. Q. ZHANG, *Fractional independence number and fractional matching number*, preprint.
- [HP] P. HALL, *On representatives of subsets*, J. London Math. Soc., 10 (1935), pp. 26–30.
- [MB] B. MOHAR, *Isoperimetric number of graphs*, preprint.
- [MG] G. J. MINTY, *On maximal independent sets of vertices in claw-free graphs*, J. Combin. Theory Ser. B, 28 (1977), pp. 283–304.
- [WD] D. R. WOODALL, *The binding number of a graph and its Anderson number*, J. Combin. Theory Ser. B, 15 (1973), pp. 225–255.

IRREGULAR ASSIGNMENTS OF TREES AND FORESTS*

M. AIGNER† AND E. TRIESCH‡

Abstract. Let G be a graph on n vertices. An *irregular assignment* of G is a weighting $w: E(G) \rightarrow \{1, \dots, m\}$ of the edge-set of G such that all weighted degrees $w(v) = \sum_{v \in e} w(e)$ are distinct. The minimal number m for which this is possible is called the *irregularity strength* $s(G)$ of G . Lehel and others have shown that $s(G) < \infty$ implies $s(G) \leq n - 1$ for connected graphs on $n \geq 4$ vertices, and $s(G) \leq 2n - 3$ for arbitrary graphs. By using decompositions of the additive group $\mathbb{Z}_r$ (integers mod r), these results are strengthened. Main Theorem: $s(G) \leq n + 1$ for any graph with $s(G) < \infty$.

Key words. irregular assignments, partitions of congruences, extremal number theory

AMS(MOS) subject classifications. 05C35, 10A50, 15A36

1. Introduction. One of the first facts we learn about the degree sequence of a (simple) graph G is that in any graph with at least two vertices there are always two of the same degree. Hence the following natural problem arises: consider a weighting $w: E(G) \rightarrow \{1, \dots, m\}$ and denote by $w(v) = \sum_{v \in e} w(e)$ the weighted degree of the vertex $v \in V$. We call the weighting w *admissible* or an *irregular assignment* if all weighted degrees are distinct. What is the minimum number m for which an admissible weighting is possible? This number $s(G)$ is called the *irregularity strength* of G . Our observation above says $s(G) \geq 2$ for any graph G . It is an easy exercise to show that $s(G)$ exists if and only if G has no isolated edge and at most one isolated vertex.

In particular, $s(G)$ exists for any connected graph with at least three vertices. The study of $s(G)$ was apparently initiated in [1] where it was shown that $s(G) \leq 2n - 3$ for any graph G . In [2], the following result was derived: Any connected graph on $n \geq 4$ vertices has irregularity strength $s(G) \leq n - 1$. Since, clearly, $s(St_n) = n - 1$ for the star St_n , this bound is attained for every n .

The aim of this paper is to describe a simple method, using decompositions of the additive group $\mathbb{Z}_n$, to derive sharper results on trees, forests and general graphs.

2. Upper bound for trees and connected graphs. Let T be a tree on $n \geq 3$ vertices, and let B be its incidence matrix, with the rows corresponding to the vertices and the columns corresponding to the edges. We note four facts:

- (1) Since T is bipartite, there exists a vector $u \in \mathbb{Z}^n$ with entries $+1$ and -1 such that $u^T B = 0$.
- (2) A weighting $x \in \mathbb{N}^{n-1}$ of the edge-set $E(T)$ with the associated weighted degree vector $y \in \mathbb{N}^n$ corresponds to $Bx = y$. Our aim is thus to find x such that $y = Bx$ has different entries.
- (3) Viewing B as the linear map $B: \mathbb{R}^{n-1} \rightarrow \mathbb{R}^n$ we notice that B is injective (since $\text{rg}(B) = n - 1$) where $\text{im}(B)$ has dimension $n - 1$. Since for $y \in \text{im}(B)$ we have $u^T y = u^T Bx$ for some $x \in \mathbb{R}^{n-1}$, it follows that $y \in \text{im}(B) \Leftrightarrow u^T y = 0$.
- (4) Since T is bipartite, the matrix B is totally unimodular. Hence for any $y \in \text{im}(B) \cap \mathbb{Z}^n$ the unique preimage x is in $\mathbb{Z}^{n-1}$.

Let $r \in \mathbb{N}$, and $y, y' \in \mathbb{Z}^n$. We write $y \equiv y' \pmod{r}$ if and only if $y_i \equiv y'_i \pmod{r}$ for all $i = 1, \dots, n$.

* Received by the editors July 24, 1989; accepted for publication November 9, 1989.

† II. Mathematisches Institut, Freie Universität Berlin, Arnimallee 3, D-1000 Berlin 33, West Germany.

‡ Mathematisch-Naturwissenschaftliche Fakultät, Rheinisch-Westfälisch Technische Hochschule, Templergraben 64, D-5700 Aachen, West Germany.

LEMMA 1. Let T be a tree with incidence matrix B , and $r \in \mathbb{N}$. For $y \in \text{im}(B) \cap \mathbb{Z}^n$, there exists $\tilde{x} \in \{1, \dots, r\}^{n-1}$ such that $B\tilde{x} = \tilde{y}$ and $\tilde{y} \equiv y \pmod{r}$.

Proof. Let x be such that $Bx = y$. Then by (4), $x \in \mathbb{Z}^{n-1}$. We reduce $x \pmod{r}$ to $\tilde{x}$ and set $\tilde{y} = B\tilde{x}$. Then clearly $\tilde{y} \equiv y \pmod{r}$, as required. $\square$

As an example, suppose $r = n$ and $y \in \text{im}(B) \cap \mathbb{Z}^n$ has pairwise incongruent entries $\pmod{n}$. Then $\tilde{x}$ with $B\tilde{x} = \tilde{y}$ as in the lemma is an admissible weighting. In fact, it is enough to consider $y \in \text{im}(B) \cap \mathbb{Z}^n \pmod{n}$, since any $y \in \mathbb{Z}^n$ with $u^T y \equiv 0 \pmod{n}$ can clearly be transformed into $y' \in \text{im}(B) \cap \mathbb{Z}^n$ with $y' \equiv y \pmod{n}$. Hence in order to find $y \in \text{im}(B) \cap \mathbb{Z}^n$ we need to find a partition $\mathbb{Z}_n = R \cup S$ with R, S corresponding to the color classes of T such that $\sum_{a \in R} a \equiv \sum_{b \in S} b \pmod{n}$. The method consists then in finding appropriate partitions of $\mathbb{Z}_n$, or, in general, appropriate subsets $R, S \subseteq \mathbb{Z}_r$ where $\mathbb{Z}_r$ denotes the additive group modulo r .

The following notation will be useful. For $R \subseteq \mathbb{Z}_r$ we write $\sum_R$ shortly for $\sum_{a \in R} a \pmod{r}$. By a pair $P_i = \{i, r-i\}$ we always mean a pair of inverses, summing to 0 $\pmod{r}$, $i \leq r/2$. Note that for odd r there are $(r-1)/2$ pairs, thus $\mathbb{Z}_r = \bigcup_{i=1}^{(r-1)/2} P_i \cup \{0\}$ whereas for even r there are $r/2 - 1$ pairs, thus $\mathbb{Z}_r = \bigcup_{i=1}^{r/2-1} P_i \cup \{0, r/2\}$.

LEMMA 2. Let $r \in \mathbb{N}$, $r \geq 3$ and $2 \leq m \leq k \leq r$, $(m, k) \neq (r-2, r-1)$. Then there exist $R, S \subseteq \mathbb{Z}_r$ such that

- (a) $|R| = k$, $|S| = r + m - k$, $R \cup S = \mathbb{Z}_r$, $|R \cap S| = m$;
- (b) $\sum_R \equiv \sum_S \pmod{r}$.

Proof. Note first that for $m = r-2$, $k = r-1$ such that R, S cannot exist, since in this case $R = \mathbb{Z}_r - \{i\}$, $S = \mathbb{Z}_r - \{j\}$, $i \neq j \pmod{r}$ whence $\sum_R \not\equiv \sum_S \pmod{r}$. We set $R = M \cup X$, $S = (\mathbb{Z}_r \setminus R) \cup M$, $|M| = m$, $|X| = k - m$. Condition (b) is thus satisfied if and only if $2 \sum_X \equiv \sum_{\mathbb{Z}_r} - \sum_M \pmod{r}$.

Case 1. $r \equiv 1 \pmod{2}$. In this case $\sum_{\mathbb{Z}_r} \equiv 0$, hence our condition is

$$(5) \quad 2 \sum_X \equiv - \sum_M \pmod{r}, \quad |M| = m, \quad |X| = k - m, \quad M \cap X = \emptyset.$$

If $k \equiv m \equiv 0 \pmod{2}$, set $M = \bigcup_1^{m/2} P_i$, $X = \bigcup_{m/2+1}^{k/2} P_i$. Then $2 \sum_X \equiv \sum_M \equiv 0 \pmod{r}$.

If $k \equiv m \equiv 1 \pmod{2}$, set $M = \bigcup_1^{(m-1)/2} P_i \cup \{0\}$, $X = \bigcup_{(m+1)/2}^{(k-1)/2} P_i$.

If $k \equiv 1$, $m \equiv 0 \pmod{2}$, set $M = \bigcup_1^{m/2} P_i$, $X = \bigcup_{m/2+1}^{(k-1)/2} P_i \cup \{0\}$.

If $k \equiv 0$, $m \equiv 1 \pmod{2}$, we have $3 \leq m < k \leq r-1$. Suppose first $k \leq r-3$, and set $M = \bigcup_3^{(m+3)/2} P_i \cup \{r-2\}$, $X = \bigcup_{(m+5)/2}^{k/2+1} P_i \cup \{1\}$. Then $2 \sum_X \equiv 2 \equiv - \sum_M \pmod{r}$. For $k = r-1$, $m \leq r-4$, we set $M = \bigcup_3^{(m+3)/2} P_i \cup \{r-2\}$, $X = (\mathbb{Z}_r \setminus M) \setminus \{1\}$, then $2 \sum_X \equiv -2 \sum_M - 2 \equiv 2 \equiv - \sum_M$.

Case 2. $r \equiv 0 \pmod{2}$. Here we have $\sum_{\mathbb{Z}_r} \equiv r/2 \pmod{r}$, hence our condition is

$$(6) \quad 2 \sum_X \equiv r/2 - \sum_M \pmod{r}, \quad |M| = m, \quad |X| = k - m, \quad M \cap X = \emptyset.$$

If $k \equiv m \equiv 0 \pmod{2}$, set $M = (\bigcup (m/2 - 1) \text{ pairs}) \cup \{0, r/2\}$, $X = \bigcup (k - m)/2 \text{ pairs}$. Then $2 \sum_X \equiv 0 \equiv r/2 - \sum_M \pmod{r}$.

If $k \equiv m \equiv 1 \pmod{2}$, set $M = \bigcup_1^{(m-1)/2} P_i \cup \{r/2\}$, $X = \bigcup_{(m+1)/2}^{(k-1)/2} P_i$.

If $k \equiv 0$, $m \equiv 1 \pmod{2}$, set $M = (\bigcup (m-1)/2 \text{ pairs}) \cup \{r/2\}$, $X = (\bigcup (k - m - 1)/2 \text{ pairs}) \cup \{0\}$.

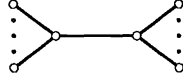
If $k \equiv 1$, $m \equiv 0 \pmod{2}$, we distinguish the cases $r \equiv 0$ and $r \equiv 2 \pmod{4}$. Suppose first $r \equiv 0 \pmod{4}$, and suppose $k \leq r-3$. Setting $M = \bigcup_{i \neq r/4} m/2 \text{ pairs } P_i$, $X = \{r/4\} \cup \bigcup_{i \neq r/4} (k-1-m)/2 \text{ pairs } P_i$, we have $\sum_M \equiv 0 \equiv r/2 - 2 \sum_X \pmod{r}$. For $k = r-1$, $m \leq r-4$, set $M = \bigcup_{i \neq r/4} m/2 \text{ pairs } P_i$, $X = (\mathbb{Z}_r \setminus M) \setminus \{r/4\}$, and we again

have $\sum_M \equiv 0 \equiv r/2 - 2 \sum_X \pmod{r}$. Now suppose $r \equiv 2 \pmod{4}$. For $k \leq r - 3$, set $M = (\cup_{i \neq 1, (r-2)/4} m/2 - 1 \text{ pairs } P_i) \cup \{0, 1\}$, $X = (\cup_{i \neq 1, (r-2)/4} (k - m - 1)/2 \text{ pairs } P_i) \cup \{(r - 2)/4\}$. Then $\sum_M \equiv 1 \equiv r/2 - 2(r - 2)/4 \equiv r/2 - 2 \sum_X \pmod{r}$. This works unless $1 = (r - 2)/4$, i.e., $r = 6$, $k = 3$, $m = 2$, in which case we take $M = \{0, 1\}$, $X = \{4\}$. Finally, for $k = r - 1$, $m \leq r - 4$, we set $M = \{0, r - 1\} \cup (\cup_{i \neq 1, (r+2)/4} m/2 - 1 \text{ pairs } P_i)$, $X = (\mathbf{Z}_r \setminus M) \setminus \{(r + 2)/4\}$, and obtain $\sum_M \equiv -1$, $\sum_X \equiv r/2 - \sum_M - (r + 2)/4 \equiv (r - 2)/4 + 1 \equiv (r + 2)/4$, i.e., $\sum_M \equiv r/2 - 2 \sum_X \pmod{r}$. $\square$

THEOREM 1. *Let T be a tree on $n \geq 3$ vertices which has a matching containing m end-vertices, $2 \leq m \leq n/2$. Then $s(T) \leq n - m$, except when possibly $m = (n/2) - 1$, and the two color classes contain $n/2$ vertices each.*

Proof. Set $r = n - m$, and consider the color classes R, S of T , where $|R| = k$, $|S| = n - k = r + m - k$. By Lemma 2, we can choose $y \in \text{im}(B) \cap \mathbf{Z}^n$, where B is the incidence matrix of T , with $\sum_{u \in R} y(u) \equiv \sum_{v \in S} y(v) \pmod{r}$ and m pairs of congruent coordinates $y(u_i) \equiv y(v_i) \pmod{r}$ ($u_i \in R, v_i \in S$), and $Bx = y$, $x \in \{1, \dots, r\}^{n-1}$. Note that according to the lemma we are completely free to assign the congruences within R , respectively, within S . In particular, we may choose the m pairs $\{u_i, v_i\}$ of the matching. Since $n \geq 3$, the neighbor of an end-vertex has degree at least 2, and we infer $y(u_i) \neq y(v_i)$ for all i . Hence y consists of n different numbers, and we are finished. $\square$

COROLLARY 1. *Let T be a tree on $n \geq 3$ vertices. Then*

- (a) $s(T) \leq n - 2$ unless T is a star
 (b) $s(T) \leq n - 3$ unless T is a star or $T =$  or a path P_n ($n \leq 6$).

Proof. (a) If T is not a star, then there are two end-vertices which do not have a common neighbor. Hence $m \geq 2$ in Theorem 1. Note that the theorem applies to $n \neq 6$, but the case $n = 6$ can easily be checked directly.

(b) The trees that do not have a required matching are easily determined, and the result follows by checking again some small cases, and by appealing to a result of [1] that $s(P_n) \leq (n/2) + 1$. $\square$

As an application of our congruence-method we give a short alternative proof that $s(G) \leq n - 1$ for connected graphs G on $n \geq 4$ vertices. To extend our results on trees to arbitrary connected graphs we proceed as follows. First we choose a spanning tree T and a weighting $x : E(T) \rightarrow \{1, \dots, n - 1\}$. We extend x to a weighting $\bar{x}$ on all of $E(G)$ by assigning the weight $n - 1$ to all edges in $E(G) \setminus E(T)$. In this way the congruences $\pmod{n - 1}$ are preserved. The following lemma is proved in a similar fashion as before.

LEMMA 3. *Let $r \in \mathbb{N}$, $r \geq 4$, $r \neq 5, 6$, $2 \leq k \leq r - 2$.*

i) *There exist $R, S \subseteq \mathbf{Z}_r$ and $2 \leq i \leq r$, such that*

(a) $|R| = k$, $|S| = r + 1 - k$, $R \cup S = \mathbf{Z}_r$, $R \cap S = \{i\}$,

(b) $\sum_R \equiv \sum_S \pmod{r}$.

(c) $R \cap \{1, \dots, i - 1\} \neq \emptyset$, $S \cap \{1, \dots, i - 1\} \neq \emptyset$.

ii) *There exist $R, S \subseteq \mathbf{Z}_r$ and $2 \leq i \leq r$, $i \neq 1$, such that*

(a) $|R| = k$, $|S| = r - k$, $R \cup S = \mathbf{Z}_r$, $i \in R$

(b) $i + \sum_R \equiv \sum_S \pmod{r}$.

(c) $R \cap \{1, \dots, i - 1\} \neq \emptyset$, $S \cap \{1, \dots, i - 1\} \neq \emptyset$.

The essence of Lemma 3 is the following. Let T be a tree on $n \geq 5$, $n \neq 7$, vertices, $r = n - 1$. According to (i) we can find a weighting x of $E(T)$, $x \in \{1, \dots, n - 1\}^{n-1}$ such that $y = Bx$ has pairwise incongruent entries $\pmod{n - 1}$, except for one pair $\{u, v\}$ in different color classes. According to (ii) the weighting x has the property that $y = Bx$ has incongruent entries except for one pair in the same color class, and this special residue i satisfies $i \neq 1$.

THEOREM 2 [2]. *Let G be a connected graph on $n \geq 3$ vertices. Then $s(G) \leq n - 1$, unless for $G = K_3$, when $s(K_3) = 3$.*

Proof. The cases $n = 3, 4, 7$ can be handled directly. So let G have $n \neq 3, 4, 7$ vertices. Clearly, a vertex u is an end-vertex of a spanning tree if and only if u is not a cut-point. Let $\delta = \delta(G)$ be the minimal degree in G .

Claim 1. If u is not a cut-point, then we may assume $d(u) = \delta$.

Consider a spanning tree T with u as end-vertex. Let v be a vertex in the color class different from u . If $d(v) < d(u)$, then applying Lemma 3(i) we can choose an appropriate weighting x with $y = Bx$ and $y(u) \equiv y(v) \pmod{n-1}$. By extending x to $\bar{x}$ for all of $E(G)$ we have $\bar{y}(u) = y(u) + (d(u) - 1)(n - 1)$, and $\bar{y}(v) = y(v) + (d(v) - d_T(v))(n - 1) \leq y(u) + (d_T(v) - 1)(n - 1) + (d(v) - d_T(v))(n - 1) = y(u) + (d(v) - 1)(n - 1) < \bar{y}(u)$. Hence $\bar{x}$ is an admissible weighting. Now if v is in the same color class as u and $d(v) < d(u)$, then we may apply Lemma 3(ii).

Claim 2. We may assume that G contains cut-points.

Otherwise G is δ -regular by Claim 1. Take any spanning tree with u, v as end-vertices and w a neighbor of v , w not adjacent to u . (Such a tree exists since otherwise G would be a star for which we know $s(G) = n - 1$.) Choose the weighting according to Lemma 3 with $y(u) \equiv y(w) \equiv i \not\equiv 1 \pmod{n-1}$. Since $y(v) \not\equiv y(u)$, the weight j of the edge vw must be $\neq i$ and since $i \neq 1$ we may choose $j < i$. Extending x to $\bar{x}$ as before we have $\bar{y}(w) \leq j + (\delta - 1)(n - 1) < i + (\delta - 1)(n - 1) = \bar{y}(u)$.

Claim 3. We may assume that two non-cut-points are never joined by an edge.

Let w be a cut-point with $G \setminus w = G_1 \cup G_2 \cup \cdots \cup G_t$ ($t \geq 2$) its partition into connected components. Suppose u, v are non-cut-points which are joined by an edge in, say, G_1 . Choose a non-cut-point z in G_2 . Then there exists a spanning tree T with z and u as end-vertices and u joined to v . Choose the weighting x according to Lemma 3 with $y(v) \equiv y(z) \equiv i \pmod{n-1}$, $i \neq 1$. By the same argument as in Claim 2, the edge uv receives a weight $j \neq i$, and we may assume $j < i$. Since $d(v) = d(z) = \delta$, we conclude for the extended weighting $\bar{y}(z) > \bar{y}(v)$ as before.

Now if G is a connected graph without edges between non-cut-points it is immediately seen that G must contain a vertex u of degree 1. But now we are finished by choosing a weighting x with $y(u) \equiv y(v)$ where v is the neighbor of u , since then $\bar{y}(u) = y(u) < y(v) \leq \bar{y}(v)$ for the extended weighting. $\square$

3. Upper bound for forests and arbitrary graphs. Let $F = T_1 \cup \cdots \cup T_t$ be a forest with connected components T_i , $i = 1, \dots, t$. We denote by $B_1, \dots, B_t$ the incidence matrices of T_i . The incidence matrix B of F is then of the form

$$B = \begin{bmatrix} \boxed{B_1} & & & \\ & \boxed{B_2} & & 0 \\ & & \ddots & \\ & 0 & & \boxed{B_t} \end{bmatrix}.$$

Let $u_i \in \mathbb{Z}^n$ be the ± 1 -vector corresponding to the color-classes of the tree T_i with 0's in the other positions. Viewing $B: \mathbb{R}^{n-1} \rightarrow \mathbb{R}^n$ again as a map, we have $\text{rg}(B) = n - t$ and thus $y \in \text{im}(B) \Leftrightarrow u_i^T y = 0$ ($i = 1, \dots, t$). Set $n_i = |V(T_i)|$ ($i = 1, \dots, t$). Our task is then to find sets $Z_1, \dots, Z_t \subseteq \mathbb{Z}_r$ with $|Z_i| = n_i$ and decompositions $Z_i = R_i \cup S_i$ such that $\sum_{R_i} \equiv \sum_{S_i}$ for each i .

We choose an odd integer r large enough (we will see that $r \leq n + 1$ will do). A component tree T_i is of one of three forms. Either both $|R_i|, |S_i|$ are even, both are

odd, or one is even and one is odd. Let the number of these trees be c_{00} , c_{11} , c_{01} , respectively. The first two cases are easily disposed of. When $|R_i|$, $|S_i|$ are both even, we choose complementary pairs (mod r) for R_i , S_i so that $\sum_{R_i} \equiv \sum_{S_i} \equiv 0$. If both $|R_i|$, $|S_i|$ are odd, we choose $(|R_i| - 1)/2$, $(|S_i| - 1)/2$ pairs for R_i , respectively, S_i , and add any element k to both R_i and S_i . Then $\sum_{R_i} \equiv \sum_{S_i} \equiv k$, and by our usual method of assigning k on one side of T_i to an end-vertex, this will give an admissible weighting. If c_{11} is even, then we choose for these special elements $c_{11}/2$ pairs, and if c_{11} is odd, then we choose $(c_{11} - 1)/2$ pairs plus 0. Hence if n_{00} and n_{11} denote the total number of vertices, then we have used up $(n_{00} + n_{11} - c_{11})/2$ pairs, respectively, $(n_{00} + n_{11} - c_{11} - 1)/2$ pairs plus 0.

The really interesting case arises when $|R_i|$ is even and $|S_i|$ is odd. Here the “pair distribution” will not work. We must have a relation $x + y \equiv z \pmod{r}$ and then fill up with pairs. To cope with this situation we need a lemma from extremal number theory which is interesting in its own right. For $m \in \mathbb{N}$, let $t(m)$ be the smallest number such that there exist m pairwise disjoint triples $D_i = \{x_i, y_i, z_i\}$ with $1 \leq x_i, y_i, z_i \leq t(m)$ and $x_i + y_i = z_i$. Obviously, $t(m) \geq 3m$. Suppose $t(m) = 3m$, and assume first that m is even. Let a_{11} , a_{01} be the number of triples with $x_i \equiv y_i \equiv 1$ and $x_i \not\equiv y_i \pmod{2}$, respectively. Looking at the odd numbers among $\{1, \dots, 3m\}$ we conclude

$$\frac{3m}{2} = 2a_{11} + 2a_{01},$$

and thus $m \equiv 0 \pmod{4}$.

When m is odd, we obtain in the same way

$$\frac{3m+1}{2} = 2a_{11} + 2a_{01},$$

and thus $m \equiv 1 \pmod{4}$.

Hence only for $m \equiv 0, 1 \pmod{4}$ is $t(m) = 3m$ possible, whereas for $m \equiv 2, 3 \pmod{4}$ the best we can do is $t(m) = 3m + 1$. The construction in [6] for $m \equiv 0, 1 \pmod{4}$ and in [5] for $m \equiv 2, 3 \pmod{4}$, respectively, shows that this can indeed be achieved.

LEMMA 4 [5], [6]. *For $m \in \mathbb{N}$, we have $t(m) = 3m$ when $m \equiv 0, 1 \pmod{4}$, and $t(m) = 3m + 1$, when $m \equiv 2, 3 \pmod{4}$, where it can be achieved that $3m$ is the missing number.*

THEOREM 3. *Let F be a forest on $n \geq 3$ vertices without isolated edges and at most one isolated vertex. Then $s(F) \leq n + 1$.*

Proof. Denote by c_{00} , c_{11} , and c_{01} the number of component trees as before.

Case 1. $c_{01} = 2m \equiv 0 \pmod{2}$. Then $n \equiv 0 \pmod{2}$. Set $r = n + 1$, r odd. Suppose first $m \equiv 0, 1 \pmod{4}$. By the lemma, there are disjoint triples $D_i = \{x_i, y_i, z_i\}$, $x_i + y_i = z_i$, $1 \leq x_i, y_i, z_i \leq 3m = 3c_{01}/2 \leq n/2$. We replace each D_i by $D'_i = 2D_i = \{2x_i, 2y_i, 2z_i\}$ ($i = 1, \dots, m$) and define $D'_{m+i} = -D'_i = \{-2x_i, -2y_i, -2z_i\} \pmod{r}$. Clearly, all triples $D'_j = \{x'_j, y'_j, z'_j\}$ with $x'_j + y'_j \equiv z'_j \pmod{r}$ ($j = 1, \dots, 2m$) are pairwise disjoint in $\mathbb{Z}_r$ (since r is odd and $6m \leq n = r - 1$). To every one of the c_{01} odd trees T_j with $|R_j| \equiv 0$, $|S_j| \equiv 1 \pmod{2}$ we assign x'_j, y'_j to R_j and z'_j to S_j . Note that the $6m$ elements in $\cup D'_j$ are made up of $3m = 3c_{01}/2$ pairs. Hence there are $n/2 - 3c_{01}/2 = \frac{1}{2}[n_{00} + n_{11} + n_{01} - 3c_{01}]$ pairs left over, so that the rest of the odd trees and all the even trees can be filled up with pairs. In this way, we always have $\sum_{R_i} \equiv \sum_{S_i}$, and hence obtain a required weighting by our usual method.

Now suppose $m \equiv 2, 3 \pmod{4}$. If $n \geq 3c_{01} + 2 = 6m + 2$, then we choose our triples $D_i = \{x_i, y_i, z_i\}$, $x_i + y_i = z_i$, $1 \leq x_i, y_i, z_i \leq 3m + 1$ according to the lemma.

Setting $D'_i = 2D_i$ and $D'_{m+i} = -D'_i \pmod{r}$ as before, we have $1 \leq x'_i, y'_i, z'_i \leq 6m + 2 \leq n$, and all triples are distinct, whence we can proceed as before. If $n = 3c_{01}$, then F consists of c_{01} paths of size 3, and here the result can be seen directly. In fact, Kinch and Lehel have proved that in this case $s(F) \leq \lceil (5n - 1)/7 \rceil + 1$ holds [3]. For a precise statement of this result see § 4.

Case 2. $c_{01} = 2m + 1 \equiv 1 \pmod{2}$. Set $r = n$, and note that r is odd.

Suppose first $m + 1 \equiv 0, 1 \pmod{4}$. We choose, according to the lemma, for $i = 1, \dots, m + 1$ triples $D_i = \{x_i, y_i, z_i\}$, $x_i + y_i = z_i$, $1 \leq x_i, y_i, z_i \leq 3m + 3$. Assume $n \geq 6m + 7 = 3c_{01} + 4$. Setting $D'_i = 2D_i \pmod{r}$ for $i = 1, \dots, m + 1$ and $D'_{m+1+i} = -D'_i \pmod{r}$ for $i = 1, \dots, m$, we obtain disjoint triples $D'_j = \{x'_j, y'_j, z'_j\}$ with $x'_j + y'_j \equiv z'_j \pmod{r}$ and $1 \leq x'_j, y'_j, z'_j \leq 6m + 6$. Suppose $D'_{m+1} = \{k, l, k + l\}$, then the triple $-k, -l, -(k + l)$ together with 0 has not been used yet. If $c_{11} \equiv 1 \pmod{2}$, then we assign to the last such tree T_i (which has received 0, $i, -i \in R_i$, $0 \in S_i$ so far) instead $-k, -l, 0$ to R_i , and $-(k + l)$ to S_i . If $c_{11} \equiv 0 \pmod{2}$, then we replace D'_{m+1} by $\{k, -k, 0\}$ and thus have only pairs left.

The other two possibilities are $n = 3c_{01} + 2$ and $n = 3c_{01}$. In the latter case F consists of paths of size 3 which was dealt with before. In the former case, F consists of $c_{01} - 1$ paths of size 3 and one tree T of size 5. We choose our triples D'_j as before, however, one triple, say D'_{m+1} , is $D'_{m+1} = \{k, l, 6m + 6 \equiv 1\} \pmod{r = 6m + 5}$, with $k + l \equiv 1$. The element 1 appears already in another triple. Hence for T , corresponding to D'_{m+1} we have the elements $k, l, -k, -l, 0$ at our disposal (-1 has also already been used). Clearly, these elements can be assigned to T in a proper fashion, and we are through.

There remains the final case when $m + 1 \equiv 2, 3 \pmod{4}$. Here the analysis is a little more subtle, but otherwise proceeds along the same lines. $\square$

THEOREM 4. *For any graph G on $n \geq 3$ vertices without isolated edges and with at most one isolated vertex, $s(G) \leq n + 1$.*

Proof. Let $G = G_1 \cup \dots \cup G_t$ be the partition of G into its connected components. Our method consists in choosing a spanning forest $F = T_1 \cup \dots \cup T_t$, an admissible weighting x for F as in Theorem 3, and to extend this weighting to $\bar{x}$ on all of $E(G)$ by assigning $n + 1$ if $r = n + 1 \equiv 1 \pmod{2}$, and n if $r = n \equiv 1 \pmod{2}$ to all edges of $E(G) \setminus E(F)$ as in the proof of Theorem 2. Note that in the construction of Theorem 3, we have chosen distinct congruence classes for most of these trees, but e.g., for the c_{11} even trees T_i we have used the same congruence class once in either color class. If G_i corresponding to T_i has a vertex u of degree 1, then this presents no difficulties, since we may choose $y(u) \equiv y(v) \pmod{r}$ for the neighbor v whence $\bar{y}(v) > \bar{y}(u)$ in case $y(u) = y(v)$. But if G has no end-vertices we have to slightly modify our method.

To cope with this additional difficulty we proceed as follows: Consider an even component G_i . Whenever a spanning tree T_i exists in G_i with $|R_i| \equiv |S_i| \equiv 0 \pmod{2}$ choose such a tree. Let the total number of the even graphs G_i with $|R_i| \equiv |S_i| \equiv 0 \pmod{2}$ be c_{00} . Consider any other even graph G_j . Either G_j has a spanning tree T_j whose color classes R_j, S_j satisfy $|R_j| \geq 3, |S_j| \geq 3$, or the only spanning trees of G_j are stars T_j . In the second case, G_j must be itself a star since otherwise G_j would contain a spanning tree which is not a star. Let c'_{11} be the number of star components, and note that any star has end-vertices whence we may safely assign the same congruence class to both color classes as before. Denote by c_{11} the number of all other even graphs G_j , i.e., whose spanning trees T_j satisfy for their color classes $|R_j| \equiv |S_j| \equiv 1 \pmod{2}$, $|R_j| \geq 3, |S_j| \geq 3$. Finally, denote by c_{01} the number of odd graphs G_i . The total number of vertices for these four types of component graphs are denoted by $n_{00}, n'_{11}, n_{11}, n_{01}$, respectively.

Now we choose a spanning tree T_i for each graph G_i , and set $r = n + 1$ if n is even and $r = n$ if n is odd. For the c_{00} trees T_i of type 1 we choose $|R_i|/2$ and $|S_i|/2$ pairs (mod r) on either side, thus $\sum R_i \equiv \sum S_i$. For the c'_{11} stars T_i with $|R_i| = 1$, $|S_i| \equiv 1$ (mod 2), $|S_i| \geq 3$, we choose $(|S_i| - 1)/2$ pairs for S_i plus an element k for both R_i and S_i . Thus $\sum R_i \equiv \sum S_i \equiv k$ (mod r). As in the case of forests these special elements k are chosen in pairs plus 0 if $c'_{11} \equiv 1$ (mod 2). For the c_{11} trees T_i of type 3 we choose a triple (a, b, c) with $a + b \equiv c$ (mod r), assign $a, b, -c$ to R_i , $-a, -b, c$ to S_i and fill up with $(|R_i| - 3)/2$ and $(|S_i| - 3)/2$ pairs, respectively. For the c_{01} odd trees we choose as before $x + y \equiv z$ (mod r), assign x, y to R_i , z to S_i , and fill up with pairs.

It remains to show that, except for the stars, we can accomplish this assignment without duplicating congruence classes. This is done in a fashion entirely similar to the proof of Theorem 3.

Let us just treat the case when c_{01} is even. Set $m = c_{11} + c_{01}/2$, and suppose first $m \equiv 0, 1$ (mod 4). Note that n is even in this case, whence $r = n + 1$ is odd.

By Lemma 4, there are disjoint triples $D_i = \{x_i, y_i, z_i\}$, $x_i + y_i = z_i$ ($i = 1, \dots, m$), with $1 \leq x_i, y_i, z_i \leq 3m = 3c_{11} + 3c_{01}/2$. Set $D'_i = 2D_i$ and $D'_{m+i} = -D'_i$ (mod r). Then for $D'_j = \{x'_j, y'_j, z'_j\}$ ($j = 1, \dots, 2m$), $x'_j + y'_j = z'_j$ (mod r), and since $6m = 6c_{11} + 3c_{01} \leq n_{11} + n_{01} \leq n$, all triples are distinct, and comprise among them $3m = 3c_{11} + \frac{3}{2}c_{01}$ pairs. Hence we may assign these triples to all $c_{11} + c_{01}$ trees as outlined above and fill up with the remaining

$$n/2 - 3c_{11} - \frac{3}{2}c_{01} = \frac{1}{2}[n_{00} + n'_{11} + (n_{11} - 6c_{11}) + (n_{01} - 3c_{01})]$$

pairs.

Now if $m \equiv 2, 3$ (mod 4), then by the same procedure we can choose $2m$ disjoint triples D'_j mod r , arising from triples $D_i = \{x_i, y_i, z_i\}$, $x_i + y_i = z_i$, $1 \leq x_i, y_i, z_i \leq 3m + 1$, as long as $6m + 2 \leq n$, i.e., $6c_{11} + 3c_{01} + 2 \leq n$. If $n = 6c_{11} + 3c_{01}$, then the spanning forest F consists of c_{01} paths of size 3 and c_{11} trees T_i with $|R_i| = |S_i| = 3$. If $n = 3$, then trivially $s(G) \leq 3$. Assume $n \geq 6$: Then the triples $D'_m = \{k, l, k + l = 6m + 2 = n + 2 \equiv 1 \pmod{r}\}$, and $D'_{2m} = \{-k, -l, -1\}$ appear. Now recall that according to Lemma 4, $3m$ is the unique number missing among the D'_i 's. Thus $6m = n \equiv -1$ (mod r) and 1 are the unique elements mod r missing in the D'_j 's. Hence D'_m and D'_{2m} can be taken after all. $\square$

It is probably safe to conjecture that it is possible to prove $s(G) \leq n - 1$ for any n -vertex graph G with $s(G) < \infty$, except for K_3 and possibly some other small cases, by looking more deeply into the construction of Lemma 4.

4. Disjoint paths of size 3. Suppose $F_n = T_1 \cup \dots \cup T_k$ is a forest all of whose component trees T_i are paths $\bullet - \bullet - \bullet$ of size 3. Thus F_n has $n = 3k$ vertices. Kinch and Lehel [3] showed for the irregularity number of F_n

$$\left\lceil \frac{5n-1}{7} \right\rceil \leq s(F_n) \leq \left\lceil \frac{5n-1}{7} \right\rceil + 1,$$

and Kinch conjectured that actually $s(F_n) = \lceil 5n - 1/7 \rceil$ for all $n \equiv 0$ (mod 3).

In this section we give the precise result for $s(F_n)$ which differs slightly from the conjecture.

Let $k \geq 1$. A weighting $\bullet \xrightarrow{a} \bullet \xrightarrow{b} \bullet$ of a path of size 3 corresponds to a triple $(a, b, a + b)$. Hence our task consists in finding k triples $T_i = (a_i, b_i, a_i + b_i)$, $1 \leq i, b_i \leq r$, such that all numbers $a_i, b_i, a_i + b_i$ ($i = 1, \dots, k$) are distinct. Let $s(k)$ be the smallest r for which this is possible, in other words $s(k) = s(F_n)$. Clearly, $s(k) \geq 2k$. The

smallest cases are $s(1) = 2, s(2) = 5, s(3) = 7, s(4) = 9$ with the triples

1 2 3, 1 2 3, 1 2 3, 1 2 3

4 5 9 4 6 10 4 9 13

5 7 12 5 7 12

6 8 14.

Our main tool of proof will be a graph-theoretic interpretation of the problem. Suppose $s(k) = 2k + m$, $m \geq 0$, with the k pairs (a_i, b_i) given where we always assume $a_i < b_i$. We call (a_i, b_i) an *inner pair* if $a_i + b_i \leq s(k)$, otherwise an *outer pair*. Since $s(k) = 2k + m$, there are at most m inner pairs. For example, in our examples above, we have no inner pair for $k = 1$, and for $k = 2, 3, 4$ one inner pair, namely $(1, 2)$.

Now we associate to the k pairs the following bipartite graph G_k . The vertices on both sides are the numbers $1, \dots, s(k)$, which we write on the left-hand side L in the order $1, 2, \dots, s(k)$ from top to bottom, on the right-hand side R in the order $s(k), \dots, 2, 1$. Set $S = s(k) + 1$. The *height* $h(v)$ of a vertex is its height from the bottom, thus $h(a) = S - a$ for $a \in L$, and $h(b) = b$ for $b \in R$. We call $a \in L, b \in R$ *complements* of each other if $h(a) = h(b)$ ($\Leftrightarrow a + b = S$).

For any outer pair (a_i, b_i) we put a_i into L and designate it by a dot $\bullet$, b_i in R and designate it by $\bullet$, and join a_i, b_i by an edge. The height $h(a_i, b_i)$ of the edge $a_i b_i$ is $h(a_i, b_i) = h(b_i) - h(a_i) = a_i + b_i - S$. Note that $h(a_i, b_i) \geq 0$ since (a_i, b_i) is an outer pair. Now comes the crucial fact: Since $a_i + b_i \neq a_j + b_j$ for any two outer pairs $(a_i, b_i), (a_j, b_j)$, we conclude that the heights $h(a_i, b_i)$ must all be distinct, and ≥ 0 .

We complete the graph G_k as follows. If dots $a_i \in L$ and $b_j \in R$ ($i \neq j$) are complements of each other, then we also join them by an edge. Suppose there are $i \leq m$ inner pairs, comprising $3i$ numbers $\leq s(k)$ among them. Then $2(k - i) + 3i = 2k + i$ numbers $\leq s(k)$ have been used in all k triples, and hence $m - i$ numbers are missing. We denote the set of "inner" numbers by I and the set of missing numbers by M , thus $|I| = 3i$, $|M| = m - i$. We designate all numbers of I by a circle $\circ$. If a circle c is a complement of a dot $a_i \in L$, then we put c in R , and if c is a complement of a dot $b_j \in R$, then we put c in L , otherwise we put c in L or R at will. Note that c cannot be complementary to both $a_i \in L, b_j \in R$ since otherwise $a_i = b_j$. Finally, we designate all numbers of M by a symbol $\times$, and proceed in the same way as for circles. In this way, we have designated all numbers $1, 2, \dots, s(k)$ in G_k by $\bullet, \circ, \times$ exactly once.

The following example explains the construction. Consider $k = 10, s(k) = 22, m = 2, i = 1$. Hence $|I| = 3, |M| = 1$. Figure 1 shows the triples of the left, and the graph G_{10} on the right, where we delete the uninteresting lower part.

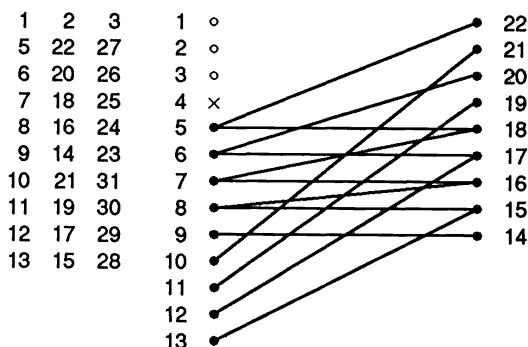


FIG. 1

Consider the subgraph H_k consisting of the edges of G_k . Any vertex of H_k has degree 1 or 2. Hence H_k is composed of (even) cycles, alternating with outer edges and complementary edges, and paths from a vertex $a \in L$ to a vertex $b \in R$. But since all outer edges lead upward from left to right, there can be no cycles. Hence H_k consists of paths $P = [a, b]$, a end-vertex in L , b end-vertex in R . Let the height of the path be $h(P) = h(b) - h(a)$. If we denote by H the total height sum, $H = \sum h(a_i, b_i)$ over all outer edges (a_i, b_i) , then clearly

$$H = \sum h(P) \quad \text{over all paths } P \text{ of } H_k.$$

In our example above, we have $H = \sum_{j=0}^8 j = 36$, and $h[9, 14] = 0$, $h[10, 21] = 8$, $h[11, 19] = 7$, $h[12, 20] = 9$, $h[13, 22] = 12$ which again sums to 36.

With these preparations we can proceed to the main lemma.

LEMMA 5. Suppose $s(k) = 2k + m$, $m \geq 0$, then $k \leq 7m + 1$, and $k = 7m + 1$ only if $t(m) = 3m$ in the notation of Lemma 4, i.e., there exist m disjoint triples $D_i = \{x_i, y_i, x_i + y_i\}$ with $1 \leq x_i, y_i, x_i + y_i \leq 3m$ ($\Leftrightarrow m \equiv 0, 1 \pmod{4}$).

Proof. Suppose there are $i \leq m$ inner pairs, and hence $m - i$ missing numbers. Suppose further that H_k consists of l paths. We assume $m \equiv 0 \pmod{2}$, the case $m \equiv 1 \pmod{2}$ is completely analogous. We use the notation as above. Note that $S = 2k + m + 1 \equiv 1 \pmod{2}$, hence complementary pairs consist of different numbers. (This is the only difference to the case $m \equiv 1 \pmod{2}$). Let I_L and M_L denote the sets of $\circ$ and $\times$ complementary to a $\bullet$ in R ; and similarly I_R, M_R the sets complementary to a $\bullet$ in L . What are the end-vertices of the l paths in H_k ? Consider the vertex-set L . The end-vertices are of four types: a may be complementary to I_R or M_R , or the complement may also be in L , in which case the heights sum to S . Finally (a, b) may be a complementary outer edge, resulting in a path of size 2. The analogous statement holds for R .

If we denote by $\sum_{H_L}$ and $\sum_{H_R}$ the sum of the heights of the end-vertices in L , respectively, in R , then we infer

$$(7) \quad H = \sum_{H_R} - \sum_{H_L}$$

and

$$(8) \quad \sum_{H_L} = \sum_{j \in I_R} j + \sum_{j \in M_R} j + \begin{cases} \frac{l - |I_R| - |M_R|}{2} S & \text{if no complementary pair } (a, b) \\ \frac{l - 1 - |I_R| - |M_R|}{2} S + b & \text{if } (a, b) \text{ is there} \end{cases}$$

$$(9) \quad \sum_{H_R} = \sum_{j \in I_L} (S - j) + \sum_{j \in M_L} (S - j) + \begin{cases} \frac{l - |I_L| - |M_L|}{2} S & \text{if } (a, b) \text{ is absent} \\ \frac{l - 1 - |I_L| - |M_L|}{2} S + b & \text{if } (a, b) \text{ is there.} \end{cases}$$

It follows that

$$(10) \quad H = \sum_{H_R} - \sum_{H_L} = \frac{|I_R| + |M_R| + |I_L| + |M_L|}{2} S - \sum_{I_R} j - \sum_{M_R} j - \sum_{I_L} j - \sum_{M_L} j.$$

Let $D = I_R \cup M_R \cup I_L \cup M_L$, $|D| = d$. Then $d \leq 3i + (m - i) = m + 2i$. Since D consists of different numbers we have

$$(11) \quad \sum_{j \in D} j \geq \frac{d(d+1)}{2}.$$

Furthermore, since all $k - i$ outer edges have different height, we have

$$(12) \quad H \geq \sum_0^{k-i-1} j = \frac{(k-i-1)(k-i)}{2},$$

and thus conclude with $S = 2k + m + 1$ and (7) to (12)

$$(k-i-1)(k-i) \leq d(2k+m+1) - d(d+1),$$

or

$$(13) \quad k^2 - k(2i + 2d + 1) + i^2 + i - md + d^2 \leq 0.$$

Solution of the quadratic (13) yields

$$(14) \quad k \leq i + d + \frac{1}{2} + \sqrt{d(2i+m+1) + \frac{1}{4}}.$$

Since $d \leq m + 2i \leq 3m$, we infer

$$d(2i+m+1) + \frac{1}{4} \leq 3m(3m+1) + \frac{1}{4} = (3m + \frac{1}{2})^2$$

and hence from (14)

$$k \leq 3m + i + \frac{1}{2} + 3m + \frac{1}{2} = 6m + i + 1 \leq 7m + 1.$$

If $k = 7m + 1$, then by our inequalities above, we must have $i = m$, i.e., $M = \emptyset$, and $I = D$ must consist of precisely the numbers $\{1, 2, \dots, 3m\}$, i.e., $t(m) = 3m$. $\square$

Now we come to the upper bound.

LEMMA 6. *If $7m - 6 < k \leq 7m + 1$, and $m \equiv 0, 1 \pmod{4}$, then $s(k) = 2k + m$.*

Proof. Let $k = 7m + r$, $r \in \{1, 0, -1, \dots, -5\}$. Again there are two cases depending on whether $r \equiv 0$ or $\equiv 1 \pmod{2}$. We treat the case $r \equiv 1 \pmod{2}$, the other being analogous. First we choose m disjoint triples, comprising the numbers $1, \dots, 3m$, which is possible by Lemma 4. The remaining $k - m = 6m + r$ pairs (the outer pairs) are divided into two groups of sizes $3m + (r+1)/2$ and $3m + (r-1)/2$. Let us denote the pairs of the first group (a_j, b_j) ($j = 1, \dots, 3m + (r+1)/2$), and those of the second group (α_j, β_j) ($j = 1, \dots, 3m + (r-1)/2$). Now we set

$$\begin{aligned} a_j &= 3m + j, & b_j &= 2k + m + 2 - 2j, \\ \alpha_j &= 6m + \frac{r+1}{2} + j, & \beta_j &= 2k + m + 1 - 2j. \end{aligned}$$

Clearly

$$\begin{aligned} 3m &< a_1 < a_2 < \dots < a_{3m+(r+1)/2} < \alpha_1 < \dots < \alpha_{3m+(r-1)/2}, \\ 2k+m &= b_1 > \beta_1 > b_2 > \beta_2 > \dots > \beta_{3m+(r-1)/2} > b_{3m+(r+1)/2}. \end{aligned}$$

Since

$$b_{3m+(r+1)/2} = 2k - 5m - r + 1 = k + 2m + 1 > k + 2m = \alpha_{3m+(r-1)/2},$$

all these numbers are distinct, and between $3m + 1$ and $2k + m$. For the sums we obtain

$$a_j + b_j = 2k + 4m + 2 - j,$$

$$\alpha_j + \beta_j = 2k + 7m + \frac{r+3}{2} - j.$$

Hence

$$a_1 + b_1 > a_2 + b_2 > \cdots > a_{3m+(r+1)/2} + b_{3m+(r+1)/2},$$

$$\alpha_1 + \beta_1 > \alpha_2 + \beta_2 > \cdots > \alpha_{3m+(r-1)/2} + \beta_{3m+(r-1)/2}.$$

Finally, we have

$$a_1 + b_1 = 2k + 4m + 1 < 2k + 4m + 2 = \alpha_{3m+(r-1)/2} + \beta_{3m+(r-1)/2}$$

and

$$a_{3m+(r+1)/2} + b_{3m+(r+1)/2} = 2k + m + \frac{3-r}{2} > 2k + m,$$

thus all triples are distinct. $\square$

For the case when $k < 7m + 1$ and $m \equiv 2, 3 \pmod{4}$, we again make use of Lemma 4 which asserts that m disjoint triples exist, comprising the number $1, 2, \dots, 3m + 1$ with $3m$ missing. The details of the proof of the following lemma are otherwise completely analogous.

LEMMA 7. *If $7m - 6 < k < 7m + 1$ and $m \equiv 2, 3 \pmod{4}$, then $s(k) = 2k + m$.*

The case remains when $k = 7m + 1$ and $m \equiv 2, 3 \pmod{4}$. We already know that $s(k) > 2k + m$ from Lemma 5, and it is easily seen that $s(k)$ has to be raised by 1.

LEMMA 8. *If $k = 7m + 1$, $m \equiv 2, 3 \pmod{4}$, then $s(k) = 2k + m + 1$.*

We come to the main result of this section.

THEOREM 5. *Let F_n be the forest consisting of $n/3$ paths of size 3. Then $s(F_n) = \lceil (5n - 1)/7 \rceil$ unless when $n \equiv 45, 66 \pmod{84}$ in which case $s(F_n) = \lceil (5n - 1)/7 \rceil + 1$.*

Proof. In the notation of our lemmas, $s(F_n) = s(k)$ with $n = 3k$. By Lemmas 5–8, $s(k) = 2k + \lceil (k - 1)/7 \rceil = \lceil (15k - 1)/7 \rceil$, unless for $k = 7m + 1$ with $m \equiv 2, 3 \pmod{4}$, i.e., $k \equiv 15, 22 \pmod{28}$, when $s(k) = \lceil (15k - 1)/7 \rceil + 1$. Hence

$$s(F_n) = \left\lceil \frac{15n - 3}{21} \right\rceil = \left\lceil \frac{5n - 1}{7} \right\rceil,$$

unless for $n \equiv 45, 66 \pmod{84}$, when

$$s(F_n) = \left\lceil \frac{5n - 1}{7} \right\rceil + 1. \quad \square$$

REFERENCES

- [1] G. CHARTRAND, M. JACOBSON, J. LEHEL, O. OELLERMAN, S. RUIZ, AND F. SABA, *Irregular networks*, Proc. of the 250th Anniversary Conf. on Graph Theory, Fort Wayne, Indiana, 1986.
- [2] M. JACOBSON AND J. LEHEL, *A bound for the strength of an irregular network*, to appear.
- [3] L. KINCH AND J. LEHEL, *The irregularity strength of $t.P_3$* , to appear.
- [4] J. LEHEL, *Facts and quests on degree irregular assignments*, to appear.
- [5] E. S. O'KEEFE, *Verification of a conjecture of Th. Skolem*, Math. Scand., 9 (1961), pp. 80–82.
- [6] TH. SKOLEM, *On certain distributions of the integers in pairs with given difference*, Math. Scand., 5 (1957), pp. 57–68.

THE RANDOM WALK CONSTRUCTION OF UNIFORM SPANNING TREES AND UNIFORM LABELLED TREES*

DAVID J. ALDOUS†

Abstract. A random walk on a finite graph can be used to construct a uniform random spanning tree. It is shown how random walk techniques can be applied to the study of several properties of the uniform random spanning tree: the proportion of leaves, the distribution of degrees, and the diameter.

Key words. spanning tree, random tree, random walk on graph

AMS(MOS) subject classifications. 05C05, 05C80, 60C05, 60J10

1. Introduction. Let G be a finite connected graph on n vertices. A *random walk* on G is the discrete-time Markov chain with transition matrix P of the form

$$P(v, w) = 1/r_v \quad \text{if } (v, w) \text{ is an edge,} \\ = 0 \quad \text{if not,}$$

where r_v is the degree of v .

Let $(X_j; j \geq 0)$ be a random walk on G with X_0 arbitrary. For each vertex v let T_v be the first hitting time:

$$T_v = \min \{j \geq 0 : X_j = v\}.$$

It is elementary that the *cover time*

$$C = \max_v T_v$$

is finite (with probability 1). In terms of the first C steps of the walk we can define a subgraph $\mathcal{T}$ of G to consist of the $n - 1$ edges

$$(1) \quad (X_{T_v-1}, X_{T_v}); v \neq X_0.$$

It is clear that $\mathcal{T}$ is a random spanning tree. It is not immediately clear that it is a *uniform* random spanning tree, but such is in fact the case.

PROPOSITION 1. *Let $N(G)$ be the number of spanning trees t of G . Then $P(\mathcal{T} = t) = 1/N(G)$ for each t .*

We could regard $\mathcal{T}$ as a *rooted* tree with root X_0 . Then by taking X_0 uniform on G , we make $\mathcal{T}$ uniform on the set of all rooted spanning trees of G .

The proof, given in § 2, is part of a known circle of ideas relating spanning trees to Markov chain stationary distributions. Clearly Proposition 1 yields an algorithm for generating a uniform spanning tree in time $O(EC)$. It is known [5] that $EC \leq n^3$, and [14] that for regular graphs $EC \leq 8n^2$. Broder [8] has independently noted Proposition 1 and discusses these algorithmic aspects at greater length. Our emphasis is rather on theoretical properties of uniform spanning trees, a topic on which there seems to be no literature at all. Proposition 1 opens the door to their study via modern random walk techniques. Proposition 5 bounds the probabilities of specified vertices being leaves of $\mathcal{T}$. Proposition 9 gives the degree of vertices in $\mathcal{T}$, where G has strong symmetry prop-

* Received by the editors December 27, 1988; accepted for publication (in revised form) January 25, 1990.

† Department of Statistics, University of California, Berkeley, California 94720. This research was supported by National Science Foundation grant MCS87-01426.

erties, and Proposition 10 gives asymptotics in that setting. Theorem 15 bounds the diameter of $\mathcal{T}$ on highly connected graphs.

Of course, $N(G)$ is in principle calculable via the classical matrix tree theorem given in many textbooks on graph theory. So we could in principle calculate $P(v \text{ is a leaf of } \mathcal{T})$ by calculating $N(G)$ and $N(G')$, where G' is the subgraph obtained by deleting v . But such direct combinatorial methods do not seem helpful for studying the diameter of $\mathcal{T}$, say, and apparently have not been studied even in the simple context of leaf probabilities.

Now let K_n denote the complete graph on vertices $1, \dots, n$. If $\mathcal{T}_n$ is a uniform spanning tree of K_n then clearly $\mathcal{T}_n$ is precisely the uniform random tree on n labelled vertices. Thus Proposition 1, applied to the complete graph, yields a construction of this random tree; in § 2 we observe that the construction can be rephrased as the following two-stage procedure.

ALGORITHM 2. Fix $n \geq 2$.

(i) For $2 \leq i \leq n$ connect vertex i to vertex $V_i = \min(U_i, i - 1)$, where $U_2, \dots, U_n$ are independent and uniform on $1, \dots, n$.

(ii) Relabel vertices $1, \dots, n$ as $\pi(1), \dots, \pi(n)$, where π is a uniform random permutation of $1, \dots, n$.

PROPOSITION 3. The random tree $\mathcal{T}_n$ produced by Algorithm 2 is uniform: $P(\mathcal{T}_n = t) = 1/n^{n-2}$ for each tree t on vertices $1, \dots, n$.

There are several schemes for enumerating labelled trees (e.g., Prufer code), and any such scheme can be used to generate the uniform random labelled tree. Broder (personal communication) has observed that the enumeration scheme in [15, § 2.3.4.4, Exercise 18] leads to a generation method which seems qualitatively similar, but not identical, to Algorithm 2. However, this algorithm does not seem to have been explicitly noted before. Its special feature is that when we consider the natural way of drawing the generated tree, we can easily intuit the limiting ($n \rightarrow \infty$) large-scale (diameter $\Theta(n^{1/2})$) behavior of uniform random trees, and define a limiting “continuum random tree” which determines all such limit properties. The topic is pursued in detail in [1].

We could seek to specialize Proposition 1 to other graphs to produce specific efficient algorithms for generating uniform spanning trees on that specific graph. Algorithm 4 in § 2 does this for the complete bipartite graph. It is not clear whether this algorithm corresponds to some explicit simple enumeration of the spanning trees of the complete bipartite graph.

Finally, let us mention some fascinating work-in-progress by Pemantle [19], who uses the random walk construction to discuss uniform random spanning trees of the infinite integer lattice $\mathbb{Z}^d$.

2. Proofs of Propositions 1 and 3. A frequently rediscovered result (related to the matrix tree theorem) gives the stationary distribution of a finite irreducible Markov chain. For each rooted tree t directed toward the root, let $q(t) = \prod P(v, w)$, the product taken over directed edges (v, w) of t . Let $p(v) = \sum_{t: \text{root}(t)=v} q(t)$. Then the stationary distribution is proportional to $p(\cdot)$. This result has a natural probabilistic interpretation, as follows. Run the stationary chain X_j from time $-\infty$ to time 0, and construct the random tree $\mathcal{T}$ with directed edges (X_{L_w}, X_{L_w+1}) , $w \neq X_0$ where L_w is the time of the final visit to w . Then it can be verified that $P(\mathcal{T} = t)$ is proportional to $q(t)$ and so, summing over trees t rooted at v , $P(X_0 = v)$ is proportional to $p(v)$. The author heard this interpretation from Anantharam and Tsoucas [6] and from Diaconis (personal communication), who attributes it to Peter Doyle; it appears in an undergraduate thesis of a student of Doyle’s [21].

Proposition 1 is formally merely a specialization of these ideas, in which we use the time-reversibility of random walks on graphs to express the result in terms of first hitting times instead of last exit times. A direct proof is sufficiently straightforward that it seems worth giving. Recall the proposition asserts that the tree $\mathcal{T}$ constructed via (1) is uniform over the set of all spanning trees.

Proof of Proposition 1. Suppose first that G is r -regular. Let $\mathcal{S}$ be the set of rooted spanning trees in G . We can consider the random walk as a stationary process $(X_j; -\infty < j < \infty)$ indexed by the integers. Write S_j for the tree constructed by $(X_j, X_{j+1}, \dots)$. That is, the tree rooted at X_j and with edges

$$(X_{T_v^j - 1}, X_{T_v^j}); v \neq X_j$$

where

$$T_v^j = \min \{k \geq j : X_k = v\}.$$

Then $(S_j; -\infty < j < \infty)$ is a stationary $\mathcal{S}$ -valued Markov chain which (see below) is irreducible. Consider the transition matrix Q for this chain in reversed time:

$$Q(t, t') = P(S_{-1} = t' | S_0 = t).$$

Then

(a) given t , there are exactly r trees t' such that $Q(t, t') = 1/r$, and $Q(t, t') = 0$ for other t' .

(b) given t' , there are exactly r trees t such that $Q(t, t') = 1/r$, and $Q(t, t') = 0$ for other t .

Indeed, given $S_0 = t$ with $\text{root}(t) = v$, say, then X_{-1} has chance $1/r$ to be each of the r neighbors of v ; each possibility leads to S_{-1} being some tree t' , and there are no other possibilities for S_{-1} . A similar argument gives (b). Now (a) and (b) say that Q is doubly-stochastic, so its stationary distribution (the distribution of S_0) is uniform on $\mathcal{S}$. This establishes the result for rooted trees; the unrooted case follows.

In the nonregular case, write $r(t)$ for the degree of the root of t . Then (a) and (b) become:

(a') Given t , there are exactly $r(t)$ trees t' such that $Q(t, t') = 1/r(t)$, and $Q(t, t') = 0$ for other t' .

(b') Given t' , there are exactly $r(t')$ trees t such that $Q(t, t') = 1/r(t)$, and $Q(t, t') = 0$ for other t .

Thus for fixed t' ,

$$\sum_t r(t)Q(t, t') = r(t')$$

and it follows that S has stationary distribution proportional to $r(t)$.

So we have shown that when X_0 has the (stationary) distribution proportional to r_v , the tree $\mathcal{T}$ constructed by Proposition 1 has $P(\mathcal{T} = t)$ proportional to $r(t)$. Thus conditional on $X_0 = w$ (for arbitrary w), $\mathcal{T}$ is uniform on rooted spanning trees rooted at w ; and if we consider $\mathcal{T}$ as an unrooted tree it is uniform on the set of all unrooted spanning trees of G . Also, when X_0 is uniform, then $\mathcal{T}$ is uniform on all rooted spanning trees.

Finally, we need to check that the chain $(\mathcal{S}_j)$ is irreducible. It is enough to show that, for any spanning tree t , there exists a finite path $(x_0, x_1, \dots, x_c)$ which visits every

vertex and is such that the tree constructed via (1) is t . But this is easy: consider, e.g., the “depth-first search” of the tree t .

Proof of Proposition 3. Recall that the Proposition asserts that the random tree $\mathcal{T}_n$ constructed by Algorithm 2 is uniform over the set of all trees on vertices $\{1, 2, \dots, n\}$.

Let $Z_0, Z_1, \dots$ be independently uniform on $1, \dots, n$. Let $\pi_1, \pi_2, \dots$ be the distinct states hit first, second, $\dots$ by the process Z , and let $\xi_1, \xi_2, \dots$ be the corresponding first hitting times. Precisely,

$$\begin{aligned}\xi_1 &= 0, \\ \pi_1 &= Z_0, \\ \xi_{j+1} &= \min \{m > \xi_j : Z_m \text{ not in } \{\pi_1, \dots, \pi_j\}\}, \\ \pi_{j+1} &= Z_{\xi_{j+1}}.\end{aligned}$$

Let $L_{j+1} = Z_{\xi_{j+1}-1}$. Now consider the tree in which π_{j+1} is connected to L_{j+1} , for each $j \geq 1$. First, we assert that this is the uniform random labelled tree on n vertices. For the construction is unaffected if Z is replaced by the subsequence Z' in which terms Z_i identical with their predecessor Z_{i-1} are deleted. But Z' is a random walk on the complete graph, and the construction above is the construction of Proposition 1 on the complete graph, so the tree is a uniform spanning tree in the complete graph, i.e., the uniform labelled tree.

Thus we need to argue that this construction is equivalent to that of Algorithm 2. Regard π as a random permutation of $1, \dots, n$. Clearly π is the *uniform* random permutation. The construction of the tree above can be rephrased as a two-stage construction.

- (i) Connect $(j+1)$ to $\pi^{-1}(L_{j+1})$, $j = 1, \dots, n-1$.
- (ii) Relabel $1, \dots, n$ as $\pi_1, \dots, \pi_n$.

Now condition on the entire permutation π and consider the operation of stage (i). We assert that, regardless of the value of π , the entire stage (i) operates in the same way as stage (i) of Algorithm 2; this will complete the proof. To prove the assertion, fix j and condition on the “past” process $(Z_m : m \leq \xi_j)$. With probability $1 - j/n$ we have $\xi_{j+1} = \xi_j + 1$, which implies that $L_{j+1} = Z_{\xi_j}$ and hence $\pi^{-1}(L_{j+1}) = j$. Otherwise, $\xi_{j+1} = \xi_j + M + 1$ for some random $M \geq 1$. Conditionally on $\{M = m\}$, we have:

$$(Z_{\xi_j+1}, \dots, Z_{\xi_j+m}) \text{ are independent uniform}$$

$$\text{on the previously visited states } \{\pi_1, \dots, \pi_j\}.$$

So in particular, $L_{j+1} = Z_{\xi_j+m}$ is uniform on the previously visited states $\{\pi_1, \dots, \pi_j\}$, and so $\pi^{-1}(L_{j+1})$ is uniform on $\{1, \dots, j\}$. Combining these two facts, we see that

$$P(\pi^{-1}(L_{j+1}) = u | Z_1, \dots, Z_{\xi_j}) = 1/n \quad \text{for } 1 \leq u \leq j-1,$$

$$P(\pi^{-1}(L_{j+1}) = j | Z_1, \dots, Z_{\xi_j}) = 1 - (j-1)/n.$$

This conditional probability is unaffected by the value of π , and agrees with stage (i) of Algorithm 2.

Here is a similar algorithm for the complete bipartite (n_1, n_2) graph. The proof is similar, and so is omitted.

ALGORITHM 4. (i) Choose $(1, 1)$ or $(2, 1)$ as the root, with probabilities $n_1/(n_1 + n_2)$, $n_2/(n_1 + n_2)$, respectively.

(ii) Suppose we have constructed a bipartite tree with vertices $(1, j), j \leq j_1; (2, j), j \leq j_2$, and suppose, (without loss of generality), that the last vertex added was $(1, j_1)$. Then do one of the following.

- (a) With probability $(n_2 - j_2)/n_2$, add vertex $(2, j_2 + 1)$ and connect it to $(1, j_1)$.
 - (b) With probability $j_1 j_2 (n_2 - j_2) / (n_2 (n_1 n_2 - j_1 j_2))$, add vertex $(2, j_2 + 1)$ and connect it to $(1, J)$, where J is chosen uniformly on $1, \dots, j_1$.
 - (c) With probability $j_2 (n_2 - j_2) / (n_1 n_2 - j_1 j_2)$, add vertex $(1, j_1 + 1)$ and connect it to $(2, J)$, where J is chosen uniformly on $1, \dots, j_2$.
- (iii) Finally, relabel $(1, j), j \leq n_1$ as $(1, \pi(j)), j \leq n_1$, where π is a uniform random permutation of $1, \dots, n_1$; and similarly for $(2, j), j \leq n_2$.

Remark. In the special case of the complete graph, the tree process of the first paragraph of this section becomes the following Markov chain on the set of all rooted trees on vertices $\{1, \dots, n\}$. At each step pick a vertex v uniformly. If v is not the old root, remove the edge from v to its parent, create a new edge from v to the old root, and make v the new root. So the stationary distribution of this chain is the uniform distribution on all n^{n-1} rooted trees. This fact (easy to verify directly) is used by Fowler [12] to rederive distributions of vertex depths in the uniform random labelled tree.

3. Leaves of spanning trees. The rest of the paper gives some applications of the random walk construction of Proposition 1. We work with regular graphs for simplicity.

PROPOSITION 5. Let $\mathcal{T}$ be the uniform random spanning tree in an r -regular graph G , for $r \geq 3$.

- (i) $P(v \text{ is a leaf of } \mathcal{T}) \leq \exp(-(r-1)/2r)$ for all $v \in G$.
- (ii) $\text{ave}_{v \in G} P(v \text{ is a leaf of } \mathcal{T}) \geq \alpha(r)$, where

$$\alpha(r) = \sum_{j=2}^{r-1} r^{-1} (1 - j/r) \left(1 + \prod_{i=1}^{j-1} (1 - r^{-1} (1 - i/r)) \right).$$

Thus we have an absolute upper bound on probabilities of particular vertices being leaves of the random spanning tree. We cannot have such a lower bound—removing v may split G into r components, in which case v has degree r in all spanning trees. Instead, (ii) gives a lower bound on average leaf probabilities. We can calculate $\alpha(3) = \frac{2}{81}$, $\alpha(r) \uparrow e^{-1/2} - \frac{1}{2} \approx 0.106$ as $r \rightarrow \infty$. For references to deterministic extremal bounds (e.g., the maximum number of leaves, over all spanning trees of G) see [13].

Proof of Proposition 5. As in Proposition 1 let X_j be the random walk. Fix v and let A be the set of neighbors of v . Let ξ_i be the time of the i th visit to A . Let $\mathcal{T}_i$ be the tree constructed by the random walk up to time ξ_i . Define D_i by: $1 + D_i$ is the degree of v in $\mathcal{T}_i$, with the convention that $D_i = 0$ if v has not been visited before time ξ_i . Note that $D_1 = 0$ and $D_{i+1} = D_i$ or $D_i + 1$. Let $\Gamma(\xi_i)$ be the number of distinct elements of A visited through time ξ_i . We assert

$$P(D_{i+1} = D_i + 1 \mid D_1, \dots, D_i, \Gamma(\xi_i), X_{\xi_i}) = r^{-1} (1 - \Gamma(\xi_i)/r).$$

For the right side gives the conditional probability of the event

$$X_{\xi_i+1} = v \quad \text{and} \quad X_{\xi_i+2} \text{ is a previously unvisited vertex,}$$

and $D_{i+1} = D_i + 1$ if and only if this event occurs.

Since $\Gamma(\xi_i) \leq i$, we obtain

$$(2) \quad P(D_{i+1} = D_i + 1 \mid D_1, \dots, D_i) \geq r^{-1} (1 - i/r).$$

Next, for $1 \leq j \leq r$,

$$(3) \quad \begin{aligned} P(D_j = 0) &= \prod_{i=1}^{j-1} P(D_{i+1} = 0 \mid D_i = 0) \\ &\leq \prod_{i=1}^{j-1} (1 - r^{-1}(1 - i/r)) \text{ using (2).} \end{aligned}$$

We can now argue (i), because

$$\begin{aligned} P(v \text{ is a leaf of } \mathcal{T}) &\leq P(D_r = 0) \\ &\leq \prod_{i=1}^{r-1} (1 - r^{-1}(1 - i/r)). \end{aligned}$$

The bound in (i) follows using $1 - y \leq e^{-y}$.

To argue (ii), write $M = \min(i : D_i = 1) \leq \infty$. Write $1 + D_\infty$ for the degree of v in $\mathcal{T}$. Then

$$D_\infty = 1_{(M < \infty)} + \sum_{j=1}^{\infty} (D_{j+1} - D_j) 1_{(D_j > 0)}$$

and so

$$\begin{aligned} ED_\infty &= P(M < \infty) + \sum_{j=1}^{\infty} E(D_{j+1} - D_j) 1_{(D_j > 0)} \\ &\geq P(M < \infty) + \sum_{j=1}^{r-1} r^{-1}(1 - j/r) P(D_j > 0) \text{ using (2).} \end{aligned}$$

Now $M < \infty$ is the event that v is not a leaf of $\mathcal{T}$, so using (3) we find

$$P(v \text{ is a leaf of } \mathcal{T}) \geq 1 - ED_\infty + \alpha(r).$$

In any tree on n vertices the average degree equals $2 - 2/n$. So $\text{ave}_v ED_\infty = 1 - 2/n$ and (ii) follows.

For later use (Corollary 12) let us indicate how the argument for (i) leads to bounds for the entire distribution of D_∞ , rather than just $P(D_\infty = 0)$. Let $c(i, r)$ be constants,

$$c(0, r) = 0, \quad c(i, r) \leq c(i+1, r) \leq c(i, r) + 1.$$

COROLLARY 6. *Let $\mathcal{T}$ be the uniform random spanning tree in an r -regular graph G for $r \geq 3$. Let $1 + D$ be the degree of v in $\mathcal{T}$. Let $c(i, r)$ be as above. Let $\Gamma(\xi_i)$ be the number of distinct elements of A visited through the time of the i th visit to A , the neighbor set of v . Let D_i^* be independent $\{0, 1\}$ -valued random variables with $P(D_i^* = 1) = r^{-1}(1 - c(i, r)/r)$ and let $D^* = \sum_i D_i^*$. Then for each $u \geq 0$,*

$$P(D \leq u) \leq P(D^* \leq u) + P(\Gamma(\xi_i) > c(i, r) \text{ for some } i).$$

Proof. Write $\mathcal{F}_i$ for the σ -field of events up to time ξ_i . Write

$$G_i = \{ \Gamma(\xi_i) \leq c(i, r) \}.$$

The argument for (2) gives

$$P(D_{i+1} - D_i = 1 \mid \mathcal{F}_i) \geq r^{-1}(1 - c(i, r)/r) \quad \text{on } G_i.$$

There is now a natural construction of $\{0, 1\}$ -valued D_i^* such that

$$P(D_{i+1}^* = 1 \mid \mathcal{F}_i, D_1^*, \dots, D_i^*) = r^{-1}(1 - c(i, r)/r)$$

and

$$D_{i+1} - D_i \geq D_{i+1}^* \quad \text{on } G_i.$$

Writing $D^* = \sum_i D_i^*$, we see $D \geq D^*$ on $\cap G_i$, and the result follows.

4. Degrees in highly symmetric graphs. For highly symmetric graphs it is reasonable to hope we can compute explicitly the degree of a vertex in the uniform random spanning tree. Proposition 9 below deals with the very special class of graphs G which satisfy the following (very strong) hypothesis. Recall [7] that G is vertex-transitive if its automorphism group acts transitively on vertices.

Hypothesis 7. (i) G is vertex-transitive, with degree $r \geq 3$.

(ii) For each $v \in G$ and distinct neighbors w_1, w_2, w_3 of v , there exists a graph automorphism γ such that

$$\gamma(v) = v, \quad \gamma(w_1) = w_1, \quad \gamma(w_2) = w_3, \quad \gamma(w_3) = w_2.$$

This class includes the cube graphs Q_d , for example. This class is perhaps related to the class of *distance regular* graphs discussed exhaustively in [10].

We need some preliminaries about the random walk on such a graph G . Fix a vertex v with neighbors $\{w_1, \dots, w_r\} = A$, say. Distinguish w_1 and write $B = \{w_2, \dots, w_r\}$ for the “other neighbors.” We will consider for a while the random walk started at w_1 , and write P_{w_1} for probabilities associated with this walk. Recall that T_x is the first hitting time on x ; similarly let

$$T_B = \min \{j \geq 0 : X_j \in B\}$$

be the first time some other neighbor is hit, and let T^+ be the first time the walk returns to w_1 . The purpose of symmetry hypothesis (ii) is to imply the following. Let $w' \in A$ and let $B' = A \setminus \{w'\}$. Then

$$(4) \quad P_{w'}(X(T_{B'}) = w) = \frac{1}{r-1}; \quad w \in B'$$

$$(5) \quad P_{w'}(X(T_{B'}) = w, \quad X(T_{B'} - 1) = v) \text{ is uniform in } w.$$

We will use several parameters (a, b, c, ψ, θ) of the random walk, which are equivalent in the sense that any one of them determines the others. Define

$$(6) \quad a = P_{w_1}(T_{w_2} < T^+).$$

By the symmetry hypothesis, for any pair of vertices z, z' distance 2 apart, a is the chance that the random walk started at z hits z' before returning to z . Now consider the chance the walk hits some other neighbor before returning to w_1 :

$$b = P_{w_1}(T_B < T^+).$$

By symmetry there is chance $1/(r-1)$ that $X_{T_B} = w_2$; and if this does not occur, there is chance $\frac{1}{2}$ that the walk will thereafter hit w_2 before w_1 . Thus

$$a/b = 1/(r-1) + \frac{r-2}{2(r-1)}$$

which rearranges to

$$b = 2a(1 - r^{-1}).$$

Now consider the walk started at w_1 and run until it either hits B or returns to w_1 . Exactly one of the four alternatives below must happen; their probabilities are noted.

- (i) $\text{prob} = r^{-2}$ $X_1 = v, \quad X_2 = w_1$
- (ii) $\text{prob} = r^{-1} - r^{-2}$ $X_1 = v, \quad X_2 \in B$
- (iii) $\text{prob} = c, \text{ say}$ $T_B < \min(T^+, T_v)$
- (iv) $\text{prob} = 1 - r^{-1} - c$ $T^+ < \min(T_B, T_v)$.

Clearly b is the chance of (ii) or (iii), so $b = r^{-1} - r^{-2} + c$, which rearranges to

$$c = (2a - r^{-1})(1 - r^{-1}).$$

Recall now the *craps principle*. If a chance experiment has a set of alternate outcomes, and is repeated until the first time one of a specified subset of outcomes occurs ("the ultimate outcome"), then the relative probabilities of the possible ultimate outcomes are exactly their original relative probabilities. We apply this to calculate

$$\psi = P_{w_1}(T_B < T_v)$$

using the experiment: "start a random walk at w_1 and continue until it returns to w_1 ." We want the chance that (iii) occurs before (i) or (ii), and the craps principle says

$$\psi = c / (c + r^{-1})$$

which rearranges to

$$(7) \quad \psi = 1 - \frac{1}{2ar - 2a + r^{-1}}.$$

Similarly, consider

$$\theta = P_{w_1}(X_{T_B-1} \neq v).$$

This is the chance that (iii) occurs before (ii), so

$$\theta = c / (c + r^{-1} - r^{-2})$$

which rearranges to

$$(8) \quad \theta = 1 - \frac{1}{2ar}.$$

We now specify a certain joint distribution for random variables $\mathbf{V} = (V_0, V_1, V_2)$ and use it to state the main result of this section.

CONSTRUCTION 8. (i) $V_0 \geq 1$ and, for $1 \leq j \leq r - 1$,

$$P(V_0 \geq j + 1 \mid V_0 \geq j) = \frac{\psi(r-j)/(r-1)}{1 - \psi + \psi(r-j)/(r-1)}.$$

(ii) Given V_0 , $P(V_1 = 0) = V_0/r$ and $P(V_1 = 1) = 1 - V_0/r$.

(iii) Given V_0, V_1 , the variable V_2 has binomial $(r - V_0 - V_1, 1 - \theta)$ distribution.

PROPOSITION 9. Let G be a graph satisfying Hypothesis 7. Let $1 + D$ be the degree of a specified vertex v in the uniform random spanning tree $\mathcal{T}$ of G . Then D has the distribution of $V_1 + V_2$, for $\mathbf{V}$ as in Construction 8.

Remarks. An explicit expression for this distribution can be found from the construction, but is not very appealing. One simple feature is

$$P(D=r-1)=(1-1/r)(2ar-2a-1/r)^{-1}(2ar)^{2-r}.$$

Note that the distribution depends only on (r, a) and not on the size n of the graph. The asymptotics are simpler: see Proposition 10. For the complete graph it is known ([18, eq. (7.3)]) that D has binomial $(n-2, 1/n)$ distribution: here Proposition 9 applies with $r=n-1$ and $a=\frac{1}{2}n/(n-1)$. For the cube graph Q_d we can calculate

$$(9) \quad a = \frac{1}{2}(1-1/d)/(1-2^{1-d})$$

and deduce the distribution of D from the Proposition. Finally, recall that a priori $ED = 1 - 2/n$, so by taking expectations in Proposition 9 we get an expression for n in terms of (a, r) . This suggests that a is perhaps determined by (n, r) .

Proof of Proposition 9. Consider the random walk X_i generating the tree $\mathcal{T}$ as in Proposition 1. Let us suppose $X_0 \neq v$: we shall later indicate the modification for the case $X_0 = v$. Write $\Gamma(m)$ for the number of distinct neighbors of v visited through time m . Define V 's in terms of the random walk as follows.

- (a) $V_0 = \Gamma(T_v)$.
- (b) $V_1 = 0$ or 1 accordingly as X_{T_v+1} was or was not previously visited.
- (c) V_2 is the number of states j satisfying both

$$T_j > T_v + 1; \quad X_{T_j-1} = v.$$

Then the degree of v in $\mathcal{T}$ is $1 + V_1 + V_2$, so the issue is proving that the joint distribution of these V 's is as specified in Construction 8.

As in § 3 let ξ_i be the time of the i th visit to the set A of neighbors of v . Fix i . Let S be the first time after ξ_i that the walk hits v or $A \setminus \{X_{\xi_i}\}$. There are three alternatives, with probabilities as stated.

- (i) prob = $1 - \psi$ $X_S = v$.
- (ii) prob = $\psi \frac{r - \Gamma(\xi_i)}{r - 1}$ $X_S \in A, \quad \Gamma(S) = \Gamma(\xi_i) + 1$.
- (iii) prob = $\psi \frac{\Gamma(\xi_i) - 1}{r - 1}$ $X_S \in A, \quad \Gamma(S) = \Gamma(\xi_i)$.

Now suppose $\Gamma(\xi_i) = j$, say, and that v has not yet been hit. Then $\Gamma(\cdot)$ will reach $j+1$ before v is visited if and only if (ii) occurs before (i), and by the craps principle this has chance

$$\frac{\psi(r-j)/(r-1)}{1 - \psi + \psi(r-j)/(r-1)}.$$

This gives part (i) of Construction 8. Part (ii) is clear.

For part (iii), fix i , suppose $\xi_i > T_v$ and $\Gamma(\xi_i) = j$. Let S be the first time after ξ_i that the walk hits $A \setminus \{X_{\xi_i}\}$. Again there are three possibilities, with probabilities stated.

- (i) prob = j/r $\Gamma(S) = j$.
- (ii) prob = $(1 - j/r)(1 - \theta)$ $\Gamma(S) = j + 1, \quad X_{S-1} = v$.
- (iii) prob = $(1 - j/r)\theta$ $\Gamma(S) = j + 1, \quad X_{S-1} \neq v$.

(The latter probabilities follow from (5)).

Now consider the first time S' that $\Gamma(\cdot)$ reaches $j + 1$. Then $X_{S'-1} = v$ if and only if (ii) occurs before (iii), and by the craps principle that has chance $1 - \theta$. This leads to part (iii) of Construction 8.

Finally, consider the case $X_0 = v$. We assert that the tree constructed by the walk $v = X_0, X_1, \dots$ has the same degree as the tree constructed by $X_1, \dots$, so this case is no different from the previous case. Let T^+ be the time of first return to v . Then the tree constructed by $X_0, \dots, X_{T^+}$ is the same as the tree constructed by $X_1, \dots, X_{T^+}$, except that the former has an edge (v, X_1) where the latter has an edge (X_{T^+-1}, v) . The set of neighbors visited thus far is the same in each case, so the subsequent evolution of edges (v, w) is the same.

5. Asymptotics of degrees. For each K let $G = G^K$, $v = v^K$ satisfy Hypothesis 7. Let $D = D^K$, $r = r^K$, $a = a^K$, etc., be as in the previous section. Here we study the behavior of D^K as $K \rightarrow \infty$. For notational convenience we will omit the superscripts K . Write $\mathcal{P}_\lambda$ for a Poisson (λ) random variable:

$$P(\mathcal{P}_\lambda = i) = e^{-\lambda} \lambda^i / i!, \quad i \geq 0.$$

Suppose $r \rightarrow \infty$ and $a \rightarrow \frac{1}{2}$. It is then straightforward to show that, for $\mathbf{V}$ as in Construction 8,

$$(V_0/n, V_1, V_2) \xrightarrow{d} (U_0, U_1, U_2)$$

where $\xrightarrow{d}$ denotes convergence in distribution, and where the limit distribution is described as follows.

- (i) U_0 is uniform on $[0, 1]$.
- (ii) $U_1 = 0$ or 1 ; $P(U_1 = 1 | U_0) = 1 - U_0$.
- (iii) Given U_0, U_1 , the variable U_2 has Poisson $(1 - U_0)$ distribution.

Now an easy transform argument shows

$$U_1 + U_2 \stackrel{d}{=} \mathcal{P}_1.$$

So Proposition 9 has the following asymptotic corollary.

PROPOSITION 10. *Let $1 + D^K$ be the degree of v^K in the uniform spanning tree on G^K , where (v^K, G^K) is a sequence satisfying Hypothesis 7. Suppose $r^K \rightarrow \infty$ and $a^K \rightarrow \frac{1}{2}$. Then $D^K \rightarrow \mathcal{P}_1$.*

For complete graphs (i.e., for the uniform random labelled tree) this is well known and immediate from the exact binomial distribution of D . On the cube graphs, (9) shows that Proposition 10 applies, and gives an apparently new result.

We suspect this asymptotic result holds without any explicit symmetry conditions.

CONJECTURE 11. *Let r^K be the degree of v^K in G^K . Let $1 + D^K$ be the degree of v^K in the uniform spanning tree on G^K . For w in the neighbor set A of v^K , let $\psi_w^K = P_w(T_{A \setminus w} < T_v)$. Suppose*

$$r^K \rightarrow \infty; \quad \sup_w |r^K(1 - \psi_w^K) - 1| \rightarrow 0.$$

Then $D^K \xrightarrow{d} \mathcal{P}_1$.

Now let us reconsider Corollary 6. Put $c(i, r) = i$. As $r \rightarrow \infty$, $ED^* \rightarrow \int_0^1 (1 - x) \times dx = \frac{1}{2}$. Then the Poisson limit theorem for independent events shows $D^* \xrightarrow{d} \mathcal{P}_{1/2}$, giving the following result.

COROLLARY 12. *Let $1 + D^K$ be the degree of v^K in the uniform spanning tree on G^K , where G^K is an r^K -regular graph and $r^K \rightarrow \infty$. Then*

$$\liminf_K P(D^K \leq u) \leq P(\mathcal{P}_{1/2} \leq u); \quad u = 0, 1, 2, \dots$$

CONJECTURE 13. *Corollary 12 holds with $\mathcal{P}_{1/2}$ replaced by $\mathcal{P}_1$.*

This is closely related to open problems concerning covering by reversible Markov chains, which we now describe briefly. In fact, Corollary 6 shows that Conjecture 13 is true if Conjecture 14 below is true; the latter could be applied to the random walk looked at only on the neighbor set of v . Note also that, since $\text{ave}_v D_v^K \rightarrow 1$, the truth of Conjecture 13 would imply that $D_v^K \rightarrow \mathcal{P}_1$ for “almost all” v .

Consider a Markov chain on r states with some *symmetric* transition matrix P . Let C_i be the number of distinct states visited during the first $i - 1$ steps. On the complete graph (with $r = n - 1$ large) we have $C_i \approx r(1 - \exp(-i/r))$. To prove Conjecture 13 we want some result of the type “no reversible chain can hit more distinct states in time t than does the random walk on the complete graph.” The following would suffice.

CONJECTURE 14. *Let $b(\varepsilon, r)$ be the maximum, over all r -state Markov chains with symmetric transition matrices, of*

$$P(C_i > (1 + \varepsilon)r(1 - \exp(-i/r)) \text{ for some } i).$$

Then $b(\varepsilon, r) \rightarrow 0$ as $r \rightarrow \infty$, for each $\varepsilon > 0$.

Known weaker results, in the context of the time taken to visit *all* states, are in [3].

6. Diameter of spanning trees. Another natural quantity to study is the diameter Δ of the uniform random spanning tree $\mathcal{T}$. For fixed $r \geq 3$ it is easy to construct r -regular graphs G on n vertices such that *every* spanning tree has diameter $\Omega(n)$, or such that *every* spanning tree has diameter $O(\log n)$. Thus we cannot expect any interesting “universal” results, analogous to Proposition 5, for the diameter. For the uniform random labelled tree, we have [22]:

$$(10) \quad \Delta/n^{1/2} \xrightarrow{d} L \text{ where } L \text{ has a certain nondegenerate distribution.}$$

It seems likely that exactly the same result holds for some class of graphs including the cube graphs Q_d ; this will be studied elsewhere. Here we present only some crude bounds in Theorem 15 below.

As before, let us consider only regular graphs G for simplicity. The transition matrix P for the random walk is symmetric, and so has real eigenvalues

$$1 = \lambda_1 > \lambda_2 \geq \lambda_3 \cdots.$$

Call $\tau = 1/(1 - \lambda_2)$ the *relaxation time*. Work on random walks on graphs has produced many results of the form “a result for the complete graph remains approximately true on graphs G for which τ is suitably small relative to n ”; see e.g., [2], [9], [20]. Theorem 15 shows that, if $\tau = o(n^\varepsilon)$ for all $\varepsilon > 0$ then $\Delta = \Omega(n^{1/2-\varepsilon})$ and $O(n^{1/2+\varepsilon})$ for all $\varepsilon > 0$.

Let K denote an absolute constant, not necessarily the same from line to line.

THEOREM 15. *Let Δ be the diameter of the uniform random spanning tree in a regular graph G on n vertices. Let τ be the relaxation time, as above. Then*

$$K^{-1}\tau^{-1}n^{1/2}(\log n)^{-1} \leq E\Delta \leq K\tau^{1/2}n^{1/2} \log n.$$

Let us make two vague conjectures.

- (i) $\Delta = \Omega(n^{1/2})$ on graphs which are not too tree-like.
- (ii) $\Delta = O(n^{1/2})$ on graphs where τ is polynomial in $\log n$ and where the chance of the random walk returning locally to its origin is bounded away from 1 (more precisely, where $a(G)/n$ is bounded above, for $a(G)$ as in Corollary 20).

The argument rests on the two lemmas below, concerning the random walk X_j on a regular graph, started uniformly. The first is a routine consequence of spectral theory; the second is a specialization of Theorems 5 ($\tau_4 \leq K\tau_1$) and 8 of [4].

LEMMA 16. For $1 \leq a \leq b$,

$$\sum_{m=a}^b P(X_m = X_0) \leq b/n + K\tau n e^{-a/\tau}.$$

LEMMA 17. For all initial distributions ρ and all $B \subset G$,

$$P_\rho(T_B > K\tau n \log(n)/|B|) \leq \frac{1}{2}.$$

Proof of Theorem 15. As usual, we consider the spanning tree $\mathcal{T}$ constructed by the random walk X_j , as in Proposition 1. Suppose X_0 is uniform on G . Write $d(v, w)$ for the distance in $\mathcal{T}$ from v to w .

To establish the lower bound, fix $1 \leq a < b$. Let Z be the number of pairs (l, m) such that

$$0 \leq l < m \leq b, \quad m - l > a, \quad X_m = X_l.$$

If $Z = 0$ then, for each $i \leq b$, the path in $\mathcal{T}$ from X_i to X_0 starts with an edge (X_i, X_j) for some $i > j \geq i - a$. So $d(X_b, X_0) \geq b/a$. Thus

$$\begin{aligned} P(\Delta < b/a) &\leq P(d(X_b, X_0) < b/a) \\ &\leq P(Z > 0) \\ (11) \quad &\leq EZ \\ &\leq b \sum_{m=a}^b P(X_m = X_0) \\ &\leq b^2/n + bK\tau n e^{-a/\tau} \text{ using Lemma 16.} \end{aligned}$$

To prove the lower bound we may suppose that n is arbitrarily large and that $\tau = o(n^{1/2}/\log n)$. Put $b =$ the integer part of $n^{1/2}/2$, and $a =$ the integer next larger than $\tau \log(2n^{3/2}K\tau)$. Then for sufficiently large n we have $1 \leq a < b$ and $a \leq K\tau \log n$. Then inequality (11) gives

$$P(\Delta < b/a) \leq \frac{1}{2}$$

and so

$$\begin{aligned} E\Delta &\geq (1/2)(b/a) \\ &\geq n^{1/2}/(K\tau \log n) \end{aligned}$$

which is the desired lower bound.

To establish the upper bound, fix v_0 . Let $v_0, v_1, \dots, v_c$ be a path in G . Let B be the set $\{v_0, \dots, v_c\}$. Let B_p be the event “the path in $\mathcal{T}$ from v_0 to the root X_0 starts out as $v_0, v_1, \dots, v_c$.” The argument rests on the following remarkable identity, whose proof we defer. Recall $T_B^+ = \min \{n \geq 1 : X_n \in B\}$.

LEMMA 18.

$$P(T_{v_c} = l | B_p) = \frac{P_{v_c}(T_B^+ > l)}{E_{v_c} T_B^+}, \quad l = 0, 1, 2, \dots.$$

Now define

$$s = \text{integer next larger than } K\tau n \log n/(c+1).$$

Then Lemma 17 shows

$$P_\rho(T_B \geq s) \leq \frac{1}{2} \text{ for all distributions } \rho.$$

Then by iterating,

$$P_{v_c}(T_B^+ > js) \leq (1/2)^{j-1} P_{v_c}(T_B^+ > s)$$

for $j = 1, 2, \dots$. Applying Lemma 18 with $l = js$ and again with $l = s$,

$$\begin{aligned} P(T_{v_c} = js | B_p) &\leq (1/2)^{j-1} P(T_{v_c} = s | B_p) \\ &\leq (1/2)^{j-1} s^{-1} \end{aligned}$$

using the fact that the right quantity in Lemma 18, and hence the left quantity, is decreasing in l . Using this decreasing property again,

$$P(js \leq T_{v_c} < (j+1)s | B_p) \leq (1/2)^{j-1}$$

and so, summing,

$$P(js \leq T_{v_c} | B_p) \leq (1/2)^{j-2}.$$

Given event B_p occurs, then $d(v_0, X_0) = c + d(v_c, X_0) \leq c + T_{v_c}$. Conversely, if $d(v_0, X_0) > c$ then B_p must occur for some unique path $v_0, \dots, v_c$. Therefore, we obtain

$$P(d(v_0, X_0) > c + js) \leq (1/2)^{j-2}.$$

But $\Delta/2 \leq \max_v d(v, X_0)$, and so

$$P(\Delta/2 > c + js) \leq n(1/2)^{j-2}.$$

It is now routine to show

$$\begin{aligned} E\Delta &\leq 2c + Ks \log n \\ &\leq 2c + K\tau n \log^2 n/c \end{aligned}$$

from the definition of s . Choosing $c = \text{integer part of } \tau^{1/2} n^{1/2} \log n$ gives the desired upper bound.

Proof of Lemma 18. We may write

$$\{T_{v_c} = l\} \cap B_p = \{T_{v_c} = l = T_B\} \cap \left(\bigcap_{i=0}^{c-1} D_i \right),$$

where

$$D_i = \{T_{\{v_0, v_1, \dots, v_i\}} = T_{v_i}, X(T_{v_i} - 1) = v_{i+1}\}.$$

The key fact is that, using the Markov property, the conditional probability $P(\cap D_i | T_{v_c} = T_B = l)$ does not depend on l . It follows that we can write

$$P(T_{v_c} = l | B_p) = \alpha P(T_{v_c} = T_B = l), \quad l = 0, 1, 2, \dots$$

where α does not depend on l . But

$$\begin{aligned}
 P(T_{v_c} = T_B = l) &= \sum_w P(X_0 = w, T_{v_c} = T_B = l) \\
 &= \sum_w P(X_0 = w, X_l = v_c, X_i \in G \setminus B \text{ for } 0 \leq i \leq l-1) \\
 &= \sum_w P(X_0 = v_c, X_l = w, X_i \in G \setminus B \text{ for } 1 \leq i \leq l) \\
 &= P_{v_c}(T_B^+ > l)
 \end{aligned}$$

where the third identity uses time-reversal for the stationary random walk. This establishes the lemma, up to the normalizing constant in the denominator, but this must be

$$\sum_{l=0}^{\infty} P_{v_c}(T_B^+ > l) = E_{v_c} T_B^+.$$

7. Counting spanning trees. The matrix tree theorem gives a complicated expression for the number $N(G)$ of spanning trees in an arbitrary graph G . For r -regular G , there is an expression involving only the eigenvalues $(\lambda'_2, \dots, \lambda'_n)$ of the adjacency matrix ([11, Prop. 1.4 and § 7.6]), which can then be expressed in terms of the eigenvalues $(\lambda_2, \dots, \lambda_n)$ of P , as follows.

$$(12) \quad N(G) = n^{-1} r^{n-1} \prod_{i=2}^n (1 - \lambda_i).$$

McKay [17] gives good upper bounds on $N(G)$, implicitly using random walk ideas, and in [16] discusses the case of random regular graphs.

Proposition 1 yields a probabilistic alternative to (12): pick some simple tree t and calculate the probability that the random walk construction gives t . The simplest tree is a path.

PROPOSITION 19. *Suppose $v_1, v_2, \dots, v_n$ is a Hamiltonian path in G . Let $B_i = \{v_1, \dots, v_i\}$. Let ρ_i be the probability that the random walk on G started at v_i will return to v_i before exiting B_i . Then*

$$N(G) = \prod_{i=1}^{n-1} r_{v_i} (1 - \rho_i).$$

Proof. Let t be the path $v_1, \dots, v_n$, considered as a tree rooted at v_1 . Use the random walk started at v_1 to construct the random tree $\mathcal{T}$. Then

$$P(\mathcal{T} = t) = \prod_{i=1}^{n-1} P_{v_i}(\text{exit } B_i \text{ along } (v_i, v_{i+1}))$$

where the “exit” event means that the random walk, when it first exits the subset B_i (which may or may not occur on the first step), does so along the edge (v_i, v_{i+1}) . Then Proposition 1 implies

$$(13) \quad N(G) = \prod_{i=1}^{n-1} 1/P_{v_i}(\text{exit } B_i \text{ along } (v_i, v_{i+1})).$$

Starting from v_i , the mean number of visits to v_i before exiting B_i is $1/(1 - \rho_i)$, and each visit has chance $1/r_{v_i}$ to be followed by traversal of (v_i, v_{i+1}) . So

$$P_{v_i}(\text{exit } B_i \text{ along } (v_i, v_{i+1})) = \frac{1}{r_{v_i}} \frac{1}{1 - \rho_i}$$

and the result follows.

It is not clear if Proposition 19 is ever useful for counting $N(G)$: presumably, if we have enough structure to calculate or bound the ρ_i then we have enough structure to calculate or bound the eigenvalues and use (12), at least in the regular case. On the other hand, having two conceptually different ways of calculating the same quantity is, throughout mathematics, often useful for some purpose or other. We end by pointing out two consequences of (12) and Proposition 19.

Markov chain theory shows that on a regular n -vertex graph G the following are equal—call the common value $a(G)$.

- (i) $\sum_{k=2}^n 1/(1 - \lambda_k)$ where λ_k are the eigenvalues of P .
- (ii) $n^{-1} \sum_w E_v T_w$ for each fixed v .
- (iii) $\sum_v \sum_{i=0}^{\infty} (P_v(X_i = v) - 1/n)$.

Moreover $a(G) \geq n - 1$, with equality if and only if G is the complete graph. Because of (ii) we interpret $a(G)$ as the “average first hitting time” for the random walk on G . Putting together (i), (12) and the “geometric mean $\leq$ arithmetic mean” inequality gives the following corollary.

COROLLARY 20. *For an r -regular graph G on n vertices,*

$$(N(G))^{1/(n-1)} \geq \frac{rn^{1/(n-1)}(n-1)}{a(G)}.$$

Thus we can relate $N(G)$ to a natural property of the random walk on G .

Our second observation concerns a specific graph, the $K \times K$ discrete torus G_K . Consider the growth exponent

$$c = \frac{1}{4} \lim_K (N(G_K))^{1/K^2}.$$

Using (12) and the well-known eigenvalues for the simple random walk on the discrete torus we obtain (c.f. [11, § 7.8, Exercise 10])

$$c = \exp \left(\int_0^1 \int_0^1 \log (\sin^2 \pi x + \sin^2 \pi y) \, dx \, dy \right) \approx 0.825.$$

On the other hand, by applying Proposition 19 with the natural Hamiltonian path

$$(1, 1), (1, 2), \dots, (1, K), (2, K), (2, K-1), \dots, (2, 1), (3, 1), (3, 2), \dots,$$

and approximating the random walk on G_K by simple symmetric random walk Y_j on the infinite square lattice $\mathbf{Z}^2$, we can obtain the following expression for c .

$$c = P_{(0,0)}(Y_j \text{ does not return to } (0,0) \text{ before exiting } B), \text{ where}$$

$$B = \{(i, j) : j < 0\} \cup \{(i, 0) : i \leq 0\}.$$

Thus we have used spanning trees to evaluate a (nontrivial) probability associated with two-dimensional random walk!

Acknowledgments. This work arose in conversation with Persi Diaconis, who as always deserves thanks. Thanks also to Andrei Broder and Umesh Vazirani for helpful discussions.

REFERENCES

- [1] D. J. ALDOUS, *The continuum random tree I*, Ann. Probab., to appear.
- [2] ———, *Hitting times for random walks on vertex-transitive graphs*, Math. Proc. Cambridge Phil. Soc., 106 (1989), pp. 179–191.
- [3] ———, *Lower bounds for covering times for reversible Markov chains and random walks on graphs*, J. Theoretical Probab., 2 (1989), pp. 91–100.
- [4] ———, *Some inequalities for reversible Markov chains*, J. London Math. Soc., 25 (1982), pp. 564–576.
- [5] R. ALELIUNAS, R. M. KARP, R. J. LIPTON, L. LOVASZ, AND C. RACKOFF, *Random walks, universal traversal sequences, and the complexity of maze traversal*, in proc. 20th IEEE Found. Comp. Sci. Symp., Institute for Electrical and Electronics Engineers, New York, 1979, pp. 218–233.
- [6] V. ANANTHARAM AND P. TSOUKAS, *A proof of the Markov chain tree theorem*, Stat. Probab. Letters, 8 (1989), pp. 189–192.
- [7] N. BIGGS, *Algebraic Graph Theory*, Cambridge University Press, Cambridge, 1974.
- [8] A. BRODER, *Generating random spanning trees*, in Symp. Foundations of Computer Sci., Institute for Electrical and Electronics Engineers, New York, 1989, pp. 442–447.
- [9] A. BRODER AND A. R. KARLIN, *Bounds on the cover time*, J. Theoretical Probab., 2 (1989), pp. 101–120.
- [10] A. E. BROUWER, A. M. COHEN, AND A. NEUMAIER, *Distance Regular Graphs*, Springer-Verlag, Berlin, New York, 1989.
- [11] D. M. CVETKOVIC, M. DOOB, AND H. SACHS, *Spectra of Graphs*, Academic Press, New York, 1980.
- [12] R. J. FOWLER, *Analyzing a simple stochastic process to derive the distribution of vertex depths in random rooted labelled trees*, Technical Report, Computer Science Department, University of Rochester, Rochester, NY, 1988.
- [13] J. R. GRIGGS, D. J. KLEITMAN, AND A. SHASTRI, *Spanning trees with many leaves in cubic graphs*, J. Graph Theory, 13 (1989), pp. 669–695.
- [14] J. D. KAHN, N. LINIAL, N. NISAN, AND M. E. SAKS, *On the cover time of random walks on graphs*, J. Theoretical Probab., 2 (1989), pp. 121–128.
- [15] D. E. KNUTH, *The Art of Computer Programming*, Volume 1, Addison-Wesley, Reading, MA, 1968.
- [16] B. D. MCKAY, *Spanning trees in random regular graphs*, in Proc. Third Caribbean Conference on Combinatorics, Univ. West Indies, 1981.
- [17] ———, *Spanning trees in regular graphs*, Europ. J. Combinatorics, 4 (1983), pp. 149–160.
- [18] J. W. MOON, *Counting Labelled Trees*, Canadian Mathematical Congress, Vancouver, 1970.
- [19] R. PEMANTLE, *Choosing a spanning tree for the integer lattice uniformly*, Technical Report, Cornell University, Ithaca, NY, 1989.
- [20] A. SINCLAIR AND M. JERRUM, *Approximate counting, uniform generation and rapidly mixing Markov chains*, Inform. Comput., 82 (1989), pp. 93–133.
- [21] D. E. SYMER, *Expanded ergodic Markov chains and cycling systems*, senior thesis, Dartmouth College, Hanover, New Hampshire, 1984.
- [22] G. SZEKERES, *Distribution of labelled trees by diameter*, in Combinatorial Mathematics X, Lecture Notes in Mathematics 1036, Springer-Verlag, Berlin, New York, 1982, pp. 392–397.

A CHARACTERIZATION OF AN EXTENSION OF THE BANZHAF VALUE TO MULTICANDIDATE VOTING GAMES*

E. M. BOLGER†

Abstract. A family of power indices for r -candidate voting games is studied and it is determined which members of this family are dummy-independent. It is then shown that an extension of the absolute Banzhaf index is the only such dummy-independent power index preserved under composition of games.

Key words. Banzhaf index, voting games, composition of games

AMS(MOS) subject classification. 90D

1. Introduction. Simple games serve as models for elections in which a group of voters is to elect one of two candidates. We will consider elections in which a group of voters is to elect one of several candidates. We model such elections by “voting games among r candidates.”

If precisely one of r candidates is to be elected, then the voters must partition themselves among the r candidates. The voting rules may be specified by stating, for each partition $\{S_1, S_2, \dots, S_r\}$ of the voters, whether each S_j is a winning or a losing coalition with respect to this partition. We formalize this in the following definition.

DEFINITION 1. Let N be a finite set. A voting game on N among r candidates is a listing of all the partitions of N which consist of r or fewer subsets together with a rule which specifies, for each subset in such a partition, whether that subset wins or loses with respect to the given partition. The elements of N are called the voters; subsets of N are called coalitions of voters.

Example 1. Let $N = \{1, 2, 3, 4\}$. Consider the 3-candidate voting game in which a coalition of voters is winning if it contains voter 1 and has more members than any other coalition. That is, all 3- or 4-voter coalitions which contain voter 1 are winning. Moreover, voter 1's coalition wins in each of the following partitions: $\{\{1, 2\}, \{3\}, \{4\}\}$, $\{\{1, 3\}, \{2\}, \{4\}\}$, $\{\{1, 4\}, \{2\}, \{3\}\}$.

For 2-candidate voting games, voter j is said to *pivot* with respect to coalition S if S wins and $S - \{j\}$ loses.

Since the early 1950s, mathematicians and political scientists have used “power indices” based on the notion of pivot to study the a priori voting power inherent in a voting system. (The intent is to ignore political factors such as the particular beliefs of the current members of a political body.)

As Banzhaf [1965] pointed out, a legislator's voting power is measured by his ability—by his vote—to affect the passage or defeat of an issue. Thus, Banzhaf proposed that the voting power of an individual voter should be proportional to the number of ways this voter can change the outcome by changing his vote.

For a 2-candidate voting game, the raw Banzhaf index for voter j is the number of pivots for voter j . If these numbers are normalized to sum to 1, we obtain the usual Banzhaf indices; if, instead, the numbers of pivots are divided by 2^{n-1} , we obtain the absolute Banzhaf indices.

The theoretical properties of the Banzhaf index for 2-candidate voting games have been widely studied. See, for example, Brams [1975], Straffin [1977], Owen [1978], Dubey and Shapley [1979], Holzman, Lehrer, and Linial [1987].

* Received by the editors June 13, 1988; accepted for publication (in revised form) December 18, 1989.

† Department of Mathematics and Statistics, Miami University, Oxford, Ohio 45056.

Power indices have been applied in studying the distribution of power in major voting bodies. See, for example, Shapley and Shubik [1954], Mann and Shapley [1962], Riker and Shapley [1968], Riker and Ordeshook [1973, Chap. 6], Owen [1975], Lucas [1976], Owen and Carreras [1988], Brams, Affuso, and Kilgour [1988], Brams [1988].

Furthermore, the Banzhaf index for 2-candidate voting games has been used in several court cases in the state of New York as the appropriate measure of legislator power and as the criterion of fair representation in weighted voting schemes. See, for example, Johnson [1969], Papayanopoulos [1973], Imrie [1973], Grofman and Scarrow [1979], Lucas et al. [1982].

First we shall generalize the concept of pivot to r -candidate voting games. To do so, it is convenient to introduce some terminology and notation.

DEFINITION 2. Let Γ be a partition of N and let $S \in \Gamma$. Following Myerson [1977], we call (S, Γ) an *embedded coalition* and denote the collection of all embedded coalitions by ECL. The number of nonempty subsets in Γ shall be denoted by $|\Gamma|$. $\Gamma(i)$ denotes the member of Γ to which i belongs.

If v is a voting game among r candidates and if (S, Γ) is an embedded coalition with $|\Gamma| \leq r$, we set $v(S, \Gamma) = 1$ if S wins with respect to Γ whereas $v(S, \Gamma) = 0$ if S loses with respect to Γ . For $S \subset N$, $|S|$ denotes the number of players in S . We also put $n = |N|$.

DEFINITION 3. Let $i \in N$. Let $(S, \Gamma) \in \text{ECL}$ be such that $|\Gamma| \leq r$ and $i \in S$. Consider a partition Γ' obtained by moving player i from S to some other (possibly empty) member T of Γ . The mapping $\alpha_{iT}: \Gamma \rightarrow \Gamma'$ defined by

$$\alpha_{iT}(S) = S - \{i\},$$

$$\alpha_{iT}(T) = T \cup \{i\},$$

$$\alpha_{iT}(S^*) = S^* \text{ for all other } S^* \in \Gamma$$

is called a *move* for player i . Such a move is called a *positive move* if S wins with respect to Γ and $S - \{i\}$ loses with respect to Γ' , whereas a move is called *negative* if S loses and $S - \{i\}$ wins.

In Example 1 above, the partition $\{\{1, 2\}, \{3\}, \{4\}\}$ yields two positive moves for each of players 1 and 2 whereas the partition $\{\{1, 2, 3\}, \{4\}, \emptyset\}$ yields two positive moves for player 1 and one positive move for each of voters 2 and 3. In all, voter 1 has 14 positive moves whereas voter 2 has 4 positive moves. In this example, as in most voting games, there are no negative moves. (However, it is desirable, for theoretical reasons, to also consider games in which the addition of a player to a winning coalition causes the new coalition to lose.)

The best known power indices for simple (2-candidate) games are based on the concept of pivot and can be written in the form

$$\phi_i(v) = \sum_{\{S \subset N: i \in S\}} f(|S|, n)[v(S) - v(S - \{i\})].$$

In this paper we will consider power indices of the form

$$(1) \quad \psi(i, v) = \sum_{\Gamma} a(|\Gamma(i)|, n, |\Gamma|) \sum_{\substack{T \in \Gamma \\ T \neq \Gamma(i)}} [v(\Gamma(i), \Gamma) - v(\Gamma(i) - \{i\}, \alpha_{iT}(\Gamma))]$$

where the sum is over all partitions of N into exactly r subsets. Note that the inner sum gives the difference between the number of positive moves and the number of negative moves for voter i in a partition Γ . The function $a(|\Gamma(i)|, n, |\Gamma|)$ will also, in general,

depend on the value of r . However, since r is to be held fixed, this dependence on r will, for simplicity of notation, be suppressed.

An extension of the Shapley–Shubik index and an extension of the Banzhaf index, each of which is of the above form, are discussed in Bolger [1986]. A different extension of the Shapley value appears in Myerson [1977].

2. Dummy-independent power indices.

DEFINITION 4. Let v be a voting game on N among r candidates. Voter j is called a *dummy* if voter j has neither any positive moves nor any negative moves in the game.

If player j is a dummy, then $\psi(j, v) = 0$. Moreover, since the addition or removal of a dummy player has no effect on winning or losing coalitions, it may be reasonable to assume that the addition or removal of a dummy voter from the game should have no effect on the power indices of the remaining voters.

DEFINITION 5. Let v be a voting game on N among r candidates and let Γ be a partition of N . Let $d \notin N$. We obtain partitions Γ^d of $N \cup \{d\}$ from Γ by placing d into an existing set in Γ and if $|\Gamma| < r$ we can also place d into a previously empty set. In fact, all partitions of $N \cup \{d\}$ can be obtained in this manner. The *dummy extension* of v by d is defined by

$$v^d(S, \Gamma^d) = v(S - \{d\}, \Gamma) \quad \text{for each } S \in \Gamma^d.$$

DEFINITION 6. A power index ψ on the class of voting games among r candidates is called *dummy-independent* if

$$\psi(i, v^d) = \psi(i, v) \quad \text{for each } i \in N.$$

Both the Shapley–Shubik and the absolute Banzhaf indices for 2-candidate voting games are dummy-independent. (See Owen [1978].) Our first goal is to determine which power indices of form (1) (for r -candidate voting games) are dummy-independent.

LEMMA 1. If ψ is dummy-independent (DI) and if $1 \leq t \leq n - m$, then

(a) for $m < r - 1$,

$$a(t, n, m + 1) = a(t + 1, n + 1, m + 1) + ma(t, n + 1, m + 1) + a(t, n + 1, m + 2)$$

and

(b) for $m = r - 1$,

$$a(t, n, m + 1) = a(t + 1, n + 1, m + 1) + (r - 1)a(t, n + 1, m + 1).$$

Proof. Let Γ be any partition of N . Let $S \in \Gamma$. Without loss of generality, we can assume S consists of the last $|S|$ players in N ; i.e.,

$$S = \{n - |S| + 1, \dots, n - 1, n\}.$$

Suppose there are k ($k \geq 0$) sets in $\Gamma - \{S\}$ which contain more than one element each. Denote these sets by $T_1, T_2, \dots, T_k$. Further suppose there are j ($j \geq 0$) one element sets, say $\{1\}, \{2\}, \dots, \{j\}$. Then, let $m = j + k$ and

$$\Gamma = \{\{n - |S| + 1, \dots, n - 1, n\}, T_1, \dots, T_k, \{1\}, \dots, \{m - k\}, \phi, \dots, \phi\}$$

so that Γ has exactly r sets. Note that

$$|\Gamma| = m + 1.$$

Now let $v^{S, \Gamma}$ denote the r -candidate voting game in which Γ is the only partition that contains a winning set and in which S is the *only* winning set. That is,

$$v^{S, \Gamma}(S^*, \Gamma^*) = \begin{cases} 1 & \text{if } (S^*, \Gamma^*) = (S, \Gamma) \\ 0 & \text{otherwise.} \end{cases}$$

Then

$$\psi(n, v^{S, \Gamma}) = (r-1)a(|S|, n, m+1).$$

Let v^d be obtained from $v^{S, \Gamma}$ by adding a dummy player, d , to $v^{S, \Gamma}$. The partitions that contain a winning set (with the winning set listed first) are:

$$\begin{aligned} & \{ \{n - |S| + 1, \dots, n, d\}, T_1, \dots, T_k, \{1\}, \dots, \{m-k\}, \phi, \dots, \phi \} \\ & \{ \{n - |S| + 1, \dots, n\}, T_1 \cup \{d\}, \dots, T_k, \{1\}, \dots, \{m-k\}, \phi, \dots, \phi \} \\ & \quad \vdots \quad \quad \quad \vdots \\ & \{ \{n - |S| + 1, \dots, n\}, T_1, \dots, T_k \cup \{d\}, \{1\}, \dots, \{m-k\}, \phi, \dots, \phi \} \\ & \{ \{n - |S| + 1, \dots, n\}, T_1, \dots, T_k, \{1, d\}, \dots, \{m-k\}, \phi, \dots, \phi \} \\ & \quad \vdots \quad \quad \quad \vdots \\ & \{ \{n - |S| + 1, \dots, n\}, T_1, \dots, T_k, \{1\}, \dots, \{m-k, d\}, \phi, \dots, \phi \} \end{aligned}$$

and if $m+1 < r$

$$\{ \{n - |S| + 1, \dots, n\}, T_1, \dots, T_k, \{1\}, \dots, \{m-k\}, \{d\}, \phi, \dots, \phi \}.$$

Then if $m+1 < r$,

$$\begin{aligned} \psi(n, v^d) &= a(|S| + 1, n+1, m+1) \cdot (r-1) \\ &\quad + ma(|S|, n+1, m+1)(r-1) + a(|S|, n+1, m+2)(r-1) \end{aligned}$$

whereas for $m+1 = r$,

$$\psi(n, v^d) = a(|S| + 1, n+1, r) \cdot (r-1) + (r-1)^2 a(|S|, n+1, r).$$

The result follows immediately.

We now prove the converse of Lemma 1. The converse will first be proved for the game $v^{S, \Gamma}$, the collection of which form a “basis” for the class of all r -candidate voting games.

LEMMA 2. ψ is DI on the class of games $\{v^{S, \Gamma}: (S, \Gamma) \in \text{ECL}\}$ if and only if (a) and (b) of Lemma 1 are satisfied.

Proof. It remains to prove that if (a) and (b) are satisfied then ψ is DI on the above class. With notation as in the proof of Lemma 1, it follows that

$$\psi(1, v^{S, \Gamma}) = -(r-m)a(|S| + 1, n, m),$$

whereas for $m-k < j \leq n - |S|$,

$$\psi(j, v^{S, \Gamma}) = -a(|S| + 1, n, m+1).$$

Further, in the case $m+1 = r$, we get

$$\psi(1, v^d) = -a(|S| + 2, n+1, r-1) - a(|S| + 1, n+1, r) - (r-2)a(|S| + 1, n+1, r-1)$$

and for $m-k < j \leq n - |S|$,

$$\psi(j, v^d) = -a(|S| + 2, n+1, r) - (r-1)a(|S| + 1, n+1, r).$$

For the case $m+1 < r$,

$$\begin{aligned} \psi(1, v^d) &= -a(|S| + 2, n+1, m)(r-m) \\ &\quad - (m-1)a(|S| + 1, n+1, m)(r-m) - a(|S| + 1, n+1, m+1)(r-m) \end{aligned}$$

and for $m - k < j \leq n - |S|$,

$$\begin{aligned}\psi(j, v^d) = & -a(|S| + 2, n + 1, m + 1) - ma(|S| + 1, n + 1, m + 1) \\ & - a(|S| + 1, n + 1, m + 2).\end{aligned}$$

It is easy to check directly that (a) and (b) imply

$$\psi(1, v^d) = \psi(1, v^{S, \Gamma})$$

and

$$\psi(j, v^d) = \psi(j, v^{S, \Gamma}).$$

In order to extend Lemma 2 to the class of r -candidate voting games, we shall use the collection $\{v^{S, \Gamma}\}$ as a "basis."

DEFINITION 7. Let v and w be r -candidate voting games on N . Then we define $v \vee w$ and $v \wedge w$ by

$$v \vee w(S, \Gamma) = \max \{v(S, \Gamma), w(S, \Gamma)\}$$

and

$$v \wedge w(S, \Gamma) = \min \{v(S, \Gamma), w(S, \Gamma)\}.$$

Now let v be an r -candidate voting game on N and let the collection of distinct winning ECL's for v be $\{(S_1, \Gamma_1), \dots, (S_t, \Gamma_t)\}$. Then we may write

$$v = v^{S_1, \Gamma_1} \vee \dots \vee v^{S_t, \Gamma_t}.$$

Noting that

$$v^{S_b, \Gamma_i} \wedge v^{S_b, \Gamma_j} = 0,$$

we have

$$\psi(v) = \psi(v^{S_1, \Gamma_1}) + \dots + \psi(v^{S_t, \Gamma_t}).$$

Further, if v^d is a dummy extension of v ,

$$v^d = (v^{S_1, \Gamma_1})^d \vee \dots \vee (v^{S_t, \Gamma_t})^d.$$

Then, using Lemma 2, we can conclude that the following theorem holds.

THEOREM 1. ψ is DI on the class of all voting games among r candidates if and only if (a) and (b) of Lemma 1 are satisfied.

Examples. An extension of the Shapley-Shubik index is obtained from $a(t, n, m) = (t-1)!(n-t)!(r-2)!/(n!(r-1)^{n-t}(r-m)!)$; an extension of the absolute Banzhaf index is obtained from $a(t, n, m) = (r-2)!/(r^{n-1}(r-m)!)$. Each of these is DI.

COROLLARY 1. If ψ is DI, then

$$a(n+1, n+1, 1) = a(1, 1, 1) - \sum_{i=1}^n a(i, i+1, 2).$$

Proof. The proof is obtained by putting $m = 0$ in (a).

COROLLARY 2. If ψ is DI, then

$$a(t, r+k, r) = \frac{1}{(r-1)^{k+1-t}} \sum_{i=t}^{k+1} (-1)^{i-t} \binom{k+1-t}{i-t} a(i, r+i-1, r)$$

provided $k \geq 1$ and $t \leq k$.

Proof. The proof is obtained by using (b) and induction on $k - t$.

COROLLARY 3. *If ψ is DI then for $0 \leq p \leq r - 2$, $t \leq k + p$,*

$$a(t, r+k, r-p) = \sum_{j=0}^p \sum_{l=1}^{j+1} \frac{(r-p-1+j)}{(-1)^{l-1}(l-1)!(j+1-l)!(r-p-l+j)^{k+p+2-j-l}} \cdot \sum_{i=t}^{k+p+1-j} (-1)^{i-t} \binom{k+p+1-t-j}{i-t} a(i, r+i-p-1+j, r-p+j).$$

Proof. Although the proof is fairly easy, it is rather long and messy. Consequently, it is omitted. (The proof of Corollary 3 as well as the proof of Lemma 7, which appears near the end of the paper, are presented in a separate appendix which may be obtained by writing to the author.)

Remark. It follows from the above corollaries that the values of $a(t, k, m)$ for $k \leq n$ and $t \leq k - m$ can be determined from the values of $a(i, k, k - i + 1)$ for $k \leq n$ and $i \leq k$. For example, for $r = 4$ and $n \leq 5$, we have to specify $a(1, 1, 1)$, $a(1, 2, 2)$, $a(1, 3, 3)$, $a(2, 3, 2)$, $a(1, 4, 4)$, $a(2, 4, 3)$, $a(3, 4, 2)$, $a(2, 5, 4)$, $a(3, 5, 3)$, $a(4, 5, 2)$. The rest, including $a(1, 5, 2)$, $a(2, 5, 2)$, $a(3, 5, 2)$, $a(1, 5, 3)$, $a(2, 5, 3)$, $a(1, 5, 4)$, $\dots$ can then be determined from the corollaries. In particular,

$$a(1, 5, 3) = \frac{1}{4}[a(1, 3, 3) - 2a(2, 4, 3) + a(3, 5, 3)] - \frac{5}{12}[a(1, 4, 4) - a(2, 5, 4)].$$

3. Behavior of the power indices under composition of games. Owen [1978] proved that the absolute Banzhaf value is preserved under composition of games in characteristic function form. In this section we show that there is at most one dummy-independent power index ψ which satisfies a composition axiom for games in partition function form.

The intuitive idea is to replace player n by a committee consisting of player n and two new voters. This committee will cast the vote (or votes) originally assigned to voter n . We will require that the power index of player n be changed by a factor equal to the power of player n within the committee and that the power of players $1, 2, \dots, n - 1$ remain unchanged. Several definitions will be stated to formalize these ideas.

DEFINITION 8. Let Γ be a partition of N and let $M \subset N$. Then Γ/M is the restriction of Γ to M . That is, if

$$\Gamma = \{S_1, S_2, \dots, S_q\}, \quad \text{then } \Gamma/M = \{S_1 \cap M, \dots, S_q \cap M\}.$$

DEFINITION 9. An r -candidate voting game is called *decisive* (or constant-sum) if for each partition Γ there is precisely one set $S \in \Gamma$ such that S wins with respect to Γ .

Example 2. We shall let Δ_{123} denote the simple majority game on three voters with the tiebreaker that voter 1's set wins in $\{\{1\}, \{2\}, \{3\}\}$. It is easy to check that $\psi(1, \Delta_{123}) = 2a(2, 3, 2)(r-1) + a(1, 3, 3)(r-1)$; $\psi(2, \Delta_{123}) = ra(2, 3, 2)$.

DEFINITION 10. Let v be an r -candidate voting game with player set N . Let $w_1, w_2, \dots, w_n$ be decisive games with pairwise disjoint player sets $M_1, \dots, M_n$. Let $\Gamma = \{S_1, S_2, \dots, S_q\}$ be a partition of $M_1 \cup M_2 \cup \dots \cup M_n$. For $i = 1, \dots, n$ and $j = 1$ to q , form the set T_j :

$$i \in T_j \Leftrightarrow w_i(S_j \cap M_i, \Gamma/M_i) = 1.$$

Then we say that S_1 wins with respect to Γ in the *composed game* $\mu = v[w_1, \dots, w_n] \Leftrightarrow T_1$ wins with respect to $\{T_1, T_2, \dots, T_q\}$ in v .

Example 3. Let $v_{(j)}$ denote the one-person game in which $\{j\}$ wins and let v be the unanimity game in which only the grand coalition $\{1, 2, \dots, n\}$ wins. In the com-

posed game $v[v_{(1)}, \dots, v_{(n-1)}, \Delta_{n,n+1,n+2}]$, the following partitions contain a winning set (in each case the winning set is listed first):

$$\begin{aligned} & \{ \{1, 2, \dots, n, n+1, n+2\}, \phi \}, \quad \{ \{1, 2, \dots, n, n+1\}, \{n+2\} \}, \\ & \{ \{1, 2, \dots, n, n+2\}, \{n+1\} \}, \quad \{ \{1, 2, \dots, n-1, n+1, n+2\}, \{n\} \}, \\ & \{ \{1, 2, \dots, n\}, \{n+1\}, \{n+2\} \}. \quad \text{Then,} \end{aligned}$$

$$\psi(i, v) = (r-1)a(n, n, 1);$$

$$\psi(1, u) = (r-1)[a(n+2, n+2, 1) + 3a(n+1, n+2, 2) + a(n, n+2, 3)];$$

$$\psi(n, u) = (r-1)[2a(n+1, n+2, 2) + a(n, n+2, 3)];$$

$$\psi(n+1, u) = ra(n+1, n+2, 2).$$

DEFINITION 11. A voting game is called *monotonic* if whenever S wins with respect to partition $\Gamma = \{S, T_1, \dots, T_m\}$, then $S \cup S'$ wins with respect to

$$\Gamma' = \{S \cup S', T_1 - S', \dots, T_m - S'\}.$$

Remark. In the following we will require that $a(t, n, m) > 0$ if $m \leq r$ and $t \leq n - m + 1$. This will guarantee that, for monotonic games, $\psi(i, v) > 0$ if i is not a dummy.

Notation. $v_{(1,2,\dots,n)}$ will denote the game on $N = \{1, 2, \dots, n\}$ in which only the grand coalition N wins.

We will require that the power index ψ satisfies the following composition axiom.

AXIOM. Let v be an r -candidate voting game and let

$$u = v[v_{(1)}, \dots, v_{(j-1)}, w_j, v_{(j+1)}, \dots, v_{(n)}]$$

where w_j is either $v_{(j)}$ or $\Delta_{j,n+1,n+2}$ or $\Delta_{n+1,j,n+2}$. Then

$$(a) \text{ for } i \neq j, \psi(i, u) = \psi(i, v)$$

and

$$(b) \psi(j, u) = \psi(j, v)\psi(j, w_j).$$

LEMMA 3. If ψ satisfies the axiom, then $a(1, 1, 1) = 1/(r-1)$.

Proof. Let $v = v_{(1,2)}$ and let $u = v[v_{(1)}, v_{(2)}]$. Then $u = v$. Letting $j = 2$ we get

$$\psi(2, u) = \psi(2, u)\psi(2, v_{(2)})$$

so that $\psi(2, v_{(2)}) = 1$. But $\psi(2, v_{(2)}) = a(1, 1, 1)(r-1)$.

Remark. It may seem artificial to allow w_j to equal $v_{(j)}$ since this amounts to replacing player j by player j . If we were to require that w_j be different from $v_{(j)}$, it would turn out that there would be a one-parameter family of power indices ψ which are DI and that satisfy the axiom, rather than a unique power index.

LEMMA 4. If ψ is DI and satisfies the axiom, then

$$(2) \quad a(n, n+1, 2) = 2a(n+1, n+2, 2) + a(n, n+2, 3),$$

$$(3) \quad 2a(n+1, n+2, 2) + a(n, n+2, 3) = a(n, n, 1)[2a(2, 3, 2) + a(1, 3, 3)](r-1),$$

and

$$(4) \quad a(n+1, n+2, 2) = a(n, n, 1)a(2, 3, 2)(r-1).$$

Proof. Let $v = v_{(1,2,\dots,n)}$. First let $u_1 = v[v_{(1)}, \dots, v_{(n-1)}, \Delta_{n,n+1,n+2}]$. Then from Examples 1 and 2,

$$\begin{aligned} \psi(1, u_1) &= \psi(1, v) \Leftrightarrow a(n, n, 1) \\ &= a(n+2, n+2, 1) + 3a(n+1, n+2, 2) + a(n, n+2, 3). \end{aligned}$$

Using Corollary 1, this reduces to (2) above. Further,

$$\psi(n, u_1) = \psi(n, v)\psi(n, \Delta_{n,n+1,n+2}) \Leftrightarrow (3)$$

is satisfied. Formula (4) comes from applying the axiom to player n in the game $u_2 = v[v_{(1)}, \dots, v_{(n-1)}, \Delta_{n+1,n,n+2}]$ after observing that $\psi(n, u_2) = \psi(n+1, u_1)$.

COROLLARY 4. *If ψ is DI and satisfies the axiom, then*

$$(5) \quad a(2, 3, 2) = [1 - (r-1)a(1, 2, 2)]a(1, 2, 2),$$

$$(6) \quad a(1, 3, 3) = [2(r-1)a(1, 2, 2) - 1]a(1, 2, 2).$$

Proof. Using (2) and (3) we get

$$(i) \quad a(1, 3, 3) = \frac{2(r-1)a(1, 2, 2) - 1}{1 - (r-1)a(1, 2, 2)} a(2, 3, 2),$$

$$(ii) \quad a(3, 4, 2) = [1 - (r-1)a(1, 2, 2) - (r-1)a(2, 3, 2)][2a(2, 3, 2) + a(1, 3, 3)].$$

From (3), we then have

$$(iii) \quad [1 - (r-1)a(1, 2, 2)]a(2, 3, 2) \\ = [1 - (r-1)a(1, 2, 2) - (r-1)a(2, 3, 2)][2a(2, 3, 2) + a(1, 3, 3)]$$

which reduces to

$$(iv) \quad a(1, 3, 3) = \frac{2(r-1)a^2(2, 3, 2) - [1 - (r-1)a(1, 2, 2)]a(2, 3, 2)}{[1 - (r-1)a(1, 2, 2) - (r-1)a(2, 3, 2)]}$$

Formula (5) is a direct consequence of (i) and (iv); (6) then follows directly from (iv) and (5).

LEMMA 5. *If ψ is DI and satisfies the axiom, then*

$$(7) \quad a(1, 3, 3) = \frac{(r-2)a^2(2, 3, 2)}{[1 - (r-1)a(1, 2, 2) - (r-1)a(2, 3, 2)]}.$$

Proof. Let $S = \{1\}$ and let $\Gamma = \{\{1\}, \{2\}, \{3\}\}$ be a partition of $\{1, 2, 3\}$. Let $v = v^{S, \Gamma}$ and let $u = v[v_{(1)}, v_{(2)}, \Delta_{435}]$. Applying the axiom, we get

$$(v) \quad a(3, 5, 3) = (r-2)a^2(2, 3, 2).$$

But from (2),

$$a(3, 4, 2) = 2a(4, 5, 2) + a(3, 5, 3).$$

Using (4) and (v), this becomes

$$(vi) \quad a(3, 4, 2) = 2[1 - (r-1)a(1, 2, 2) - (r-1)a(2, 3, 2)]a(2, 3, 2) + (r-2)a^2(2, 3, 2).$$

From (2) and (3),

$$a(3, 4, 2) = [1 - (r-1)a(1, 2, 2) - (r-1)a(2, 3, 2)][2a(2, 3, 2) + a(1, 3, 3)]$$

which, in conjunction with (vi), yields (7).

COROLLARY 5. *If ψ is DI and satisfies the axiom, then $a(1, 2, 2) = r^{-1}$; $a(2, 3, 2) = r^{-2}$; $a(1, 3, 3) = (r-2)r^{-2}$.*

Proof. These results follow directly from (5)–(7).

Remark. The values of ψ are now determined for all 3-person games. We will proceed now by induction to determine ψ for all n .

THEOREM 2. If ψ is DI and satisfies the axiom, then for $t \geq 2$,

$$a(t, k, k-t+1) = \frac{(r-2)!}{r^{k-1}[r-(k-t+1)]!}.$$

Proof. Assume inductively that the result is true for $k < n+2$. Then let $S = \{1, 2, \dots, t\}$ and $\Gamma = \{S, \{t+1\}, \{t+2\}, \dots, \{n\}\}$. Let $v = v^{S, \Gamma}$ and let $u = v[v_{(1)}, \dots, v_{(n-1)}, \Delta_{n+1, n, n+2}]$. Then a messy computation (for details, write to the author) yields

$$\begin{aligned} \psi(1, u) = (r-1) & \left[a(t, n, n-t+1) + \frac{1}{n-t} a(t+1, n+1, n-t+1) \right. \\ & \left. - \frac{n-t+1}{n-t} a(t+2, n+2, n-t+1) - \frac{1}{n-t} a(t+1, n+2, n-t+2) \right] \end{aligned}$$

and

$$\psi(n, u) = -ra(t+2, n+2, n-t+1).$$

Since

$$\begin{aligned} \psi(1, v) &= (r-1)a(t, n, n-t+1), \\ (8) \quad \psi(1, u) &= \psi(1, v) \Leftrightarrow a(t+1, n+2, n-t+2) \\ &= a(t+1, n+1, n-t+1) - (n-t+1)a(t+2, n+2, n-t+1). \end{aligned}$$

Also, $\psi(n, u) = \psi(n, v)\psi(n, \Delta_{n+1, n, n+2}) \Leftrightarrow$

$$\begin{aligned} (9) \quad a(t+2, n+2, n-t+1) &= [r-(n-t)]a(t+1, n, n-t)a(2, 3, 2) \\ &= [r-(n-t)]a(t+1, n, n-t)r^{-2}. \end{aligned}$$

Using the induction hypothesis in (9) and noting that (9) holds for $t \geq 1$,

$$\begin{aligned} a(t+2, n+2, n-t+1) &= \frac{[r-(n-t)](r-2)!}{r^{n-1}[r-(n-t)]!} \frac{1}{r^2} \\ &= \frac{(r-2)!}{r^{n+1}[r-(n-t+1)]!} \quad \text{for } t \geq 1. \end{aligned}$$

Another application of the induction hypothesis in (8) with $t = 1$ will then yield the value of $a(2, n+2, n+1)$.

To complete the characterization of ψ , we have yet to determine $a(1, k, k)$. Unfortunately, this seems to require a separate computation.

LEMMA 6. If ψ is DI and satisfies the axiom, then

$$a(1, k, k) = \frac{(r-2)!}{r^{k-1}(r-k)!}.$$

Proof. The proof was already proven for $k \leq 3$. With (S, Γ) as in the previous theorem, let for $t = 1$,

$$u = v[\Delta_{1, n+1, n+2}, v_{(2)}, \dots, v_{(n)}].$$

Then

$$\begin{aligned}\psi(1, u) &= \psi(1, v)\psi(1, \Delta_{1, n+1, n+2}) \Leftrightarrow \\ &= -\frac{1}{n-1}a(2, n+2, n+1) + \frac{2}{n-1}a(2, n+1, n) \\ &\quad - \left(1 + \frac{1}{n-1}\right)a(3, n+2, n) + \frac{1}{n-1}a(1, n+1, n+1) \\ &\quad + \left(1 - \frac{1}{n-1}\right)a(1, n, n) = a(1, n, n)\frac{r-1}{r}.\end{aligned}$$

Assuming inductively that the lemma is true for $k < n+1$ and using the previous theorem, we can determine $a(1, n+1, n+1)$.

This concludes the proof that there is at most one DI power index ψ that satisfies the composition axiom.

We turn now to the problem of showing that if we set

$$a(t, n, m) = \frac{(r-2)!}{r^{n-1}(r-m)!},$$

then the corresponding power index ψ' does satisfy the composition axiom. We will only present an outline of the proof.

Step 1. Let $v = v^{S, \Gamma}$. Let $u = v[v_{(1)}, \dots, v_{(n-1)}, \Delta_{n, n+1, n+2}]$. Prove that

$$\psi'(1, u) = \psi'(1, v)$$

and

$$\psi'(n, u) = \psi'(n, v)\psi'(n, \Delta_{n, n+1, n+2}).$$

Proof. We write $(S, \Gamma) =$

$$\left\{ \{n, n-1, \dots, n-|S|+1\}, \underbrace{T_1, \dots, T_k}_{|T_j| > 1}, \{1\}, \dots, \{m-k\}, \phi, \dots, \phi \right\}$$

and check directly that the axiom is satisfied.

Step 2. With v as above, let $u = v[v_{(1)}, \dots, v_{(n-1)}, \Delta_{n+1, n, n+2}]$ and prove directly that

$$\psi'(n, u) = \psi'(n, v)\psi'(n, \Delta_{n+1, n, n+2}).$$

The remaining steps are similar; we need to replace a player in S and check the axiom, and then replace a player in a partition set of cardinality greater than 1 and again check to see if the axiom is satisfied. This procedure proves the following lemma.

LEMMA 7. If $v = v^{S, \Gamma}$ and if

$$u = v[v_{(1)}, \dots, v_{(j-1)}, w_j, v_{(j+1)}, \dots, v_{(n)}]$$

where w_j is one of the games

$$v_{(j)}, \Delta_{j, n+1, n+2}, \Delta_{n+1, j, n+2},$$

then

$$\psi'(j, v)\psi'(j, w_j) = \psi'(j, u),$$

whereas for $i \neq j$, $\psi'(i, v) = \psi'(i, u)$.

THEOREM 3. ψ' satisfies the composition axiom.

Proof. Let v be an arbitrary r -candidate voting game on $\{1, 2, \dots, n\}$. We can write

$$v = v^{S_1, \Gamma_1} \vee \dots \vee v^{S_m, \Gamma_m}.$$

It is easy to see that

$$v[v_{(1)}, \dots, v_{(j-1)}, w_j, v_{(j+1)}, \dots, v_{(n)}] \\ = v^{S_1, \Gamma_1}[v_{(1)}, \dots, w_j, \dots, v_{(n)}] \vee \dots \vee v^{S_m, \Gamma_m}[v_{(1)}, \dots, w_j, \dots, v_{(n)}].$$

The proof then follows from the linearity of ψ' and from Lemma 7.

4. Conclusion. We have thus shown that there is a unique power index ψ' of the form (1) which is dummy-independent and satisfies the composition axiom for the class of all r -candidate voting games. The index is an extension of the absolute Banzhaf index for 2-candidate games in the sense that, for $r = 2$, it coincides with the absolute Banzhaf index for simple games. Like the Banzhaf index, this index can be used to compare the voting powers of voters in election schemes where a group of voters is to elect one candidate from a slate of candidates or to select one alternative from a list of alternatives. Typical applications would be to the United States Electoral College system and to weighted voting schemes in county legislatures.

REFERENCES

- J. F. BANZHAF, *Weighted voting doesn't work: a mathematical analysis*, Rutgers Law Review, 19 (1965), pp. 317-343.
- , *One man, 3,312 votes: a mathematical analysis of the electoral college*, Villanova Law Review, 13 (1968), pp. 304-332.
- E. M. BOLGER, *The Banzhaf index for multicandidate presidential elections*, SIAM J. Algebraic Discrete Methods, 4 (1983), pp. 422-458.
- , *Power indices for multicandidate voting games*, Internat. J. Game Theory, 14 (1986), pp. 175-186.
- S. J. BRAMS, *Game Theory and Politics*, The Free Press, A Division of Macmillan Publishing Co., Inc., New York, 1975.
- S. J. BRAMS, P. AFFUSO, AND M. KILGOUR, *Presidential power: a game-theoretic analysis*, Research Report 88-01, C. V. Starr Center for Applied Economics, New York University, New York, 1988.
- S. J. BRAMS, *Are the two houses of Congress really co-equal?*, in the Federalist Papers in Public Choice Perspective: The New Institutionalism and the Old, Bernard Grofman and Donald Wittman, eds., Agathon, New York, 1988.
- P. DUBEY AND L. S. SHAPLEY, *Mathematical properties of the Banzhaf power index*, Math. Oper. Res., 4 (1979), pp. 99-131.
- B. GROFMAN AND H. SCARROW, *Iannucci and its aftermath: the application of the Banzhaf index to weighted voting in the state of New York*, in Applied Game Theory, Physica-Verlag, Wuerzburg, Germany, 1979.
- R. HOLZMAN, E. LEHRER, AND N. LINIAL, *Some bounds for the Banzhaf index and other semivalues*, presented at the International Conference on Game Theory, Columbus, Ohio, 1987.
- R. IMRIE, *The impact of the weighted vote of representation in municipal governing bodies in New York State*, Annals of the New York Academy of Science, 219 (1973), pp. 192-199.
- R. E. JOHNSON, *An analysis of weighted voting as used in reapportionment of county governments in New York State*, Albany Law Review, 34 (1969), pp. 1-45.
- W. F. LUCAS AND R. M. THRALL, *n-person games in partition function form*, Naval Res. Logist. Quart., X (1963), pp. 281-298.
- W. F. LUCAS, *Measuring power in weighted voting systems*, Technical Report no. 227, Department of Operations Research and Industrial Engineering, Cornell University, Ithaca, N.Y., 1976; also in Political and Related Models of Modules in Applied Mathematics, Springer-Verlag, New York, 1983.

- W. F. LUCAS, J. C. MACELI, M. R. HILLIARD, AND D. C. HOUSMAN, *Reapportionment by weighted voting*, Technical Report no. 533, Department of Operations Research and Industrial Engineering, Cornell University, Ithaca, N.Y., 1982.
- I. MANN AND L. S. SHAPLEY, *The a priori voting strength of the electoral college*, Research Manuscript no. 3158, Rand Corporation, Santa Monica, California, 1962.
- R. B. MYERSON, *Values of games in partition function form*, *Internat. J. Game Theory*, 6 (1977), pp. 23–31.
- G. OWEN, *Characterization of the Banzhaf–Coleman index*, *SIAM J. Applied Math.*, 35 (1978), pp. 315–327.
- G. OWEN AND F. CARRERAS, *An analysis of the Euskarian parliament*, Department of Mathematics, Naval Postgraduate School, Monterey, California, 1988.
- L. PAPAYANOPOULOS, Ed., *Democratic Representation and Apportionment: Quantitative Methods, Measures, and Criteria*, *Ann. New York Acad. Sci.*, 219 (1973).
- W. H. RIKER AND P. ORDESHOOK, *An Introduction to Positive Political Theory*, Prentice Hall, Englewood Cliffs, New Jersey, 1973.
- W. H. RIKER AND L. S. SHAPLEY, *Weighted Voting: A Mathematical Analysis for Instrumental Judgement*, in *Representation: Nomos X*, J. Pennock and J. Chapman, eds., Atherton Press, New York, 1968.
- L. S. SHAPLEY, *Solutions of compound simple games*, in *Annals of Mathematics Study* 52, Princeton University Press, Princeton, NJ, 1964, pp. 267–306.
- , *A value for n-person games*, in *Contributions to the Theory of Games II*, H. W. Kuhn and A. W. Tucker, eds., Princeton University Press, Princeton, NJ, 1953, 307–317.
- L. S. SHAPLEY AND M. SHUBIK, *A method for evaluating the distribution of power in a committee system*, *American Political Science Review*, 48 (1954), pp. 787–792.
- P. D. STRAFFIN, JR., *Homogeneity, independence, and power indices*, *Public Choice*, XXX (1977), pp. 107–118.

BINARY PROBABILITIES INDUCED BY RANKINGS*

PETER C. FISHBURN†

Abstract. A system $[p_{ij} : i, j \in \{1, 2, \dots, n\}, i \neq j, p_{ij} + p_{ji} = 1]$ of binary probabilities is said to be *induced by rankings* if there is a probability distribution P on the set of $n!$ linear orders of $\{1, 2, \dots, n\}$ such that, for all distinct i and j , p_{ij} is the sum of the P values over all linear orders in which i precedes j . It has been known for some time that the triangle inequality $p_{ij} + p_{jk} \geq p_{ik}$ is necessary for $\{p_{ij}\}$ to be induced by rankings and that it is also sufficient if $n \leq 5$. The insufficiency of the triangle inequality when $n \geq 6$ has been known since about 1970, and other necessary conditions for $n \geq 6$ have been known since 1978.

The present paper generates additional necessary conditions that pertain to $n \geq 6$ and shows that they are independent of previous necessary conditions. It then observes that the set of conditions on the p_{ij} that are sufficient for $\{p_{ij}\}$ to be induced by rankings regardless of n must be infinite. As far as is known, identification of finite sets of necessary and sufficient conditions for each particular $n \geq 6$ remains open.

Key words. binary probabilities, ranking probabilities, linear separation, partially ordered sets

AMS(MOS) subject classifications. 05A05, 06A10, 60C05

1. Introduction. A *binary probability system* on $\mathbf{n} = \{1, \dots, n\}$ is a nonnegative function p on $\{(i, j) : i, j \in \mathbf{n}, i \neq j\}$ that, with $p_{ij} = p(i, j)$, satisfies

$$(1) \quad p_{ij} + p_{ji} = 1$$

for all distinct $i, j \in \mathbf{n}$. We denote the set of all binary probability systems on $\mathbf{n}$ by B_n . Our aim is to identify conditions on the p_{ij} that are necessary, or necessary and sufficient, for $p \in B_n$ to be induced by some probability distribution on the set of linear orders on $\mathbf{n}$. For example, if $n = 3$ and $p_{12} = p_{23} = p_{31} = \frac{3}{4}$, then p is not so induced since otherwise $(p_{12} = \frac{3}{4}, p_{23} = \frac{3}{4}) \Rightarrow p_{13} \geq \frac{1}{2}$, thus contradicting $p_{13} = 1 - p_{31} = \frac{1}{4}$.

To be precise, for each $n \geq 3$ let S_n denote the set of all permutations σ on $\mathbf{n}$. We use the convention that $\sigma(i)$ is the left-to-right position in the ranking or linear order into which σ maps i . Let $\mathcal{P}_n$ be the family of all probability distributions on S_n . We then say that $p \in B_n$ is *induced by rankings* if there is a $P \in \mathcal{P}_n$ such that, for all $1 \leq i < j \leq n$,

$$p_{ij} = \sum_{\{\sigma \in S_n : \sigma(i) < \sigma(j)\}} P(\sigma).$$

At $n = 3$, the $p \in B_3$ induced by a specific $P \in \mathcal{P}_3$ has

$$p_{12} = P(123) + P(132) + P(312)$$

$$p_{23} = P(123) + P(213) + P(231)$$

$$p_{13} = P(123) + P(132) + P(213).$$

Here (312), for example, means that $\sigma(3) = 1$, $\sigma(1) = 2$, and $\sigma(2) = 3$. We assume $n \geq 3$ henceforth.

Let B_n^* be the subset of B_n whose members are induced by rankings. Our consideration of conditions on $p \in B_n$ that are needed for membership in B_n^* begins in the next section with a lemma that says exactly when $p \in B_n^*$. The lemma involves a straightforward application of linear separation and may seem uninformative since it mixes the p_{ij} with arbitrary real coefficients c_{ij} . We show, however, that specific c_{ij} values, namely $c_{ij} \in$

* Received by the editors October 20, 1988; accepted for publication (in revised form) October 6, 1989.

† AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

$\{1, 0, -1\}$, yield interesting necessary conditions for membership in B_n^* , including the triangle condition $p_{ij} + p_{jk} - p_{ik} \leq 1$.

Section 3 presents two sequences of such conditions that arise as generalizations of a necessary condition that uses six indices in $\mathbf{n}$ and is independent of the triangle condition. The first sequence, due to Cohen and Falmagne [6], is based on $2k$ indices for $k = 3, 4, \dots$. The second is based on $4k - 2$ indices for $k = 2, 3, \dots$. After the second sequence was developed by the present author, it was called to the author's attention that McLennan [28] gives a similar condition in his Theorem 7. Section 3 verifies that no member of either sequence is implied by its predecessors and concludes by observing that the two sequences of necessary conditions are independent.

Section 4 then demonstrates that there is no finite set of necessary conditions involving definite c_{ij} values and fixed numbers of indices that is sufficient for membership in B_n^* for all values of n . The results in §§ 3 and 4 are illuminated by the simple partially ordered sets referred to as fences and crowns in [1]. The paper concludes with a brief discussion.

Our study is motivated by a large literature in economics and psychology on systems of choice probabilities $\{\rho(x, X) : x \in X \subseteq U, \sum_x \rho(x, X) = 1\}$ in which $\rho(x, X)$ is interpreted as the probability that alternative x will be chosen from X when X is the "feasible set" of alternatives from a universe U of potential alternatives. Important works that explore facets of ρ and give additional references are [2]–[5], [8], [12], [19]–[25], [30]. When X 's are confined to two-element subsets of U , we encounter binary choice probabilities $\rho(x, \{x, y\})$, written also as p_{xy} or p_{ij} . The preceding references discuss binary choice probabilities, and other references that focus mainly on binary probability systems include [10], [11], [14], [17], [26], [28], [29], [31].

Our particular concern with binary probability systems induced by rankings was highlighted by Marschak [25] who popularized the triangle inequality

$$(\Delta) \quad p_{ij} + p_{jk} \geq p_{ik} \quad \text{for all distinct } i, j, k \in \mathbf{n}$$

as a necessary condition for $p \in B_n^*$. Marschak cited Guilbaud [16] as an earlier discussant of (Δ) , noted that it is sufficient for $p \in B_3^*$, remarked that Motzkin told him that (Δ) is also sufficient for $p \in B_n^*$ when $n \in \{4, 5\}$, and conjectured without conviction that (Δ) is sufficient for $p \in B_n^*$ when $n \geq 6$.

Motzkin's claim that (Δ) is sufficient for $p \in B_5^*$ was subsequently verified by Dridi [7]. A similar proof can be obtained from Theorem 2 in Fishburn [13], which demonstrates sufficiency when all p_{ij} are rational. Since the extreme points of the set of all $p \in B_5$ that satisfy (Δ) are rational, Lemma 1 in the next section shows that (Δ) is sufficient also when some of the p_{ij} are irrational.

Marschak's conjecture that (Δ) is sufficient when $n \geq 6$ was disproved independently by McFadden and Richter [27], Cohen and Falmagne [6], Dridi [7], Campello de Souza [4], and Fishburn [13]. Their main counterexample for $n = 6$, which satisfies (Δ) but is not in B_6^* , will be recalled in § 3 as the motivator of the two sequences of necessary conditions described there. Both sequences begin with the condition

$$(p_{ia} + p_{ib} + p_{jb} + p_{jc} + p_{kc} + p_{ka}) - (p_{ic} + p_{ja} + p_{kb}) \leq 4$$

whenever $\{i, j, k\}$ and $\{a, b, c\}$ are disjoint 3-sets in $\mathbf{n}$. Using (1) and interchanging indices, this condition can also be written as $1 + p_{ia} + p_{ib} + p_{jb} + p_{jc} + p_{kc} + p_{ka} \geq p_{ic} + p_{ja} + p_{kb}$ for easy comparison to (Δ) . The two sequences then continue along different paths.

A lemma that can be used to generate and verify necessary conditions for membership in B_n^* is described next.

2. Basic lemma. Let $I_n = \{(i, j) : 1 \leq i < j \leq n\}$ and

$$I_n(\sigma) = I_n \cap \{(i, j) : \sigma(i) < \sigma(j)\}$$

for each $\sigma \in S_n$. We write $c(i, j)$ as c_{ij} .

LEMMA 1. Suppose $p \in B_n$. Then $p \in B_n^*$ if and only if for all $c : I_n \rightarrow \mathbb{R}$ it is true that

$$(*) \quad \sum_{I_n} c_{ij} p_{ij} \leq \max_{\sigma \in S_n} \sum_{I_n(\sigma)} c_{ij}.$$

Proof. Write c and p as $\binom{n}{2}$ -dimensional vectors:

$$c = (c_{12}, \dots, c_{1n}, c_{23}, \dots, c_{2n}, \dots, c_{n-1,n})$$

and $p = (p_{12}, \dots, p_{1n}, p_{23}, \dots, p_{2n}, \dots, p_{n-1,n})$. The rest of p is obtained from (1). Also let $x(\sigma) = (x_{12}(\sigma), \dots, x_{1n}(\sigma), \dots, x_{n-1,n}(\sigma))$ with $x_{ij}(\sigma) = 1$ if $\sigma(i) < \sigma(j)$ and $x_{ij}(\sigma) = 0$ otherwise. Using scalar products, our claim is that $p \in B_n^*$ if and only if for all c there exists $\sigma \in S_n$ with

$$c \cdot (p - x(\sigma)) \leq 0.$$

The basic theorem [9], [15], [18] for the existence of a solution to a finite system of linear inequalities tells us that exactly one of the following is true:

- (A) $c \cdot (p - x(\sigma)) > 0$ for all $\sigma \in S_n$ and some c ;
- (B) there is a number $r_\sigma \geq 0$ for each $\sigma \in S_n$ such that $\sum_{S_n} r_\sigma > 0$ and

$$\sum_{S_n} r_\sigma (p_{ij} - x_{ij}(\sigma)) = 0 \quad \text{for all } (i, j) \in I_n.$$

Our preceding claim is equivalent to the falsity of (A) and hence to the truth of (B), so we need only show that $p \in B_n^*$ is equivalent to (B). This is immediate: (B) implies $p \in B_n^*$ by using P defined by $P(\sigma) = r_\sigma / (\sum_{S_n} r_\sigma)$, and conversely we obtain a solution to (B) by setting $r_\sigma = P(\sigma)$ when p is realized by the distribution P . $\square$

Remarks. Because of (1), we need pay attention to only one of p_{ij} and p_{ji} for $i \neq j$, and for convenience have chosen p_{ij} with $i < j$ for the statement of Lemma 1. If conditions involving p_{ji} for $i < j$ are desired, just replace p_{ij} by $1 - p_{ji}$ in (*). Moreover, by symmetry, or because the original assignment of integers 1 through n to the n base elements is arbitrary, any condition obtained from the lemma by specific c_{ij} values is also necessary for $p \in B_n^*$ under any permutation of the indices.

We illustrate Lemma 1 and the preceding remarks by two simple c assignments before turning to more involved cases in the next section.

Suppose $c_{12} > 0$ and $c_{ij} = 0$ otherwise. Then (*) is $c_{12}p_{12} \leq c_{12}$, or $p_{12} \leq 1$. Similarly, if $c_{12} < 0$ and $c_{ij} = 0$ otherwise, we get $c_{12}p_{12} \leq 0$, or $0 \leq p_{12}$. By symmetry, we then require $0 \leq p_{ij} \leq 1$ for all $i \neq j$. Note also that $p_{12} \leq 1$ by way of $c_{12} > 0$ gives $p_{21} \geq 0$ by (1), and these are enough to obtain $0 \leq p_{ij} \leq 1$ without going through the $c_{12} < 0$ step.

Suppose next that $c_{12} = c_{23} = -1$ and $c_{13} = 1$ with $c_{ij} = 0$ otherwise. Then $\max \sum_{I_n(\sigma)} c_{ij} = 0$, so the lemma gives $p_{13} \leq p_{12} + p_{23}$ and, by symmetry, all of (Δ) as a necessary condition for $p \in B_n^*$. Alternatively, if we take $c_{12} = c_{23} = 1$, $c_{13} = -1$ and $c_{ij} = 0$ otherwise, then $\max \sum_{I_n(\sigma)} c_{ij} = 1$, so the lemma yields

$$p_{12} + p_{23} - p_{13} \leq 1$$

as a necessary condition. This is simply an instance of (Δ) in disguise since (1) converts it to $p_{31} \leq p_{32} + p_{21}$.

3. Other necessary conditions. By Motzkin's sufficiency result for $n \leq 5$, any necessary condition that augments (Δ) in a nontrivial way must use at least six indices. We begin with such a condition for exactly six indices that was motivated by the counterexample to the sufficiency of (Δ) for $p \in B_6^*$ alluded to in § 1. We focus on this condition in the next few paragraphs and then generalize its pattern in two ways.

Suppose $n \geq 6$. For application of Lemma 1 take

$$c_{14} = c_{15} = c_{25} = c_{26} = c_{36} = c_{34} = 1,$$

$$c_{16} = c_{24} = c_{35} = -1,$$

and $c_{ij} = 0$ otherwise. To determine $\max_{S_n} \sum_{I_n(\sigma)} c_{ij}$, it suffices to suppose that $n = 6$. For convenience let

$$Z = \max_{\sigma \in S_6} \sum_{I_6(\sigma)} c_{ij}.$$

Clearly $Z \leq 6$. However, if all six (i, j) for which $c_{ij} = 1$ have $\sigma(i) < \sigma(j)$, we must also have $\sigma(i) < \sigma(j)$ for at least two of the pairs in $\{(1, 6), (2, 4), (3, 5)\}$. For example, if we avoid $\sigma(1) < \sigma(6)$ but require $\sigma(1) < \sigma(4)$, $\sigma(1) < \sigma(5)$, $\dots$, $\sigma(3) < \sigma(4)$, then the linear order for σ must be of the form

$$(23)61(45)$$

where (xy) denotes either xy or yx . Since a similar conclusion applies if either $\sigma(4) < \sigma(2)$ or $\sigma(5) < \sigma(3)$, the largest that $\sum_{I_6(\sigma)} c_{ij}$ can be in such cases is $6 - 2 = 4$.

Suppose next that exactly five (i, j) for which $c_{ij} = 1$ have $\sigma(i) < \sigma(j)$. Then $\sigma(i) < \sigma(j)$ for at least one pair in $\{(1, 6), (2, 4), (3, 5)\}$. Otherwise, if $\sigma(6) < \sigma(1)$, $\sigma(4) < \sigma(2)$ and $\sigma(5) < \sigma(3)$, and if say $\sigma(1) < \sigma(4)$, $\sigma(1) < \sigma(5)$, $\sigma(2) < \sigma(5)$ and $\sigma(2) < \sigma(6)$, we get $\sigma(6) < \sigma(1) < \sigma(4) < \sigma(2) < \sigma(6)$ for the absurdity that $\sigma(6) < \sigma(6)$. Hence $\sum_{I_6(\sigma)} c_{ij} \leq 5 - 1 = 4$ when exactly five (i, j) with $c_{ij} = 1$ have $\sigma(i) < \sigma(j)$.

It follows that $Z = 4$. Therefore, by Lemma 1,

$$(p_{14} + p_{15} + p_{25} + p_{26} + p_{36} + p_{34}) - (p_{16} + p_{24} + p_{35}) \leq 4$$

is necessary for $p \in B_n^*$ when $n \geq 6$. Moreover, as suggested earlier, this condition is not implied by (Δ) , and this is where the counterexample to Marschak's conjecture comes in. That example has

$$p_{14} = p_{15} = p_{25} = p_{26} = p_{36} = p_{34} = 1,$$

$$p_{ij} = \frac{1}{2} \quad \text{for all other } i < j.$$

It is easily seen that (Δ) holds, but the initial inequality of this paragraph fails since its left side is $6 - 3(1/2) = 4.5$. Hence this p cannot be in B_n^* . A variety of other p exhibit the same behavior, for example

$$\{p_{14} = p_{25} = \dots = p_{34} = 0.90; p_{16} = p_{24} = p_{35} = 0.46; p_{ij} = \frac{1}{2} \text{ for all other } i < j\},$$

so the first necessary condition beyond (Δ) is certainly not a degenerate exception to the ability of (Δ) to yield $p \in B_n^*$ when $n \geq 6$.

The symmetry principle of index permutation gives the following general form of the preceding condition:

$$(2) \quad (p_{ia} + p_{ib} + p_{jb} + p_{jc} + p_{kc} + p_{ka}) - (p_{ic} + p_{ja} + p_{kb}) \leq 4$$

for all disjoint 3-sets $\{i, j, k\}$ and $\{a, b, c\}$ in $\mathbf{n}$.

An equivalent condition, obtained by (1) as mentioned previously, is

$$(2') \quad p_{ic} + p_{ja} + p_{kb} \leq 1 + (p_{ia} + p_{ib} + p_{jb} + p_{jc} + p_{kc} + p_{ka})$$

for all disjoint 3-sets $\{i, j, k\}$ and $\{a, b, c\}$ in $\mathbf{n}$.

The preceding observations are summarized in the following proposition.

PROPOSITION 1. *Condition (2) is necessary for $p \in B_n^*$. Moreover, $(\Delta) \not\Rightarrow (2)$ when $n \geq 6$.*

The structure of (2) can be explained by the crown of order 3 shown in the upper part of Fig. 1. The diagram represents a 6-point partially ordered set in the usual manner. Its points are numbered to correspond to the development of (2) earlier in this section. To obtain the c_{ij} assignment used there we take $c_{ij} = 1$ if and only if i covers j (there is a line from i down to j). The three $c_{ij} = -1$ are the top-bottom (i, j) with no connecting line, and all other c_{ij} for $i < j$ are set at 0.

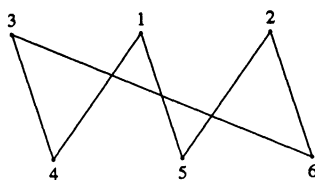
The generalization of (2) developed by Cohen and Falmagne [6] has a similar c_{ij} assignment for a $k \times k$ bipartite diagram that is complete except that for each of the k top points there is one bottom point, different for each top point, that is not covered by the top point. (In the theory of ordered sets, this is the diagram of the standard $2k$ -point poset of order dimension k .) We refer to the $k \times k$ condition used by Cohen and Falmagne as $CF(k-1)$ so that $CF(2)$ is identical to (2).

$$\sum_{i \in A} \sum_{j \in B \setminus \{f(i)\}} p_{ij} - \sum_{i \in A} p_{i, f(i)} \leq k^2$$

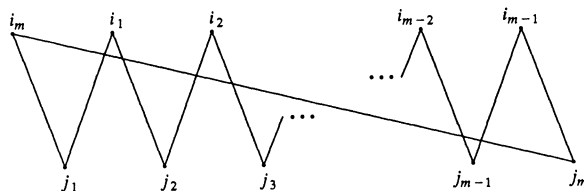
$CF(k)$

for all disjoint $(k+1)$ -sets A and B in $\mathbf{n}$

and for all bijections $f: A \rightarrow B$.



crown of order 3



crown of order m , $m \geq 3$

FIG. 1. Crowns.

An equivalent version using (1) is

$$\sum_{i \in A} p_{i,f(i)} \leq 1 + \sum_{i \in A} \sum_{j \in B \setminus \{f(i)\}} p_{ij}$$

CF(k')

for all disjoint $(k+1)$ -sets A and B in $\mathbf{n}$

and for all bijections $f: A \rightarrow B$.

Condition CF(k) is, of course, vacuous unless $n \geq 2(k+1)$. We omit the proof, which shows that CF(k) is necessary for $p \in B_n^*$ via Lemma 1 with $c_{ij} = 1$ for $(i, j) \in (\{1, \dots, k+1\} \times \{k+2, \dots, 2k+2\}) \setminus \{(1, k+2), (2, k+3), \dots, (k+1, 2k+2)\}$, $c_{ij} = -1$ for $(i, j) \in \{(i, k+1+i) : i = 1, \dots, k+1\}$ and $c_{ij} = 0$ otherwise. Necessity is covered by the first part of the following theorem, which is proved in a somewhat different way by Cohen and Falmagne [6].

THEOREM 1. *Condition CF(k) is necessary for $p \in B_n^*$ for each $k \geq 2$. Moreover, $\{(\Delta), \text{CF}(2), \dots, \text{CF}(k-1)\} \not\models \text{CF}(k)$ for each $k \geq 3$ when $n \geq 2(k+1)$.*

Proof. Necessity of CF(k) is proved in [6]. For the second part of the theorem take $k \geq 3$, $n = 2(k+1)$, $A = \{1, \dots, k+1\}$, $B = \{k+2, \dots, 2k+2\}$ and $f(i) = k+1+i$ for $i = 1, \dots, k+1$. Then define p by

$$p_{ij} = 1 \quad \text{for all } (i, j) \in A \times B \quad \text{with } j \neq f(i)$$

$$p_{i,f(i)} = (k-1)/k \quad \text{for } i = 1, \dots, k+1$$

$$p_{ij} = \frac{1}{2} \quad \text{for all other } i < j.$$

This violates CF(k) since the left side of its defining inequality is $(k+1)k - (k+1) \times [(k-1)/k] = (k+1)[k - 1 + 1/k] = k^2 + 1/k$. However, it satisfies (Δ) and CF(2) through CF(k-1). This is easily checked for (Δ) , so we consider CF(h) henceforth for $h < k$.

With $2 \leq h < k$, let A' and B' with $|A'| = |B'| = h+1$ and $A' \cap B' = \emptyset$ denote the two $(h+1)$ -sets in $\mathbf{n}$ used for CF(h). Also let $y = |A \cap A'|$ and $z = |A \cap B'|$, where A is given in the preceding paragraph, and let L denote the left side of the defining inequality for CF(h).

Suppose first that $z = 0$, so $B' \subset B$. If $y = h+1$, so that $A' \subset A$, then $L \leq (h+1)h - (h+1)[(k-1)/k] = (h+1)(h-1+1/k)$, and $(h+1)(h-1+1/k) \leq h^2$ since $h+1 \leq k$. If $y \leq h$, then

$$\begin{aligned} L &\leq \left[yh + (h+1-y) \left(\frac{1}{2}h \right) \right] - y[(k-1)/k] - (h+1-y) \frac{1}{2} \\ &= y \left[\frac{h+1}{2} - \frac{k-1}{k} \right] + (h+1)(h-1)/2. \end{aligned}$$

With $y \leq h$, the last expression is maximized at $y = h$, where it equals

$$h^2 + h/2 - \frac{1}{2} - h(k-1)/k.$$

Since this is less than h^2 when $k \geq 2$, CF(h) holds whenever $z = 0$.

Suppose next that $z \geq 1$. Then the values for the p_{ij} stated earlier give

$$\begin{aligned} L &\leq y \left[(h+1-z) + \frac{1}{2}(z-1) \right] + (h+1-y) \left[\frac{1}{2}(h+1-z) + \frac{1}{k}(z-1) \right] - \frac{1}{2}y - 0 \\ &= z[-y/2 - (k-2)(h+1-y)/(2k)] + (\text{terms without } z). \end{aligned}$$

The last expression is maximized when z is as small as possible, and it is easily seen to be $\leq h^2$ whenever $z \geq 2$ or ($z = 1, y \leq h$). The remaining case of ($z = 1, y = h + 1$) uses the fact that the one point in $B' \cap A$ is the $f(i)$ for only one $i \in A'$ to conclude that

$$L \leq h[(h-1+\frac{1}{2})-k/(k-1)]+h-\frac{1}{2}=h^2+h/2-\frac{1}{2}-h(k-1)/k < h^2$$

when $k \geq 2$. Therefore $CF(h)$ holds when $z \geq 1$.

Since this exhausts the possibilities for z , we conclude that $CF(h)$ holds for all $2 \leq h < k$. $\square$

Our new sequence of necessary conditions that generalizes (2) is based on crowns of odd orders $m \geq 5$ as illustrated in the lower part of Fig. 1. We first state these conditions and then relate them to the diagram.

Let $m = 2k - 1$ with $k \geq 2$. With the understanding that sets C^+ and C^- of ordered pairs in $\{i_1, \dots, i_{2k-1}\} \times \{j_1, \dots, j_{2k-1}\}$ depend on these index sets, let

$$C^+ = \{i_1 j_1, i_1 j_2, i_2 j_2, i_2 j_3, \dots, i_{2k-1} j_{2k-1}, i_{2k-1} j_1\}$$

$$C^- = \{i_1 j_{k+1}, i_2 j_{k+2}, \dots, i_{k-1} j_{2k-1}, i_k j_1, \dots, i_{2k-1} j_k\},$$

where parentheses have been omitted for convenience and, when all indices are different, $|C^+| = 2|C^-| = 4k - 2$. Condition (k) for $k \geq 2$ is

$$(k) \quad \sum_{ij \in C^+} p_{ij} - \sum_{ij \in C^-} p_{ij} \leq 3k - 2 \quad \text{for all disjoint} \\ (2k-1)\text{-sets } \{i_1, \dots, i_{2k-1}\} \quad \text{and} \quad \{j_1, \dots, j_{2k-1}\} \text{ in } \mathbf{n}.$$

Of course (k) $\equiv$ (2) when $k = 2$. The corresponding equivalent condition obtained by applying (1) to (k) is

$$(k') \quad \sum_{ij \in C^-} p_{ij} \leq k - 1 + \sum_{ij \in C^+} p_{ij} \quad \text{for all disjoint} \\ (2k-1)\text{-sets } \{i_1, \dots, i_{2k-1}\} \quad \text{and} \quad \{j_1, \dots, j_{2k-1}\} \text{ in } \mathbf{n}.$$

This follows the pattern of (Δ) and (2') preceding Proposition 1. Conditions (k) and (k') are vacuous unless $n \geq 4k - 2$.

With reference to the lower diagram of Fig. 1, C^+ , the set of index pairs whose p_{ij} have coefficient +1 in (k), consists of the covering pairs of the diagram. C^- , which has exactly half as many members as C^+ , matches each upper i_x vertex with the lower j_y vertex that lies midway between the two j 's covered by i_x in the cyclic arrangement of the j 's. The p_{ij} for C^- have coefficient -1 in (k), and all (i, j) that are in neither C^+ nor C^- have coefficient 0 for their p_{ij} . The midway provision for C^- is unambiguous since $m = 2k - 1$ is odd.

THEOREM 2. *Condition (k) is necessary for $p \in B_n^*$ for each $k \geq 2$. Moreover, $\{(\Delta), (2), \dots, (k-1)\} \not\models (k)$ for each $k \geq 3$ when $n \geq 4k - 2$.*

This extends Proposition 1 in the obvious way and gives another sequence of necessary conditions for $p \in B_n$ to be induced by rankings.

Proof. In view of Proposition 1, we can suppose that $k \geq 3$. Moreover, to prove Theorem 2 for k , it suffices to assume that $n = 4k - 2$ and that, by symmetry, $(i_1, \dots, i_{2k-1}) = (1, \dots, 2k-1)$ and $(j_1, \dots, j_{2k-1}) = (2k, \dots, 4k-2)$ for condition (k).

With these understandings, we show that (k) is tantamount to (*) of Lemma 1 when c_{ij} equals 1, -1 or 0 according to whether ij is in C^+ , C^- or neither, respectively. Since this c assignment equates the left sides of (k) and (*), we need only show that

$Z(\sigma) \leq 3k - 2$ for all $\sigma \in S_{4k-2}$, where

$$Z(\sigma) = \sum_{I_{4k-2}(\sigma)} c_{ij}.$$

Consider the array of $2k - 1$ rows given by

$$\begin{array}{ll} (1, 2k), (k, 3k) & (1, 3k), (k, 2k) \\ (1, 2k+1), (k+1, 3k) & (1, 3k), (k+1, 2k+1) \\ (2, 2k+1), (k+1, 3k+1) & (2, 3k+1), (k+1, 2k+1) \\ \dots & \dots \\ (k-1, 3k-2), (2k-2, 4k-2) & (k-1, 4k-2), (2k-2, 3k-2) \\ (k-1, 3k-1), (2k-1, 4k-2) & (k-1, 4k-2), (2k-1, 3k-1) \\ (k, 3k-1), (2k-1, 2k) & (k, 2k), (2k-1, 3k-1). \end{array}$$

The $2(2k - 1)$ pairs on the left are the elements in C^+ . Those on the right are the elements in C^- . Each C^- pair appears in two consecutive rows, with wrap-around. The two right pairs in each row come from the two left pairs under interchange of their second components. If two left pairs in a row, say (a, b) and (c, d) , agree with σ [a before b , c before d], then (a, d) or (c, b) must agree with σ . We can get as many as $2k - 1$ left pairs in σ , one from each row, without forcing a right pair in this way. But if there are t left pairs beyond $2k - 1$ that agree with σ , then these force at least $\lceil t/2 \rceil$ distinct pairs from the right side to agree with σ . Hence

$$Z(\sigma) \leq 2k - 1 + \max_{0 \leq t \leq 2k-1} (t - \lceil t/2 \rceil) = 3k - 2.$$

Therefore $Z(\sigma) \leq 3k - 2$ for all σ , and hence condition (k) is necessary for $p \in B_n^*$.

To complete the proof of Theorem 2 for $k \geq 3$ we show that (Δ) , (2) , $\dots$, $(k - 1)$ do not imply (k) when $n \geq 4k - 2$. With C^+ and C^- as in the initial part of this proof, define p by

$$\begin{aligned} p_{ij} &= 1 \quad \text{for all } ij \in C^+ \\ p_{ij} &= \frac{1}{2} \quad \text{for all other } i < j. \end{aligned}$$

Then (k) fails since its left side equals $|C^+| - \frac{1}{2}|C^-| = (4k - 2) - \frac{1}{2}(2k - 1) = 3k - 1.5$. But each of (Δ) , (2) , $\dots$, $(k - 1)$ holds. This is easily seen for (Δ) . For condition (j), $2 \leq j \leq k - 1$, it follows from the fact that no C^+ for (j) can be a crown, hence that $\sum_{(C^+)} p_{xy}$ for (j) cannot exceed $[(4j - 3) + \frac{1}{2}]$ while $\sum_{(C^-)} p_{xy}$ for (j) must be at least $\frac{1}{2}(2j - 1)$, so that the left side of (j) is no greater than $3j - 2$. $\square$

We conclude this section by showing that the sequences $CF(3)$, $CF(4)$, $\dots$ and (3) , (4) , $\dots$ of generalizations of (2) are independent.

Consider first the example for $k \geq 3$ that concludes the preceding proof and shows that condition (k) is not implied by (Δ) , (2) , $\dots$, $(k - 1)$. We claim that $CF(3)$, $CF(4)$, $\dots$ all hold. Indeed, if A and B are disjoint $(k + 1)$ -sets, then the left side of the inequality for $CF(k)$ can be no greater than

$$1[2(k + 1)] + \frac{1}{2}[(k + 1)k - 2(k + 1)] - \frac{1}{2}(k + 1) = (k + 1)^2/2,$$

and since this is less than k^2 whenever $k \geq 3$, all of $CF(3)$, $CF(4)$, $\dots$ hold. Thus (Δ) and all the Cohen-Falmagne conditions hold while (k) fails.

On the other hand, suppose $n = 8$ and let p be specified by

$$p_{16} = p_{17} = p_{18} = p_{25} = p_{27} = p_{28} = p_{35} = p_{36} = p_{38} = p_{45} = p_{46} = p_{47} = 1,$$

$$p_{15} = p_{26} = p_{37} = p_{48} = \frac{2}{3},$$

and $p_{ij} = \frac{1}{2}$ for all other $i < j$. This is the p used for $k = 3$ in the proof of Theorem 1, so it satisfies (Δ) and CF(2) but violates CF(3). Since $n = 8$, conditions (3), (4), $\dots$ also hold, so (Δ) and all conditions in the new sequence hold while CF(3) fails.

4. Conditions for all n . Let $\mathcal{N}$ be a set of necessary conditions for $p \in B_n^*$ and all $n \geq 3$ that use fixed coefficients on the p_{ij} and involve only finite numbers of indices (up to permutations). Conditions (Δ) , (2), (3), $\dots$ and CF(3), CF(4), $\dots$ are examples. Theorems 1 and 2 suggest that $\mathcal{N}$ must be infinite if it is sufficient for $p \in B_n^*$ for all $n \geq 3$. But the theorems do not rule out the possibility that other necessary conditions that keep $\mathcal{N}$ finite are sufficient no matter how large n gets. We show that this possibility cannot occur.

THEOREM 3. *If $\mathcal{N}$ is sufficient for $p \in B_n^*$ for all $n \geq 3$, then $\mathcal{N}$ is infinite.*

Proof. We show that $\mathcal{N}$ is not sufficient if it is finite. Assume that $\mathcal{N}$ is finite. Let K be the greatest number of indices (up to permutations) used in a condition in $\mathcal{N}$. Fix k such that $4k - 2 > K$ and let $n = 4k - 2$. As in the proof of Theorem 2, take $(i_1, \dots, i_{2k-1}) = (1, \dots, 2k - 1)$, $(j_1, \dots, j_{2k-1}) = (2k, \dots, 4k - 2)$, and, with $m = 2k - 1$ in the lower part of Fig. 1, define $p \in B_n$ by $p_{ij} = 1$ if i covers j in Fig. 1 (lower) and $p_{ij} = \frac{1}{2}$ for all other $i < j$. We already know by the final paragraph of the proof of Theorem 2 that condition (k) fails and hence that $p \notin B_n^*$.

We show that p satisfies every condition in $\mathcal{N}$. It suffices to prove that if any point is removed from $\mathbf{n}$ then the p_{ij} that remain are induced by rankings on the remaining $n - 1$ points. For then, since $K < n$, no condition in $\mathcal{N}$ can be negated by p .

For definiteness, remove $i_m = 2k - 1$ in the lower diagram of Fig. 1. What remains is a fence [1] except that the top row has one less point than the bottom row. In any event, we know that the resultant partially ordered set is two-dimensional. That is, there are σ_1 and σ_2 on $\mathbf{n} \setminus \{i_m\}$ such that

$$i \text{ covers } j \Leftrightarrow \sigma_1(i) < \sigma_1(j) \quad \text{and} \quad \sigma_2(i) < \sigma_2(j).$$

If neither i nor j covers the other and $i \neq j$, then either $\{\sigma_1(i) < \sigma_1(j), \sigma_2(j) < \sigma_2(i)\}$ or $\{\sigma_1(j) < \sigma_1(i), \sigma_2(i) < \sigma_2(j)\}$. Define P on $S_{\mathbf{n} \setminus \{i_m\}}$ by

$$P(\sigma_1) = P(\sigma_2) = \frac{1}{2}.$$

It follows immediately that the $p' \in B_{\mathbf{n} \setminus \{i_m\}}^*$ induced by P is identical to p restricted to $\mathbf{n} \setminus \{i_m\}$. Hence every instance of every condition in $\mathcal{N}$ must hold when the indices involved are included in $\mathbf{n} \setminus \{i_m\}$.

Since a similar conclusion clearly holds for any other point removed from $\mathbf{n}$, every condition in $\mathcal{N}$ holds for p . $\square$

5. Discussion. We have used linear duality through Lemma 1 to develop a sequence (3), (4), $\dots$ of conditions that are necessary for a binary probability system p to be induced by rankings. These conditions augment the triangle inequality (Δ) , (2), and the other conditions of Cohen and Falmagne [6], and these earlier conditions do not imply condition (k) when $k \geq 3$. Moreover, no finite set of "simple" necessary conditions is sufficient for p to be induced by rankings for all $n \geq 3$.

The main open problem that remains is to discover other necessary conditions that are not implied by those described above. One such family of conditions has been de-

veloped by Gilboa [14]. A more specific problem is to describe simple necessary and sufficient conditions for $p \in B_6^*$. What further conditions, if any, beyond (Δ) and (2) are sufficient to characterize all systems of binary probabilities induced by rankings when $n = 6$?

Acknowledgments. I am indebted to J. C. Falmagne and W. T. Trotter for calling to my attention the problem addressed here, to I. Gilboa for telling me about McLennan's paper [28], and to two referees for improving my proofs.

REFERENCES

- [1] K. A. BAKER, P. C. FISHBURN, AND F. S. ROBERTS, *Partial orders of dimension 2*, Networks, 2 (1971), pp. 11–28.
- [2] S. BARBERA AND P. K. PATTANAIK, *Falmagne and the rationalizability of stochastic choices in terms of random orderings*, Econometrica, 54 (1986), pp. 707–715.
- [3] H. D. BLOCK AND J. MARSCHAK, *Random orderings and stochastic theories of response*, in Contributions to Probability and Statistics, I. Olkin et al., eds., Stanford University Press, Stanford, 1960, pp. 97–132.
- [4] F. M. CAMPELLO DE SOUZA, *Mixed models, random utilities, and the triangle inequality*, J. Math. Psych., 27 (1983), pp. 183–200.
- [5] J. S. CHIPMAN, *Stochastic choice and subjective probability*, in Decision, Values and Groups, Vol. I, D. Willner, ed., Pergamon Press, New York, 1960, pp. 70–95.
- [6] M. COHEN AND J. C. FALMAGNE, *Random scale representation of binary choice probabilities: A counterexample to a conjecture of Marschak*, Department of Psychology, New York University, New York, 1978. Published as *Random utility representation of binary choice probabilities: a new class of necessary conditions*, J. Math. Psych., 34 (1990), pp. 88–94, with an introduction by A. A. J. Marley, pp. 81–87.
- [7] T. DRIDI, *Sur les distributions binaires associees a des distributions ordinales*, Math. Sci. Hum., 69 (1980), pp. 15–31.
- [8] J. C. FALMAGNE, *A representation theorem for finite random scale systems*, J. Math. Psych., 18 (1978), pp. 52–72.
- [9] P. C. FISHBURN, *Utility Theory for Decision Making*, John Wiley, New York, 1970.
- [10] ———, *Binary choice probabilities: On the varieties of stochastic transitivity*, J. Math. Psych., 10 (1973), pp. 327–352.
- [11] ———, *Binary choice probabilities between gambles: interlocking expected utility models*, J. Math. Psych., 14 (1976), pp. 99–122.
- [12] ———, *Choice probabilities and choice functions*, J. Math. Psych., 18 (1978), pp. 205–248.
- [13] ———, *Decomposing weighted digraphs into sums of chains*, Discrete Appl. Math., 16 (1987), pp. 223–238.
- [14] I. GILBOA, *A necessary but insufficient condition for the stochastic binary choice problem*, J. Math. Psych., to appear.
- [15] A. J. GOLDMAN, *Resolution and separation theorems for polyhedral convex sets*, in Linear Inequalities and Related Systems, Annals of Mathematics Study 38, H. W. Kuhn and A. W. Tucker, eds., Princeton University Press, Princeton, 1956, pp. 41–51.
- [16] G. GUILBAUD, *Sur une difficulté de la théorie du risque*, Colloques Internationaux du Centre National de Recherche Scientifique, (Econometrie), 40 (1953), pp. 19–25.
- [17] G. J. IVERSON, *Conditions under which Thurstone case III representations for binary choice probabilities are also Fechnerian*, J. Math. Psych., 20 (1979), pp. 263–271.
- [18] H. W. KUHN, *Solvability and consistency for linear equations and inequalities*, Amer. Math. Monthly, 63 (1956), pp. 217–232.
- [19] R. D. LUCE, *Individual Choice Behavior*, John Wiley, New York, 1959.
- [20] ———, *The choice axiom after twenty years*, J. Math. Psych., 15 (1977), pp. 215–233.
- [21] R. D. LUCE AND P. SUPPES, *Preference, utility, and subjective probability*, in Handbook of Mathematical Psychology, III, R. D. Luce, R. R. Bush, and E. Galanter, eds., John Wiley, New York, 1965, pp. 249–410.
- [22] M. J. MACHINA, *Stochastic choice functions generated from deterministic preferences over lotteries*, Economic J., 95 (1985), pp. 575–594.

- [23] C. F. MANSKI, *The structure of random utility models*, Theory and Decision, 8 (1977), pp. 229–254.
- [24] A. A. J. MARLEY, *Some probabilistic models of simple choice and ranking*, J. Math. Psych., 5 (1968), pp. 311–332.
- [25] J. MARSCHAK, *Binary-choice constraints and random utility indicators*, in Mathematical Methods in the Social Sciences 1959, K. J. Arrow, S. Karlin, and P. Suppes, eds., Stanford University Press, Stanford, 1960, pp. 312–329.
- [26] A. MATTENKLOTT, J. SEHR, AND K.-J. MIESCKE, *A stochastic model for paired comparisons of social stimuli*, J. Math. Psych., 26 (1982), pp. 149–168.
- [27] D. MCFADDEN AND M. K. RICHTER, *Revealed stochastic preference*, University of California, Berkeley, 1970.
- [28] A. MCLENNAN, *Binary stochastic choice*, University of Minnesota, Minneapolis, 1984.
- [29] P. J. H. SCHOEMAKER AND C. C. WAID, *A probabilistic dominance measure for binary choices: Analytical aspects of a multiattribute random weights model*, J. Math. Psych., 32 (1988), pp. 169–191.
- [30] A. TVERSKY, *Elimination by aspects: A theory of choice*, Psych. Review, 79 (1972), pp. 281–299.
- [31] A. TVERSKY AND J. E. RUSSO, *Substitutability and similarity in binary choices*, J. Math. Psych., 6 (1969), pp. 1–12.

THICKNESSES OF ORDERED SETS*

PETER C. FISHBURN†

Abstract. The *thickness* of a finite poset (partially ordered set) is the cardinality of the smallest partition of the poset into induced subposets of dimension two or less. Thickness combines aspects of width and length. It never exceeds length, can equal length for any finite length, and never exceeds about half the width. There are dimension 3 posets with arbitrarily large thicknesses, and there are thickness 2 posets with arbitrarily large dimension. For $m \geq 3$, the m -dimensional poset of the singletons and their complements of an m -set ordered by inclusion has thickness 2; the thickness of the m -dimensional poset of all subsets of an m -set ordered by inclusion goes to infinity as m gets large. Every semiorder has thickness 1 or 2, but an interval order can have arbitrarily large thickness. Ramsey theory is used to obtain some of the preceding results. Other methods are used to identify examples of posets with thickness 2 and to prove that the smallest poset with thickness 3 must have at least 15 points.

Key words. partially ordered set, thickness, width, length, dimensionality, Ramsey theory

AMS(MOS) subject classifications. primary, 06A10; secondary, 05A20, 05B40, 05C50

1. Introduction. This paper explores a parameter of finite posets referred to as thickness. We study thickness on its own and in relation to the more familiar parameters of width, length, and dimension. Each parameter describes aspects of the structure and complexity of ordered sets. Width and length do this transparently while dimension is more subtle. Thickness, too, is rather subtle and not easy to compute, but it has interesting characteristics not shared by other parameters and offers what we believe to be a good composite picture of the intricacy of a poset.

We begin with definitions and notation used throughout the paper, say, why thickness provides a salient description of poset complexity, and conclude this introduction with an outline of the paper.

It is presumed throughout that X is a nonempty finite set. A *finite poset* $P = (X, <_0)$ consists of X and an irreflexive and transitive binary relation $<_0$ on X . P 's *incomparability relation* $\sim_0$ and *equivalence relation* $\approx$ are defined on X by

$$\begin{aligned} x \sim_0 y & \text{ if } x \neq y, \text{ not } (x <_0 y) \text{ and not } (y <_0 x), \\ x \approx y & \text{ if } \{z \in X : x <_0 z\} = \{z \in X : y <_0 z\} \text{ and} \\ & \{z \in X : z <_0 x\} = \{z \in X : z <_0 y\}. \end{aligned}$$

P is a *linearly ordered set* (chain) if $\sim_0$ is empty. A *linear extension* of P is any linearly ordered set $(X, <_*)$ with $<_0 \subseteq <_*$. An *induced subposet* of P is any poset $(Y, <_0 \cap (Y \times Y))$ in which Y is a nonempty subset of X . A linearly ordered induced subposet is a *chain* in P , and an induced subposet for which $<_0 \cap (Y \times Y) = \emptyset$ is an *antichain* in P . Henceforth, subposet = induced subposet.

The *length* $L(P)$ of P is the maximum over all chains in P of the number of points in the chain, and the *width* $W(P)$ of P is the maximum over all antichains in P of the number of points in the antichain. The *dimension* [5] $D(P)$ of P is the minimum number of linear extensions of P whose intersection equals P . Dilworth [4] remarks in a footnote that $D(P) \leq W(P)$ and, in what has come to be known as *Dilworth's theorem*, proves that there are $W(P)$ chains in P the union of whose point sets equals X . Hiraguchi [14]

* Received by the editors August 14, 1989; accepted for publication January 11, 1990.

† AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

first proved that $D(P) \leq |X|/2$ when $|X| \geq 4$, which augments the earlier result of Dushnik and Miller [5] that

$$D(B_m) = m, \quad m \geq 2,$$

where B_m is the set of all singletons and their complements from $\mathbf{m} = \{1, \dots, m\}$, ordered by proper inclusion. With $2^{\mathbf{m}}$ the family of all subsets of $\mathbf{m}$, Komm [17] showed that D does not increase when other sets in $2^{\mathbf{m}}$ are added to B_m since

$$D(2^{\mathbf{m}}, \subset) = m.$$

Additional relations among dimension, width, and other parameters of posets are summarized in Kelly and Trotter [16], West [27], [28], Fishburn [7], and Trotter [23].

We refer to a poset with $D \leq 2$ as a *semichain*. All chains and antichains in a poset are semichains. So is any poset covered by two chains ($D \leq W = 2$) and any poset that is the disjoint sum of a chain and an antichain. Other examples include posets whose Hasse diagrams are downward branching rooted trees and planar posets with greatest (over all others) and least (beneath all others) elements [25]. See Fig. 1.

Our definition of thickness is motivated by Dilworth's characterization of width as the minimum number of *chains* in P whose points cover X :

The *thickness* $T(P)$ of a finite poset P is the minimum number of induced *semichains* in P whose points cover X .

It follows easily that $T < W$ when $W \geq 2$. Moreover, unlike dimension, thickness relates simply to length since $T \leq L$. Thus T is a rough composite of W and L , and every large- T poset must have large W and large L . Every poset in Fig. 1 has thickness 1.

In regard to poset complexity, thickness has the attractive properties of independence of equivalent points and independence of "extra" points in chains that are equivalent to adjacent points except for $<_0$ between the two. The first property is shared by L but not W or D ; the second by W and D but not L .

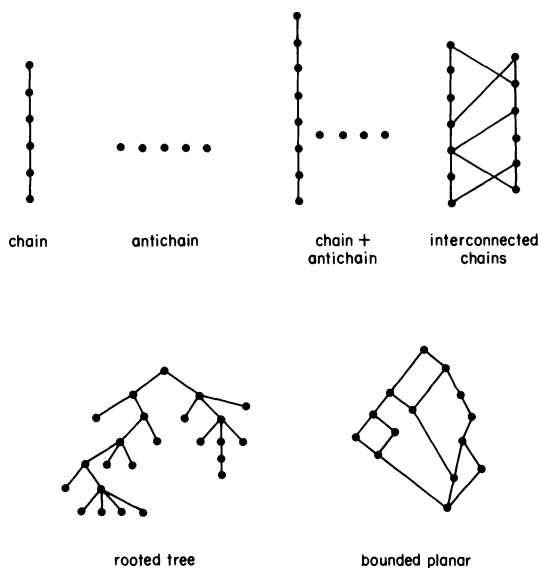


FIG. 1. *Semichains*: $T = 1$.

LEMMA 1. Given $P = (X, <_0)$, suppose $x, y \in X$, $x \neq y$ and for all $z \in X \setminus \{x, y\}$, $x <_0 z \Leftrightarrow y <_0 z$ and $z <_0 x \Leftrightarrow z <_0 y$. Let Q be P 's subposet on $X \setminus \{x\}$. Then $T(P) = T(Q)$.

Proof. Let the hypotheses hold. Clearly $T(Q) \leq T(P)$. Suppose $T(Q) = m$, realized by semichains $Q_i = (X_i, <_i)$, $i = 1, \dots, m$, with $\cup X_i = X \setminus \{x\}$. Assume, with no loss of generality, that $<_i$ is the intersection of linear orders $<_i^1$ and $<_i^2$ on X_i . Suppose $y \in X_i$. Form $P_i = (X_i \cup \{x\}, <'_i)$ as a semichain in P by adding x adjacent to y in $<_i^1$ and $<_i^2$. If $x \approx y$, take $x <_i^1 y$ and $y <_i^2 x$ augmented. If $x <_0 y$ ($y <_0 x$) take $x <_i^j y$ ($y <_i^j x$) for $j = 1, 2$. When $y \notin X_i$, let $P_i = Q_i$. Then m semichains, namely P_1 through P_m , cover X , so $T(P) \leq T(Q)$. $\square$

Bottom-up sequential applications of Lemma 1 to the tree in Fig. 1 verify $T = 1$ for that diagram.

Because T is based on subposets with $D \leq 2$, we recall characterizations of $D(P) \leq 2$ in the next section. We also note a few facts about $D \geq 3$ and identify the smallest posets for which $T = 2$. While it may be natural to generalize thickness by considering coverage of X by subposets with $D \leq d$ for various $d \geq 3$, we will not pursue this here. However, it is worth noting that while there is a polynomial algorithm to test $D(P) \leq 2$, it is NP-complete [29] for each $d \geq 3$ to determine if $D(P) \leq d$. Thus, even though the general determination of T may be hard, there is an efficient algorithm for testing whether a subposet is a semichain.

Section 3 says more about T versus W and L . In particular, $T \leq \min \{(W+1)/2, L\}$ and, for every $L \geq 1$, there is a poset for which $T = L$. The question of whether $T = (W+1)/2$ occurs for odd $W \geq 5$ is open.

Section 4 takes a closer look at $T \leq 2$. This category includes all finite semiorders, all B_m , $(2^4, \subset)$, and $(3^3, \leq)$, where

$$(\mathbf{n}^3, \leq) = (X, <_0) \quad \text{with } X = \{1, \dots, n\}^3 \text{ and } (a_1, a_2, a_3) <_0 (b_1, b_2, b_3) \text{ if} \\ a_i \leq b_i \text{ for all } i \text{ and } a_i < b_i \text{ for some } i.$$

$T(2^5, \subset)$ and $T(4^3, \leq)$ have not been determined. We observe also that every poset with fewer than 15 points and every poset with $(W, L) = (5, 3)$ has $T \leq 2$. The smallest $|X|$ for which $T = 3$ is unknown.

Section 5 notes two theorems involving D to go along with our earlier observations that $D(B_m) = m$ and $T(B_m) \leq 2$. First, $T(2^m, \subset) \rightarrow \infty$ as m gets large. Hence the dimension equality $D(B_m) = D(2^m, \subset)$ does not hold for thickness. This speaks favorably for T as a measure of complexity since 2^m is considerably more intricate than B_m for large m .

The second theorem in §5 says that $T(\mathbf{n}^3, \leq) \rightarrow \infty$ as n gets large. Since $D(\mathbf{n}^3, \leq) = 3$ for all $n \geq 2$, there are 3-dimensional posets of arbitrarily large thickness. It follows from this and the results for B_m that there is no simple inequality relationship between D and T .

Section 6 looks at thicknesses of interval orders to go along with the $T \leq 2$ result for semiorders. We consider the canonical interval order ψ_k whose points are the real intervals $[i, j]$ for all integers $1 \leq i \leq j \leq k$, with

$$[i, j] <_0 [i^*, j^*] \quad \text{if } j < i^*.$$

We recall from Bogart, Rabinovitch, and Trotter [3] that $D(\psi_3) = 2$, $D(\psi_4) = \dots = D(\psi_{11}) = 3$, $D(\psi_{12}) = 4$, $D(\psi_k) \rightarrow \infty$, and $D(\psi_k) < \log_a(k-1) + \frac{1}{2}$ with $a = 1 + \sqrt{2}$. Section 6 proves that $T(\psi_3) = 1$, $T(\psi_4) = \dots = T(\psi_{11}) = 2$, $T(\psi_k) \rightarrow \infty$, and

$T(\psi_k) \leq \log_3 [(k+1)/4] + 1$. The smallest k at which $T(\psi_k) = 3$ has not been determined.

The paper concludes with a brief summary and a list of open questions.

2. Semichains and more. Semichains, or posets with $D \leq 2$, have been studied extensively and characterized in several ways [1], [2], [5], [15], [24]. The following theorem is due mainly to Dushnik and Miller [5]. A linear extension $(X, <_*)$ of $(X, <_0)$ is *nonseparating* if, for all $x, y, z \in X$,

$$(x <_* y <_* z, x \sim_0 y, y \sim_0 z) \Rightarrow x \sim_0 z.$$

The incomparability graph $(X, \sim_0)$ with edge set $\sim_0 = \{\{x, y\} : x \sim_0 y\}$ is *transitively orientable* (or a comparability graph) if there is a poset $(X, <_1)$ such that, for all $x, y \in X$, $x \sim_0 y \Leftrightarrow (x <_1 y \text{ or } y <_1 x)$. We know [10], [11] that $(X, \sim_0)$ is transitively orientable if and only if, whenever $x_1 \sim_0 x_2 \sim_0 \cdots \sim_0 x_m \sim_0 x_1$ for odd $m \geq 3$, at least one of $x_1 \sim_0 x_3, x_2 \sim_0 x_4, \dots, x_{m-1} \sim_0 x_1$ and $x_m \sim_0 x_2$ holds.

THEOREM 1. *Suppose $P = (X, <_0)$ is not a chain. Then the following are mutually equivalent:*

- (a) $D(P) = 2$.
- (b) P has a nonseparating linear extension.
- (c) $(X, \sim_0)$ is transitively orientable.
- (d) There is an injection F from X into closed real intervals such that, for all $x, y \in X$, $x <_0 y \Leftrightarrow F(x) \subset F(y)$.
- (e) There is an injection F from X into closed real intervals such that, for all $x, y \in X$, $x <_0 y \Leftrightarrow [\inf F(x) < \inf F(y) \text{ and } \sup F(x) < \sup F(y)]$.

A proof is included in [7, pp. 85–86]. An efficient algorithm for determining whether a graph is transitively orientable [20] yields, by Theorem 1(c), a polynomial algorithm for testing whether P is a semichain. See also [12] and [24, pp. 183–186].

A different test for $D(P) = 2$ for a nonchain poset P is available through the complete characterizations of 3-irreducible posets given by Trotter and Moore [24] and Kelly [15]. They define a poset Q as 3-irreducible if $D(Q) = 3$ and every proper subposet of Q has $D \leq 2$. Their list of such posets consists of seven infinite families and 10 special posets (plus their inverses) that are not in the families. It then follows that $D(P) = 2$ for a nonchain poset if and only if no subposet of P is isomorphic to a poset in the 3-irreducible list.

The smallest 3-irreducible posets have 6 points. They are B_3 and the chevron (and its inverse) shown at the top of Fig. 2. The other posets in the figure, with inverses excluded, are the 21 (including inverses) 3-irreducible posets on 7 points. Each has $T = 2$. Kelly and Trotter [16] include these, as well as the four nonisomorphic 3-irreducible 8-point posets, in their Fig. 1.

We have already remarked that $D(B_m) = D(2^m, \subset) = m$ and that $D(\mathbf{n}^3, \leq) = 3$ for $n \geq 2$. The latter fact follows from the inclusion of B_3 as a subposet of $(\mathbf{n}^3, \leq)$ and from the lexicographic linear extensions

$$(a_1, a_2, a_3) <_1 (b_1, b_2, b_3) \text{ if } a_1 < b_1 \text{ or } (a_1 = b_1, a_2 < b_2) \text{ or}$$

$$(a_1 = b_1, a_2 = b_2, a_3 < b_3);$$

$$a <_2 b \text{ if } a_2 < b_2 \text{ or } (a_2 = b_2, a_3 < b_3) \text{ or } (a_2 = b_2, a_3 = b_3, a_1 < b_1);$$

$$a <_3 b \text{ if } a_3 < b_3 \text{ or } (a_3 = b_3, a_1 < b_1) \text{ or } (a_3 = b_3, a_1 = b_1, a_2 < b_2).$$

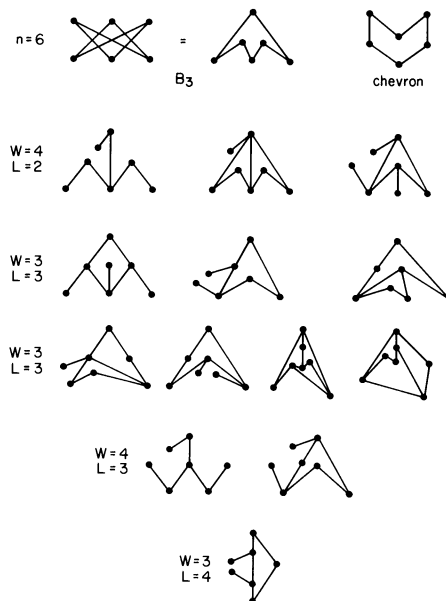


FIG. 2. 3-irreducible posets with six and seven points.

We can also use the alternative characterization of D as the dimension of the smallest dimensional Euclidean space into which X can be mapped so that $x <_0 y \Leftrightarrow f(x) < f(y)$ on every coordinate [14], [19].

A poset is an *interval order* [6] if no 4-point subposet is a disjoint sum of two length-2 chains, and is a *semiorder* [18] if no 4-point subposet is a disjoint sum of two nonempty chains. The following theorem is due to Rabinovitch [21] for D , and to Scott and Suppes [22] for f .

THEOREM 2. Suppose $P = (X, <_0)$ is a semiorder. Then $D(P) \leq 3$ and there is an $f : X \rightarrow \mathbb{R}$ such that, for all $x, y \in X$,

$$x <_0 y \Leftrightarrow f(x) + 1 < f(y).$$

The D part of the next theorem is from Bogart, Rabinovitch, and Trotter [3]; the F part is from [6].

THEOREM 3. There is no upper bound on the dimensions of interval orders. If $P = (X, <_0)$ is an interval order then there is an F from X into the closed real intervals with integer end points such that, for all $x, y \in X$,

$$x <_0 y \Leftrightarrow \sup F(x) < \inf F(y).$$

The latter part of Theorem 3 shows that, when $\approx$ is divided out, the quotient interval order is isomorphic to a subposet of some ψ_k as defined in the penultimate paragraph of § 1.

3. Thickness versus width and length.

THEOREM 4. $T \leq L$ and $T \leq (W + 1)/2$.

Remark. This holds for all finite posets. We omit P for convenience so, for example, $T \leq L$ means that $T(P) \leq L(P)$ for every finite poset P .

Proof. Suppose $L(P) = m$. Then P can be arranged in m levels (e.g., bottom-up) so that X is covered by m antichains in P . Since an antichain is a semichain, $T(P) \leq m$.

Suppose $W(P) = m$. By Dilworth's theorem, X is covered by m chains, which can be presumed to be mutually disjoint. Since two chains and their interconnections form a semichain by Dilworth's $D \leq W$, $T(P) \leq (m+1)/2$. $\square$

B_3 shows that equality can hold in $T \leq (W+1)/2$ when $W = 3$, but we do not know whether this is true for any larger odd W . However, equality is always possible in $T \leq L$.

THEOREM 5. *For every positive integer m there is a finite poset P for which $T(P) = L(P) = m$.*

We will use the following Ramsey theorem [13, p. 7].

LEMMA 2. *Let $k, s_1, \dots, s_r$ be any positive integers. There is an integer n_0 such that, for all $n \geq n_0$, if the k -element subsets of $\mathbf{n}$ are colored with r colors, then for some $i \in \mathbf{r}$ there is an s_i -element subset S of $\mathbf{n}$ such that all k -element subsets of S have color i .*

Proof of Theorem 5. An antichain and B_3 show the desired result for $m = 1, 2$, respectively. Assume henceforth that $m \geq 3$. For $n > m$ let

$$P_{m,n} = (\{A \in 2^n : 1 \leq |A| \leq m\}, \subset),$$

so $L(P_{m,n}) = m$. We suppose $T(P_{m,n}) \leq m-1$ for all $n > m$ and obtain a contradiction for large n .

Suppose $T(P_{m,n}) \leq m-1$. Then the j -element subsets of $\mathbf{n}$ ($j = 1, 2, \dots, m$) can be partitioned into $m-1$ parts each of which is a semichain in $P_{m,n}$. In particular, no part includes an induced B_3 .

We show that this B_3 conclusion is false for large n by proving that if the point set of $P_{m,n}$ is $(m-1)$ -colored then there must be a monochromatic copy of B_3 . With n large we first $(m-1)$ -color the m -element subsets of $\mathbf{n}$. In Lemma 2 take $s_1 = \dots = s_{m-1} = k_m$, with k_m as large as we please. By the lemma (n_0 depends on $m-1$ and k_m) there is a k_m -point subset S of $\mathbf{n}$, say $S = \{1, 2, \dots, k_m\}$ for definiteness, such that all m -sets in S are the same color.

Apply the same procedure to the $(m-1)$ -element subsets of S , again using $m-1$ colors with $s_1 = \dots = s_{m-1} = k_{m-1}$ in Lemma 2. We can take k_{m-1} as large as we please and adjust k_m and n accordingly to obtain a set S' of k_{m-1} points in S , say $S' = \{1, \dots, k_{m-1}\}$ for definiteness, so that the $(m-1)$ -sets in S' are all the same color.

The procedure continues down through the singletons, where we take $k_1 = 3$. For definiteness we get $k_1 < k_2 < \dots < k_m$ such that all j -sets of $\{1, 2, \dots, k_j\}$ are the same color. Since there are m levels but only $m-1$ colors, two levels have the same color. Suppose levels j and h , $j < h$, are the same color. Then points $\{1, \dots, j\}$, $\{1, \dots, j-1, j+1\}$ and $\{1, \dots, j-1, j+2\}$ from level j , and points $\{1, \dots, j, j+1\} \cup C$, $\{1, \dots, j, j+2\} \cup C$ and $\{1, \dots, j-1, j+1, j+2\} \cup C$ from level h [$C = \emptyset$ if $h = j+1$, $C = \{j+3, \dots, h+1\}$ otherwise] comprise a monochromatic copy of B_3 . $\square$

4. Examples of thickness 2. Our illustrations of posets with $T \leq 2$ begin with standard forms, go on to semiorders, and conclude with remarks for small $|X|$.

Since $D(B_m) = m$ and $L(B_m) = 2$, Theorem 4 shows that $T(B_m) = 2$ for all $m \geq 3$. $T(2^4, \subset) = 2$ is verified by the two semichains

$$\begin{aligned} &\{\emptyset, 1, 2, 12, 13, 24, 123, 124, 1234\} \\ &\{3, 4, 14, 23, 34, 134, 234\}, \end{aligned}$$

and $T(3^3, \leq) = 2$ is verified by the semichains

$$\begin{aligned} &\{111, 121, 211, 221, 122, 222, 132, 232, 133, 223, 323, 233, 333\}, \\ &\{131, 231, 311, 321, 331, 112, 212, 312, 322, 332, 113, 123, 213, 313\}. \end{aligned}$$

The latter two subposets are planar, the first is bounded by 111 and 333, and the second remains planar when 111 and 333 are added. Hence, by a theorem of Trotter and Moore [25], both have $D \leq 2$, and hence are semichains.

Theorem 2 says that $D \leq 3$ for all finite semiorders. The comparable fact for thickness is noted by the following theorem.

THEOREM 6. $T \leq 2$ for every finite semiorder.

Proof. Let $P = (X, <_0)$ be a semiorder represented by unit-length intervals $[f(x), f(x) + 1]$ as in Theorem 2. By Lemma 1 we can presume that $\approx$ never holds between distinct points in X , and can therefore identify the points in X uniquely by their f intervals.

Partition X into subsets A and B as follows. The leftmost interval and all its intersectors go into A . Delete those intervals. The leftmost interval and all its intersectors in the set that remains go into B . Delete those intervals also, and continue in the same manner with $A, B, A, B, \dots$ and successive deletions until all intervals in X are assigned to A or B . It is easily seen that $(\{f(x) : x \in A\}, <)$ is a linearly ordered set and is a nonseparating linear extension for the $x \in A$ of P 's subposet on A . The same is true for B . Hence, by Theorem 1(a), (b), $(A, <_0 \cap (A \times A))$ and $(B, <_0 \cap (B \times B))$ are semichains in P , so $T \leq 2$. $\square$

The rest of this section is motivated by the open question of the smallest $|X|$ for which some $P = (X, <_0)$ has $T = 3$. For convenience here we let $n = |X|$.

We know that $T = 1$ if $n \leq 5$ and that T can equal 2 for $n = 6$ (B_3 , chevron). Our best guess for the smallest n that allows $T = 3$ is around 20, but we have only been able to prove that it must be at least 15. Some results on the matter follow. Apart from Lemma 3, they are rather messy to verify. Outlines of their proofs are in the Appendix.

Our first lemma identifies certain types of semichains. We let x^0 denote the degree of $x : x^0 = |\{y : x <_0 y\}| + |\{y : y <_0 x\}|$.

LEMMA 3. $P = (X, <_0)$ is a semichain if X is covered by subposets A and B with α and β points, respectively, and any one of the following holds:

1. A and B are antichains, $\alpha \leq 2$;
2. A and B are antichains, $\alpha = 3$, and A contains points x and y with $x^0 \leq 1$ and $y^0 \leq 2$;
3. A and B are antichains, $\alpha = \beta = 4$, and the diagram of P includes a complete bipartite 3×3 graph $K_{3,3}$;
4. A is a chain, B is an antichain, $\alpha \leq 3$;
5. A is a chain, B is an antichain, $\beta \leq 2$.

Lemma 3 is helpful in trying to determine the largest k such that every n -point poset includes a k -point semichain (as a subposet). One result along this line is the following lemma.

LEMMA 4. Every 9-point poset includes a 7-point semichain.

Lemmas 3 and 4 ease the burden of proving the next theorem.

THEOREM 7. $T \leq 2$ if $n \leq 14$.

A related problem is to determine the largest possible $T(P)$ for all posets of fixed width and length. The first nontrivial result for this problem is Theorem 8.

THEOREM 8. $T \leq 2$ for all posets with $(W, L) = (5, 3)$.

Since the posets of Theorem 8 have as many as 15 points, the theorem supports the conjecture that $n \leq 15 \Rightarrow T \leq 2$.

5. Thickness versus dimension. For $m \geq 3$, $D(B_m) = D(2^m, \subset) = m$ and $T(B_m) = 2$. In addition, the following theorem holds.

THEOREM 9. $T(2^m, \subset) \rightarrow \infty$ as m gets large.

Proof. For this proof, see the proof of Theorem 5. $\square$

It is true also that T/D can be arbitrarily large. We showed in § 2 that $D(\mathbf{n}^3, \leq) = 3$ for $n \geq 2$.

THEOREM 10. $T(\mathbf{n}^3, \leq) \rightarrow \infty$ as n gets large.

Proof. A simple cube in $\mathbf{n}^3$ is any $\{a_1, b_1\} \times \{a_2, b_2\} \times \{a_3, b_3\}$, $1 \leq a_i < b_i \leq n$ for each i . Given any integer $k > 0$, a layered pigeonhole construction (or see [13, pp. 33–38]) shows that if n is large and the points of $\mathbf{n}^3$ are k colored, then $\mathbf{n}^3$ has a monochromatic simple cube. Since a simple cube includes a copy of B_3 , it follows that $T(\mathbf{n}^3, \leq) > k$ for sufficiently large n . $\square$

It is easily verified that if a rectangular planar grid of points with $(t+1)!(k-1) + 1$ rows and $\binom{t+1}{2}$ columns is 2 colored, then it has a monochromatic $k \times t$ rectangular subgrid. When this result is applied sequentially to the three vertical layers of $(16681 \times 21 \times 3, \leq)$, we get a monochromatic 145×6 array in the bottom layer, then a monochromatic 7×3 array in the middle layer directly over the 145×6 array, and then a monochromatic 2×2 array in the top layer directly above the 7×3 array. This forces a monochromatic simple cube and shows that there is a $T \geq 3$ poset with $(16681)(21)(3) = 1,050,903$ points.

The challenge, of course, is to use more refined methods to show that some poset with $|X| \ll 10^6$ has $T \geq 3$.

6. Thickness for interval orders. It is easily checked that $T(\psi_3) = 1$ and $T(\psi_4) = 2$, where ψ_k is the interval order defined in § 1 and remarked on briefly at the end of § 2.

THEOREM 11. For all $m \geq 1$, $T(\psi_k) \leq m$ if $k \leq 4 \cdot 3^{m-1} - 1$.

Proof. The conclusion holds for $m = 1$. Proceeding by induction, assume that it holds through $m = t$. Let $k = 4 \cdot 3^t - 1$ with $a = 4 \cdot 3^{t-1}$ and $b = 8 \cdot 3^{t-1}$. With $c = 4 \cdot 3^{t-1} - 1$, k had c points less than a , c points strictly between a and b , and c points greater than b . Let A denote the subposet of ψ_k whose elements are all intervals of ψ_k that contain neither a nor b . Then A has three disjoint parts ($<a$, between a and b , $>b$), each part has $T \leq t$ by the induction hypothesis, and therefore $T(A) \leq t$.

Let B denote the subposet of ψ_k whose elements are the intervals of ψ_k that are not in A . Each interval for B contains a or b or both. By Lemma 1, $T(B)$ equals the thickness of the poset in Fig. 3. The linear extensions

$$([a, b], [a, b-1], \dots, [a, a], [b, b], [b-1, b], \dots, [a+1, b])$$

and

$$([a, a], [a+1, b], [a, a+1], [a+2, b], [a, a+2], [a+3, b], \dots, [a, b-1], [b, b], [a, b])$$

show that it has $T = 1$. Hence $T(\psi_k) \leq t + 1$. Therefore $T(\psi_j) \leq t + 1$ whenever $j \leq k = 4 \cdot 3^t - 1$, so the conclusion of the theorem holds for $m = t + 1$. Induction shows that it holds for all $m \geq 1$. $\square$

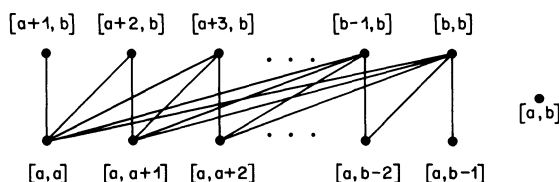


FIG. 3

Theorem 11 leaves open whether $T(\psi_{12})$ equals 2 or 3. Although we give no interesting lower bound on $T(\psi_k)$, our final theorem guarantees the existence of an increasing lower bound.

THEOREM 12. $T(\psi_k) \rightarrow \infty$ as k gets large.

Proof. Let m be any positive integer. We show that $T(\psi_k) > m$ for sufficiently large k .

Identify the intervals $[i, j]$ in ψ_k that have $i < j$ with the 2-sets $\{i, j\}$ of $\mathbf{k}$. Color the 2-sets of $\mathbf{k}$ with m colors and apply Lemma 2 with $r = m$ and $s_i = 6$ for $i = 1, \dots, r$. Then, with k large, there is a 6-set S in $\mathbf{k}$ such that all 2-sets in S are the same color. Suppose for definiteness that $S = \{1, 2, \dots, 6\}$. Then the interval order for $\{[1, 2], [3, 4], [5, 6], [2, 3], [4, 5], [1, 4], [3, 6]\}$ is the penultimate 7-point 3-irreducible poset shown in Fig. 2. It follows that the intervals in ψ_k cannot be partitioned into m sets each of which is a semichain. Hence $T(\psi_k) > m$. $\square$

7. Discussion. The thickness $T(P)$ of poset $P = (X, <_0)$ is the minimum number of semichains in P whose points cover X . The paper argues that thickness provides a good holistic measure of the complexity of a poset. It is affected neither by the presence of equivalent points nor by “extra” points in chains in P . If the diagram of P has several disjoint components, $T(P)$ is the maximum thickness of a component. If X can be partitioned into subsets $X_1, X_2, \dots, X_t$ such that $X_1 <_0 X_2 <_0 \dots <_0 X_t$, then $T(P)$ is the maximum thickness of an X_i .

We have seen that posets with $T = 1$, or dimension $D \leq 2$, have several nice characterizations and tend to have comparatively simple diagrams. $T \leq 2$ posets include all semiorders, all B_m (singletons and their complements, $\subset$), and all posets with fewer than 15 points. The smallest poset with $T = 3$ has not been determined.

Thickness is bounded by width W and length L as $T \leq \min \{(W + 1)/2, L\}$. For sufficiently large m the length L poset $(\{A \in 2^{\mathbf{m}} : 1 \leq |A| \leq L\}, \subset)$ has $T = L$. There is no simple relationship between T and D since

$$\begin{aligned} T \leq 2 \quad \text{and} \quad D \rightarrow \infty \quad \text{for the } B_m; \\ T \rightarrow \infty \quad \text{and} \quad D \leq 3 \quad \text{for the } (\mathbf{n}^3, \leq). \end{aligned}$$

Both T and D are unbounded for interval orders as well as for $(2^{\mathbf{m}}, \subset)$. Since angle orders and circle orders [26] include the interval orders [8], [9], those classes also have unbounded T .

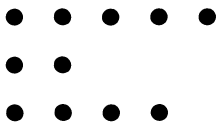
Although we have tried to paint a fairly broad picture of T , many questions about thickness remain open. Here are seven:

1. Do all 15-point posets have $T \leq 2$?
2. What is $T(2^5, \subset)$?
3. What is $T(4^3, \leq)$?
4. Does any $W = 5$ poset have $T = 3$?
5. What is the smallest k for which $T(\psi_k) = 3$?
6. Does $D(2^{\mathbf{m}}, \subset)$ grow at a faster rate than $T(2^{\mathbf{m}}, \subset)$?
7. Does $D(\psi_k)$ grow at a faster rate than $T(\psi_k)$?

More ambitious concerns focus on the smallest $|X|$ for which some poset has $T = k$ ($k = 3, 4, \dots$), or good upper bounds thereon; the largest k such that every n -point poset has a k -point semichain ($n = 7, 8, \dots$); and the maximum thickness of all posets of width W and length L , $(W, L) > (5, 3)$.

Appendix. This appendix outlines the proofs of Lemmas 3 and 4 and Theorems 7 and 8. Throughout, $n = |X|$, and $r = (r_1, \dots, r_L)$ denotes any poset of length L whose

diagram points can be arrayed in L rows, top to bottom, with r_j points in row j . Thus, with covering edges omitted, the points for $r = (5, 2, 4)$ and $n = 11$ can be arrayed as



Such an r need not be unique for a given poset (because of noncovering by an adjacent row), and any T property that holds for $r = (r_1, \dots, r_L)$ also holds for its inverse $r = (r_L, \dots, r_1)$. We generally ignore isolated points ($\sim_0$ to all others) since they do not affect T .

For any point x in an r array for a specific poset or class of posets, $u^0(x)$ is the number of covering edges between x and the row above it, and $d^0(x)$ is the number of covering edges between x and the row below it.

LEMMA 3.1. *If $r = (n - 2, 2)$ then $T = 1$.*

Proof. Lemma 1 reduces the poset to a T -equivalent poset with $n \leq 5$. $\square$

LEMMA 3.2. *If $r = (n - 3, 3)$ and there are a and $b \neq a$ in the bottom row with $u^0(a) \leq 1$ and $u^0(b) \leq 2$, then $T = 1$.*

Proof. Suppose a , b , and c are the bottom points with $u^0(a) = 1$, $u^0(b) = 2$, $u^0(c) > 0$. If a and b have a covering point in common, Lemma 1 reduces the poset to a T -equivalent poset with $n \leq 6$ and $L \leq 2$ that is not B_3 . If the covers of a and b are disjoint, Lemma 1 reduces it to a poset with $n \leq 7$ and $L \leq 2$ that does not include B_3 and is not isomorphic to any 7-point poset in the second row of Fig. 2. $\square$

LEMMA 3.3 (extended). *If $r = (m + 1, m + 1)$, with a $K_{m,m}$ in its diagram, then $T = 1$.*

Proof. Lemma 1 reduces it to an $r \leq (3, 3)$ poset that is not B_3 . $\square$

LEMMA 3.4. *If X can be partitioned into a chain with $L \leq 3$ and an antichain, then $T = 1$.*

Proof. The diagram of such a poset has a planar arrangement after addition of a greatest point and a least point. Then $D \leq 2$ by [25]. $\square$

LEMMA 3.5. *If X can be partitioned into an antichain with $W \leq 2$ and a chain, then $T = 1$.*

Proof. For this proof, see the preceding proof. $\square$

LEMMA 4. *Every 9-point poset includes a 7-point semichain.*

Proof. This is true by Lemmas 3.1 and 3.4 if $W \geq 5$ or $L \leq 2$, and by Theorem 4 and Lemma 3.5 if $W \leq 2$ or $L \geq 5$. Assume $L, W \in \{3, 4\}$ henceforth.

Suppose $L = 4$. If one point in a maximum chain is also in an antichain with three others, then Lemma 3.4 yields the desired conclusion. Assume otherwise. Then r must be $(3, 2, 2, 2)$ or $(3, 3, 2, 1)$ or some permutation thereof. If any one of 3222, 3312, 3132, 3213 or their inverses obtains, keep the four points in a maximum chain plus three others for $r = (3, 1, 1, 2)$ on 7 points with $L = 4$ (or the inverse of this r). The resultant 7-point poset must be a semichain because it includes neither a chevron nor the only $L = 4$ 3-irreducible 7-point poset (see bottom of Fig. 2). The other possibilities for r with $n = 9$ are 3321, 3231, 2331, 2322 and their inverses. If any of these obtains and cannot be turned into an r pattern already considered, the poset must contain disjoint chains with 4 and 3 points, which by Theorem 4 gives a 7-point semichain. We omit the details.

Assume henceforth that $L = 3$ and $W \in \{3, 4\}$. Then r is one of 441, 432, 423, 414, 342, and 333 (or an inverse thereof). If one of 441, 432, and 423 is possible, consider a 7-point $L = 3$ subposet with $r = 421$. This subposet is a semichain because (i) the two

$L = 3$ 7-point 3-irreducible posets in Fig. 2 that have $W = 4$ can only have a 4 in the middle position of their r 's, (ii) it does not include a chevron, whose r must be 222, 231, or 132, and (iii) it does not include a B_3 because the three lowest points from 421 for the bottom row of B_3 conflict with $L = 3$. If $r = 414$, either it yields a 421 or 124 7-point subposet as just considered, or else $u^0 = d^0 = 4$ for the point in the middle row, in which case all 9 points form a semichain. This leaves $r = 342$ and $r = 333$ for further consideration.

Suppose $r = 342$ and no other r considered previously, such as 432 or 324, is possible for the given poset. Then part of its diagram must be as shown on the left of Fig. 4. We begin there and carry out a sequential analysis. Only the main steps are noted. OK means that the poset has a 7-point semichain.

1. Suppose $x <_0 \{b, c\}$. Then OK. Assume $x \sim_0 c$ henceforth.

2. Suppose $x <_0 b$. If $q <_0 x$ also, OK, so assume $q \sim_0 x$ henceforth in this step. Suppose $y <_0 c$. If $p <_0 c$, OK, so assume $p \sim_0 c$ when $x <_0 b$ and $y <_0 c$. If $p <_0 y$, OK, so assume $p \sim_0 y$. Then OK. Assume $y \sim_0 c$ henceforth in this step. (We now have $x \sim_0 c$, $x <_0 b$, $x \sim_0 q$, $y \sim_0 c$.) If $y \sim_0 \{a, b\}$, OK: we get a 412. Assume $y <_0 a$ or $y <_0 b$. Suppose $y <_0 b$. If $p \sim_0 y$, OK ($abcxypq$), so assume $p <_0 y$. If also $y <_0 a$, OK, so take $y \sim_0 a$. Then OK. Assume henceforth in this step that $y \sim_0 b$ and $y <_0 a$. If $p <_0 y$, OK, so assume $p \sim_0 y$. Then OK. Assume $x \sim_0 b$ henceforth.

3. If $\{y, z\} \sim_0 \{b, c\}$, OK (5-point antichain). Assume henceforth for definiteness that $y <_0 b$. See Fig. 4.

4. Suppose $y \sim_0 a$. If either $q <_0 x$ or $p <_0 y$, OK, so assume $q \sim_0 x$ and $p \sim_0 y$ henceforth in this step. If $p \sim_0 b$ or $q \sim_0 a$, OK, so take $p <_0 b$ and $q <_0 a$. Then OK. Assume henceforth that $y <_0 a$.

5. Suppose $y <_0 c$. If $p <_0 y$, OK, so assume $p \sim_0 y$. Then OK. Assume $y \sim_0 c$ henceforth.

6. Suppose $q <_0 x$. If $z <_0 c$, OK, so assume $z \sim_0 c$. If $q <_0 c$, OK. Assume $q \sim_0 c$. Then OK. Assume $q \sim_0 x$ henceforth. See Fig. 4.

7. Suppose $c <_0 z$. If $z \sim_0 a$ or $z <_0 b$, OK. Assume $z <_0 a$ and $z \sim_0 b$ henceforth in this step. If $w \sim_0 c$ or $w \sim_0 b$, OK. Assume $w <_0 \{b, c\}$ henceforth in this step. If $q <_0 w$, OK. Assume $q \sim_0 w$. Then OK ($bcxywpq$). Assume $c \sim_0 z$ henceforth.

8. Suppose $q \sim_0 c$. Then OK ($abcxyzq$). Assume $q <_0 c$ henceforth. See Fig. 4.

9. Only a and b versus z remain open in the rightmost diagram of Fig. 4. However they are chosen, that diagram is a semichain.

This completes our outline for $r = 342$.

Assume $r = 333$ henceforth for $L = 3$ and that none of the previous r 's holds for the given poset. Then every point in the top row has $d^0 \geq 1$, every point in the middle row has $u^0 d^0 \geq 1$, and every point in the bottom row has $u^0 \geq 1$. It then follows easily that the poset has a 7-point semichain if any of the following holds:

- (a) The top two rows are B_3 , the bottom two rows are B_3 ;
- (b) an x in the top row has $d^0(x) = 3$;
- (c) An x in the bottom row has $u^0(x) = 3$.

Assume henceforth for $r = 333$ that none of these holds.

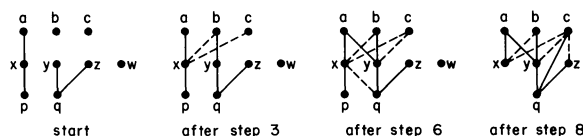


FIG. 4. Dashed lines denote $\sim_0$.

Suppose in this paragraph that the top two rows are B_3 . If one of the middle points has $d^0 = 3$, then that point plus the first and third rows forms a semichain. If one middle point has $d^0 = 2$ then that point, the two it covers in the bottom row, another point in the middle row, and the top three points form a semichain. If all three points in the middle row have $d^0 = 1$ then there is a 1-1 cover correspondence between that row and the bottom row, and the bottom two rows plus a point from the top row form a semichain. Hence there is a 7-point semichain if the top two rows are B_3 or, symmetrically, if the bottom two rows are B_3 .

Assume henceforth that there is no B_3 on the top or the bottom. At this point we proceed on the basis of the number k of covering edges between the top two rows, $k = 3, 4, 5, 6$. When $k = 3$, we have a 1-1 cover correspondence between rows 1 and 2, and those two rows plus any bottom point form a semichain. When $k = 6$, our prior assumptions force $d^0 = 2$ for each top point and up degrees of 3, 2, and 1 for the middle points. The $u^0 = 3$ point in row 2 plus the top and bottom rows form a semichain. When $k \in \{4, 5\}$, several patterns are possible for the upper edges, but it is straightforward to show that a 7-point semichain obtains for each pattern.

This completes our analysis of $r = 333$, hence of $L = 3$, and completes the proof of Lemma 4. $\square$

THEOREM 7. $T \leq 2$ if $n \leq 14$.

Proof. Let $n = 14$. If $L \leq 2$, the desired result follows from Theorem 4. If $L = 3$, one r_i in $r = (r_1, r_2, r_3)$ is 4 or less, and when part of this row is combined with a second row and the other part is combined with the third row, Lemma 3.1 implies that X is covered by two semichains.

Assume henceforth that $L \geq 4$ and $W \geq 5$. If $W \geq 6$, the desired result follows from Lemma 3.4. If $W = 5$, 7 of the 9 points not in a maximum antichain form a semichain by Lemma 4, and the other 2 of the 9 in conjunction with the antichain form another semichain by Lemma 3.1 or Lemma 3.4. $\square$

THEOREM 8. $T \leq 2$ for all posets with $(W, L) = (5, 3)$.

Proof. Let P be a poset of width 5 and length 3. If $n < 15$, the result follows from Theorem 7, so assume that $n = 15$. Form $r = (r_1, r_2, r_3)$. If it is possible to have $r_i \leq 4$ for any i then, as in the first part of the preceding proof, two semichains cover X . Assume henceforth that $r_i = 5$ for all i and that every point has a covering edge to the row above (if any) and the row below (if any).

If any row has two points with edge degrees ≤ 1 and ≤ 2 to an adjacent row, then Lemma 3.2 (antichains with 5 and 3 points) and Lemma 3.1 (antichains with 5 and 2 points) give the desired result. Assume henceforth that no row has two such points. Assume also that there is no $K_{3,3}$ for two rows since otherwise $T \leq 2$ follows from Lemma 3.3 and Lemmas 3.1 and 3.4.

At this point the analysis is based on u^0 and d^0 , mainly for the points in the middle row. Let M be the set of 5 points in that row. We proceed through the following steps (details omitted).

1. Suppose $u^0(x) = 1$ and $d^0(y) = 1$ for $x, y \in M$. Then all $z \in M \setminus \{x\}$ have $u^0 \geq 3$, all $z \in M \setminus \{y\}$ have $d^0 \geq 3$, and we get a $K_{3,3}$ between the top and bottom rows.
2. Suppose $u^0(x) = 1$ for some $x \in M$, and $d^0(y) \geq 2$ for all $y \in M$. Then $T \leq 2$. Assume henceforth that $u^0(x) \geq 2$ and $d^0(x) \geq 2$ for all $x \in M$.
3. Suppose $d^0(t) = 1$ for some t in the top row. Then $T \leq 2$. Assume henceforth that $d^0 \geq 2$ and $u^0 \geq 2$ for every applicable point in the poset.
4. Suppose $u^0(x) \geq 4$ for some $x \in M$. Then $T \leq 2$. Assume henceforth that $u^0(x) \in \{2, 3\}$ and $d^0(x) \in \{2, 3\}$ for all $x \in M$.

5. Suppose there are 10 covering edges between the top two rows, so $d^0(t) = 2 = u^0(x)$ for all t in the top row and all $x \in M$. Then $T \leq 2$. We also get $T \leq 2$ if there are 11 edges between the top two rows, whence $d^0(t) = 3$ and $u^0(x) = 3$ for one t on top and one $x \in M$.

6. Assume henceforth that at least two $x \in M$ have $u^0 = 3$ and at least two $y \in M$ have $d^0 = 3$. The x 's and y 's must be disjoint, else we get $K_{3,3}$. Further analysis of this final case yields $T \leq 2$. $\square$

REFERENCES

- [1] K. A. BAKER, P. C. FISHBURN, AND F. S. ROBERTS, *Partial orders of dimension 2, interval orders and interval graphs*, Paper P-4376, The Rand Corporation, Santa Monica, CA, 1970.
- [2] ———, *Partial orders of dimension 2*, Networks, 2 (1972), pp. 11–28.
- [3] K. P. BOGART, I. RABINOVITCH, AND W. T. TROTTER, *A bound on the dimension of interval orders*, J. Combin. Theory, 21 (1976), pp. 319–328.
- [4] R. P. DILWORTH, *A decomposition theorem for partially ordered sets*, Ann. Math., 51 (1950), pp. 161–166.
- [5] B. DUSHNIK AND E. W. MILLER, *Partially ordered sets*, Amer. J. Math., 63 (1941), pp. 600–610.
- [6] P. C. FISHBURN, *Intransitive indifference with unequal indifference intervals*, J. Math. Psych., 7 (1970), pp. 144–149.
- [7] ———, *Interval Orders and Interval Graphs*, John Wiley, New York, 1985.
- [8] ———, *Interval orders and circle orders*, Order, 5 (1988), pp. 225–234.
- [9] P. C. FISHBURN AND W. T. TROTTER, *Angle orders*, Order, 1 (1985), pp. 333–343.
- [10] A. GHOULÂ-HOURI, *Caractérisation des graphes non orientés dont on peut orienter les arêtes de manière à obtenir le graphe d'une relation d'ordre*, C. R. Acad. Sci. Paris, 254 (1962), pp. 1370–1371.
- [11] P. C. GILMORE AND A. J. HOFFMAN, *A characterization of comparability graphs and of interval graphs*, Canad. J. Math., 16 (1964), pp. 539–548.
- [12] M. C. GOLUMBIC, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York, 1980.
- [13] R. L. GRAHAM, B. L. ROTHCHILD, AND J. H. SPENCER, *Ramsey Theory*, John Wiley, New York, 1980.
- [14] T. HIRAGUCHI, *On the dimension of orders*, Sci. Rep. Kanazawa Univ., 4 (1955), pp. 1–20.
- [15] D. KELLY, *The 3-irreducible partially ordered sets*, Canad. J. Math., 29 (1977), pp. 367–383.
- [16] D. KELLY AND W. T. TROTTER, *Dimension theory for ordered sets*, in Ordered Sets, I. Rival, ed., D. Reidel, Dordrecht, 1982, pp. 171–211.
- [17] H. KOMM, *On the dimension of partially ordered sets*, Amer. J. Math., 70 (1948), pp. 507–520.
- [18] R. D. LUCE, *Semiororders and a theory of utility discrimination*, Econometrica, 24 (1956), pp. 178–191.
- [19] O. ORE, *Theory of Graphs*, Amer. Math. Soc., Providence, RI, 1962.
- [20] A. PNUELI, A. LEMPEL, AND S. EVEN, *Transitive orientation of graphs and identification of permutation graphs*, Canad. J. Math., 23 (1971), pp. 160–175.
- [21] I. RABINOVITCH, *The dimension of semiorders*, J. Combin. Theory A, 25 (1978), pp. 50–61.
- [22] D. SCOTT AND P. SUPPES, *Foundational aspects of theories of measurement*, J. Symbolic Logic, 23 (1958), pp. 113–128.
- [23] W. T. TROTTER, *Partially ordered sets*, in Handbook on Combinatorics, R. L. Graham, M. Grötschel, and L. Lovasz, eds., in preparation.
- [24] W. T. TROTTER AND J. I. MOORE, *Characterization problems for graphs, partially ordered sets, lattices, and families of sets*, Discrete Math., 16 (1976), pp. 361–381.
- [25] ———, *The dimension of planar posets*, J. Combin. Theory B, 22 (1977), pp. 54–67.
- [26] J. URRUTIA, *Partial orders and Euclidean geometry*, in Algorithms and Order, I. Rival, ed., Kluwer Academic, Dordrecht, 1989, pp. 387–434.
- [27] D. B. WEST, *Extremal problems in partially ordered sets*, in Ordered Sets, I. Rival, ed., D. Reidel, Dordrecht, 1982, pp. 473–521.
- [28] ———, *Parameters of partial orders and graphs: packing, covering and representation*, in Graphs and Order, I. Rival, ed., D. Reidel, Dordrecht, 1985, pp. 267–350.
- [29] M. YANNAKAKIS, *The complexity of the partial order dimension problem*, SIAM J. Algebraic Discrete Methods, 3 (1982), pp. 351–358.

INTEGER POLYHEDRA ARISING FROM CERTAIN NETWORK DESIGN PROBLEMS WITH CONNECTIVITY CONSTRAINTS*

MARTIN GRÖTSCHEL† AND CLYDE L. MONMA‡

Abstract. In this paper a general integer linear programming model is presented for the important practical problem of designing minimum-cost survivable networks, and this model is related to concepts in graph theory and polyhedral combinatorics. In particular, several interesting special cases of this general model are considered, including the minimum spanning tree problem, the Steiner tree problem, and the minimum cost k -edge connected and k -node connected network design problems. The integer polyhedra associated with these problems are studied, those inequalities from natural ILP-formulations that define facets are identified, the separation problem for these facets is addressed, and how good lower bounds can be obtained from the models studied here is indicated.

Key words. network design, network survivability, connectivity, integer programming, polyhedral combinatorics

AMS(MOS) subject classifications. 05C40, 90C27, 90C50

1. Introduction. For over thirty years, mathematical models arising from the design and analysis of communication networks have been a major focal point for research efforts in the fields of operations research, graph theory, and discrete mathematics. This fertile area has been stimulated by the great practical importance of the associated real-world problems and the wide range applicability of these models on the one hand, and the interesting structural and algorithmic questions and the elegant theoretical results on the other hand. The introduction of new types of networks and new technologies has resulted in a rich variety of models which have been studied over the years.

A recent trend in communication networks is the emergence of fiber optic technology as one of the major components in the “network of the future.” This transmission medium is cost effective, reliable, and provides very high transmission capacity. This combination promises to usher in new telecommunication services requiring large amounts of bandwidth. At the same time, the unique characteristics of this technology imply the need for new network design approaches.

Survivability is an important factor in the design of communication networks. Network survivability is used here to mean the ability to restore service in the event of a catastrophic failure of a network component, such as the complete loss of a transmission link or the failure of a switching node. Service could be restored by routing traffic through other existing network links and nodes, assuming that the design of the network has provided for this additional connectivity. Clearly, a higher level of redundant connectivity results in a greater network survivability and a greater overall network cost. This leads to the problem of designing a minimum-cost network which meets certain required connectivity constraints.

Survivability is a particularly important issue for fiber networks. The high capacity of fiber facilities results in much more sparse network designs with larger amounts of traffic carried by each link than is the case with traditional bandwidth-limited technologies. This increases the potential damage to network services due to link or node failures. It is necessary to trade off the potential for lost revenues and customer goodwill against the

* Received by the editors December 21, 1988; accepted for publication October 10, 1989.

† Institut für Mathematik, Universität Augsburg, Universitätsstr. 8, 8900 Augsburg, West Germany.

‡ Combinatorics and Optimization Group, Bell Communications Research, 445 South Street, Morristown, New Jersey 07960-1910.

extra costs required to increase the network survivability. Recent work on methods for designing survivable fiber communication networks by Monma and Shallcross (1989) concludes that (1) survivability is an important issue for fiber networks, (2) “two-connected” topologies provide a high level of survivability in a cost effective manner, and (3) good heuristic methods exist for quickly generating “near-optimal” networks.

In §§ 2 and 3 of this paper, we present a general integer linear programming model for the network design problem with connectivity constraints and relate this model to concepts in graph theory and polyhedral combinatorics. In the remaining sections we consider important special cases of this general model and study the associated integer polyhedra, identify which natural inequalities define facets, address the separation problem for these facets, and indicate how good lower bounds can be obtained from these models. Section 4 is concerned with the minimum spanning tree problem where a complete linear description of the associated integer polyhedron is given. This result follows easily from matroid theory (see Edmonds (1971)) and is used in later sections. Related work for the Steiner tree problem is described in § 5. Sections 6 and 7 examine the minimum-cost network design problems with edge connectivity constraints and node connectivity constraints, respectively, from a polyhedral point of view. A model which combines both edge and node connectivity constraints is considered in § 8.

2. A general model. In this section, we formalize the network design problems which are being considered in this paper. A set V of *nodes* is given which represents the locations of the switches (offices) which must be interconnected into a network in order to provide the desired services. A collection E of *edges* is also specified which represents the possible pairs of nodes between which a direct transmission link can be placed. We let $G = (V, E)$ be the (undirected) graph of possible direct link connections. Each edge $e \in E$ has a nonnegative *fixed cost* c_e of establishing the direct link connection. The graph G may have parallel edges but contains no loops. (Thus we assume throughout this paper that all graphs considered are loopless but may have parallel edges. Graphs without parallel edges are called *simple*.) The cost of establishing a network consisting of a subset $F \subseteq E$ of edges is the sum of the costs of the individual links contained in F . The goal is to build a minimum-cost network so that the required survivability conditions, which we describe below, are satisfied. We note that the cost here represents setting up the topology for the communication network and includes placing conduits in which to lay the fiber cables, placing the cables into service, and other related costs. We do not consider costs which depend on how the network is implemented such as routing or multiplexing, nor do we consider repeater costs. Although these costs are also important, it is usually the case that a topology is first designed and then these other costs are considered in a second stage of optimization.

For any pair of distinct nodes $s, t \in V$, an $[s, t]$ -*path* P is a sequence of nodes and edges $(v_0, e_1, v_1, e_2, \dots, v_{l-1}, e_l, v_l)$, where each edge e_i is incident with the nodes v_{i-1} and v_i ($i = 1, \dots, l$), where $v_0 = s$ and $v_l = t$, and where no node or edge appears more than once in P . A collection $P_1, P_2, \dots, P_k$ of $[s, t]$ -paths is called *edge-disjoint* if no edge appears in more than one path, and is called *node-disjoint* if no node (except for s and t) appears in more than one path. (Remark: In order to be consistent with standard graph theory we do not consider two parallel edges as two node disjoint paths.)

The survivability conditions require that the network satisfy certain edge and node connectivity requirements. In particular, for each pair of distinct nodes $s, t \in V$, three nonnegative integers r_{st} , k_{st} , and d_{st} are given. The numbers r_{st} represent the *edge survivability requirements* and the numbers k_{st} and d_{st} the *node survivability requirements*, meaning that the network $N = (V, F)$ to be designed has to have the property that, for

each pair $s, t \in V$ of distinct nodes, N must contain at least r_{st} edge disjoint $[s, t]$ -paths and the removal of at most k_{st} nodes (different from s and t) from N must leave at least d_{st} edge disjoint $[s, t]$ -paths. (Clearly, we may assume that $k_{st} \leq |V| - 2$ for all $s, t \in V$, and we will do this throughout this paper.) These conditions ensure that some communication path between s and t will survive a prespecified level of combined failures of both nodes and links. The levels of survivability specified depend on the relative importance placed on maintaining connectivity between different pairs of offices.

Let us now introduce, for each edge $e \in E$, a variable x_e and consider the vector space $\mathbf{R}^E$. Every subset $F \subseteq E$ induces an *incidence vector* $\chi^F = (\chi_e^F)_{e \in E} \in \mathbf{R}^E$ by setting $\chi_e^F \equiv 1$ if $e \in F$, $\chi_e^F \equiv 0$ otherwise, and vice versa, each 0/1-vector $x \in \mathbf{R}^E$ induces a subset $F^x \equiv \{e \in E \mid x_e = 1\}$ of the edge set E of G . If we speak of the incidence vector of a path in the sequel we mean the incidence vector of the edges of the path. We can now formulate the network design problem introduced above as the following integer linear program.

$$(2.1) \quad \min \sum_{ij \in E} c_{ij} x_{ij}$$

subject to

- (i) $\sum_{i \in W} \sum_{j \in V \setminus W} x_{ij} \geq r_{st}$ for all pairs $s, t \in V, s \neq t$ and
for all $W \subseteq V$ with $s \in W, t \notin W$;
- (ii) $\sum_{i \in W} \sum_{j \in V \setminus (Z \cup W)} x_{ij} \geq d_{st}$ for all pairs $s, t \in V, s \neq t$ and
for all $Z \subseteq V \setminus \{s, t\}$ with $|Z| = k_{st}$ and
for all $W \subseteq V \setminus Z$ with $s \in W, t \notin W$;
- (iii) $0 \leq x_{ij} \leq 1$ for all $ij \in E$;
- (iv) x_{ij} integral for all $ij \in E$.

Note that if, for each pair s, t of distinct nodes in V and for each set $Z \subseteq V \setminus \{s, t\}$ with $|Z| = k_{st}$, $N - Z$ contains at least d_{st} edge disjoint $[s, t]$ -paths, then all node survivability requirements are satisfied, i.e., inequalities of type (ii) need not be considered for node sets $Z \subseteq V \setminus \{s, t\}$ with $|Z| < k_{st}$. It follows from Menger's theorem that, for every feasible solution x of (2.1), the subgraph $N = (V, F^x)$ of G defines a network that satisfies the given edge and node survivability requirements.

This formulation is quite general and, as far as we know, this mixture of node and edge survivability requirements has not been considered in the published literature. (The survey paper by Christofides and Whitlock (1981) discusses some related models.) Problem (2.1) is NP-hard as it contains various NP-hard special cases. Some of them will be mentioned in the sequel.

The classical *network synthesis problem* for multiterminal flows is obtained from (2.1) by dropping constraints (ii) and (iv). In the standard formulation of the network synthesis problem the upper bounds $x_e \leq 1$ are not present. But our model allows parallel edges in the underlying direct-link graph, or equivalently, allows the upper bound in constraints (iii) to take on any nonnegative integer values for each edge. This linear programming problem has a number of constraints that is exponential in the number of nodes of G . For the case $c_{ij} = c$ for all $ij \in E$, where c is a constant, Gomory and Hu (1961) found a simple algorithm for its solution. Bland, Goldfarb, and Todd (1981) pointed out that the separation problem for the class of inequalities (i) can be solved in

polynomial time by computing a minimum capacity cut; it thus follows by the ellipsoid method that the classical network synthesis problem can be solved in polynomial time.

The *minimum spanning tree problem* can be phrased as the task to find a minimum cost connected subset $F \subseteq E$ of edges that span V . (See Graham and Hell (1985) for a survey of work on this problem.) This problem can be viewed as a special case of (2.1) as follows. We drop constraints (ii) and set, for all distinct $s, t \in V$, $r_{st} \equiv 1$ in constraints (i). In § 4 we present a complete linear description of the integer polytope for the minimum spanning tree problem which follows easily from matroid theory (see Edmonds (1971)).

Similarly, the closely-related *Steiner tree problem* is to find a minimum cost connected subset $F \subseteq E$ of edges which span a specified subset $S \subseteq V$ of nodes. This problem is a special case of (2.1) where we drop constraints (ii) and set in constraints (i), $r_{st} \equiv 1$ for all $s, t \in S$, and $r_{st} \equiv 0$ otherwise. This problem is NP-hard and will be discussed in § 5.

A graph with at least two nodes is *k-edge* (respectively, *k-node*) *connected* if there are k edge-disjoint (respectively, node-disjoint) paths between every pair of distinct nodes. For our purposes, the graph K_1 consisting of just one node is *k-edge* and *k-node* connected for every natural number k . For a graph $G \neq K_1$, the largest integer k such that G is *k-edge* connected (respectively, *k-node* connected) is denoted by $\lambda(G)$ (respectively, $\kappa(G)$) and is called the *edge connectivity* (respectively, *node connectivity*) of G . Our definition, for instance, implies that, for a graph G with $n \geq 2$ nodes such that every two nodes are linked by p edges, $\kappa(G) = n - 1$ and $\lambda(G) = p(n - 1)$ holds.

There have been many papers in the graph theory literature that study properties of *k-edge* or *k-node* connected graphs. The problem of finding an optimal *k-edge* connected network is a special case of (2.1) where all inequalities (ii) are dropped and where, for all distinct $s, t \in V$, $r_{st} = k$. The problem of finding an optimal *k-node* connected network, $k \leq |V| - 1$, is a special case of (2.1) as follows. We drop the constraints (i) and set, for all distinct $s, t \in V$, $k_{st} \equiv k - 1$ and $d_{st} \equiv 1$.

Monma, Munson, and Pulleyblank (1985) consider the *minimum cost 2-connected network design problem* where the underlying graph $G = (V, E)$ is a complete graph and the costs satisfy the triangle inequality, i.e., $c_{ik} \leq c_{ij} + c_{jk}$ for all $i, j, k \in V$. They show that, for $k = 2$, there exists an optimal *k-edge* (respectively, *k-node*) connected solution whose nodes all have degrees k or $k + 1$, and such that the removal of any 1, 2, $\dots$, or k edges does not result in all connected components still being *k-edge* (respectively, *k-node*) connected. (This is extended to arbitrary $k \geq 2$ by Bienstock, Brickell, and Monma (1987).) They also show that these conditions “characterize” the optimal 2-connected networks in an appropriate sense, but that this is not the case for $k \geq 3$. We return to the *k-connected* network design problems in §§ 6 and 7.

3. The polyhedral approach. The main objective of this paper is to study the network design problem from a polyhedral point of view to see which of the inequalities (i), (ii), (iii) of (2.1) (and which of further classes of inequalities) are suitable choices for a cutting plane approach, i.e., we want to find a tighter LP-relaxation of the IP (2.1) than the one following from (2.1) by dropping the integrality constraints (iv). To do this, we define the following polytope. Let $G = (V, E)$ be a graph, let $E_V \equiv \{st | s, t \in V, s \neq t\}$, and let $r, k, d \in \mathbb{Z}_+^{E_V}$ be given. Then

$$(3.1) \quad \text{CON}(G; r, k, d) \equiv \text{conv} \{x \in \mathbb{R}^E | x \text{ satisfies (i), } \dots, \text{(iv) of (2.1)}\}$$

is the polytope associated with the network design problem given by G and the edge and node survivability requirements r, k , and d . (Above “conv” denotes the convex hull

operator.) In the sequel we will study $\text{CON}(G; r, k, d)$ for various special choices of r , k , and d . Let us mention here a few general properties of $\text{CON}(G; r, k, d)$ that are easy to derive.

Let $G = (V, E)$, $r, k, d \in \mathbb{Z}_+^{E_V}$ be given as above. Extending the terminology of Bollobas (1978), we say that $e \in E$ is *essential with respect to* $(G; r, k, d)$ (short: $(G; r, k, d)$ -essential) if $\text{CON}(G - e; r, k, d) = \emptyset$. In other words, e is essential with respect to $(G; r, k, d)$ if its deletion from G results in a graph such that at least one of the survivability requirements cannot be satisfied. We denote the set of edges in E that are essential with respect to $(G; r, k, d)$ by $\text{ES}(G; r, k, d)$. Clearly, for all subsets $F \subseteq E \setminus \text{ES}(G; r, k, d)$, $\text{ES}(G; r, k, d) \subseteq \text{ES}(G - F; r, k, d)$ holds. Let $\dim(S)$ denote the *dimension* of a set $S \subseteq \mathbb{R}^n$, i.e., the maximum number of affinely independent elements in S minus 1.

THEOREM 3.2. *Let $G = (V, E)$ be a graph and $r, k, d \in \mathbb{Z}_+^{E_V}$ such that $\text{CON}(G; r, k, d) \neq \emptyset$. Then*

$$\begin{aligned} \text{CON}(G; r, k, d) &\subseteq \{x \in \mathbb{R}^E \mid x_e = 1 \text{ for all } e \in \text{ES}(G; r, k, d)\}, \\ \dim(\text{CON}(G; r, k, d)) &= |E| - |\text{ES}(G; r, k, d)|. \end{aligned}$$

Proof. If $e \in E$ is $(G; r, k, d)$ -essential then every vector $x \in \text{CON}(G; r, k, d)$ clearly satisfies $x_e = 1$. So $\text{CON}(G; r, k, d)$ is contained in the affine space of dimension $|E| - |\text{ES}(G; r, k, d)|$ defined by the equations $x_e = 1$, $e \in \text{ES}(G; r, k, d)$. Let $a^T x = \alpha$ be an equation satisfied by all points in $\text{CON}(G; r, k, d)$. We may assume that $a_e = 0$ for all $e \in \text{ES}(G; r, k, d)$. If $e \in E \setminus \text{ES}(G; r, k, d)$ then the incidence vectors of E and $E \setminus \{e\}$ are contained in $\text{CON}(G; r, k, d)$, and thus $a^T \chi^E = a^T \chi^{E \setminus \{e\}} = \alpha$ holds. This implies $a_e = 0$ for all $e \in E \setminus \text{ES}(G; r, k, d)$ and hence $a = 0$. Therefore, $\dim \text{CON}(G; r, k, d) = |E| - |\text{ES}(G; r, k, d)|$. $\square$

An inequality $a^T x \leq \alpha$ is *valid* with respect to a polyhedron P if $P \subseteq \{x \mid a^T x \leq \alpha\}$; the set $F_a = \{x \in P \mid a^T x = \alpha\}$ is called the *face* of P defined by $a^T x \leq \alpha$. If $\dim(F_a) = \dim(P) - 1$ and $F_a \neq \emptyset$ then F_a is a *facet* of P and $a^T x \leq \alpha$ is called *facet-defining* or *facet-inducing*.

THEOREM 3.3. *Let $G = (V, E)$ be a graph and $r, k, d \in \mathbb{Z}_+^{E_V}$ such that $\text{CON}(G; r, k, d) \neq \emptyset$. Then*

- (a) $x_e \leq 1$ defines a facet of $\text{CON}(G; r, k, d)$ if and only if $e \in E \setminus \text{ES}(G; r, k, d)$;
- (b) $x_e \geq 0$ defines a facet of $\text{CON}(G; r, k, d)$ if and only if $e \in E \setminus \text{ES}(G; r, k, d)$ and $\text{ES}(G; r, k, d) = \text{ES}(G - e; r, k, d)$.

Proof. By Theorem (3.2), none of the inequalities $0 \leq x_e \leq 1$, $e \in \text{ES}(G; r, k, d)$, defines a facet of $\text{CON}(G; r, k, d)$.

(a) If $e \in E \setminus \text{ES}(G; r, k, d)$ then the incidence vectors of E and $E \setminus \{f\}$, for each $f \in E \setminus (\text{ES}(G; r, k, d) \cup \{e\})$, satisfy $x_e \leq 1$ with equality and are linearly independent. Thus $x_e \leq 1$ defines a facet of $\text{CON}(G; r, k, d)$.

(b) Suppose $e \in E \setminus \text{ES}(G; r, k, d)$. If there is an edge $f \in \text{ES}(G - e; r, k, d) \setminus \text{ES}(G; r, k, d)$ then, for all $x \in \text{CON}(G; r, k, d)$, $x_e = 0$ implies $x_f = 1$; thus $x_e \geq 0$ does not define a facet of $\text{CON}(G; r, k, d)$. If $\text{ES}(G; r, k, d) = \text{ES}(G - e; r, k, d)$ then let $a^T x \geq \alpha$ be a facet-defining inequality satisfied by all

$$x \in F_e = \{x \in \text{CON}(G; r, k, d) \mid x_e = 0\}.$$

By Theorem 3.2 we may assume that $a_g = 0$ for all $g \in \text{ES}(G; r, k, d)$. Let $f \in E \setminus (\text{ES}(G; r, k, d) \cup \{e\})$, then the incidence vectors of $E \setminus \{e\}$ and $E \setminus \{e, f\}$ are by assumption in $\text{CON}(G; r, k, d)$ and satisfy $a^T x = \alpha$. This implies $a_f = 0$. It fol-

lows that $a^T x \geq \alpha$ is a positive multiple of $x_e \geq 0$, and thus, $x_e \geq 0$ defines a facet of $\text{CON}(G; r, k, d)$. $\square$

Theorems 3.2 and 3.3 solve the dimension problem and characterize the trivial facets. However, these characterizations are (in a certain sense) algorithmically intractable as the next observation shows.

Remark 3.4. The following three problems are NP-hard.

Instance: A graph $G = (V, E)$ and vectors $r, k, d \in \mathbb{Z}_+^{E_V}$.

Question 1: Is $\text{CON}(G; r, k, d)$ empty?

Question 2: Is $e \in E$ an element of $\text{ES}(G; r, k, d)$?

Question 3: What is the dimension of $\text{CON}(G; r, k, d)$?

Proof. Clearly, if we have shown that the first problem is NP-hard, the definition of “essential” and Theorem 3.2 immediately imply that the other two problems are difficult as well.

The NP-hardness of Question 1 follows directly from a recent result of Ling and Kameda (1987). They proved that the following problem is NP-complete.

Instance: A simple graph $G = (V, E)$; two nodes $u, v \in V, u \neq v$; two nonnegative integers a and b .

Question: Does there exist a subset $Z \subseteq V$ with $|Z| = a$ and $u, v \notin Z$ such that $G - Z$ contains at most b edge disjoint $[u, v]$ -paths?

Suppose we could determine in polynomial time whether $\text{CON}(G; r, k, d)$ is empty for the following (very special) choice of $r, k, d \in \mathbb{Z}_+^{E_V}$; $r \equiv 0$; $k_{uv} \equiv a$ and $k_{st} = 0$ otherwise; $d_{uv} \equiv b + 1$ and $d_{st} = 0$ otherwise. Then we could obviously answer the above question in polynomial time. $\square$

However, for most cases of practical interest in the design of optical fiber networks, the sets $\text{ES}(G; r, k, d)$ of essential edges can be determined easily, and thus the trivial LP-relaxation of (2.1) following from (3.2) and (3.3) can be set up without difficulties. (We will comment on this in the sequel.)

In fact, if we can determine $\text{ES}(G; r, k, d)$, we can decompose $\min c^T x, x \in \text{CON}(G; r, k, d)$ into several subproblems as follows. If $G^1, \dots, G^p$ are the components of $G - \text{ES}(G; r, k, d)$ then it is possible to compute vectors r^i, k^i, d^i ($i = 1, \dots, p$) such that G^i contains no (G^i, r^i, k^i, d^i) -essential elements and such that the incidence vector of $F \equiv F^{x^1} \cup F^{x^2} \cup \dots \cup F^{x^p} \cup \text{ES}(G; r, k, d)$ is an optimum solution of $\min c^T x, x \in \text{CON}(G; r, k, d)$, where x^i is an optimum solution of $\min (c^i)^T x, x \in \text{CON}(G^i, r^i, k^i, d^i)$ and c^i is a projection of c into an appropriate space ($i = 1, \dots, p$). The exact procedure is best described in an algorithmic framework, and we leave the details to a forthcoming paper on this subject.

The procedure outlined above shows that we can confine ourselves to the case that $\text{CON}(G; r, k, d)$ is full-dimensional, and we will do so in the following. There is another (technical) reason for this. If polyhedra are not full-dimensional, statements about non-redundancy of certain systems often become quite ugly due to the necessity to exclude equivalent inequalities. This is also true in our case. It is not difficult to derive the results for the lower dimensional cases from the results presented later. But the statements of these theorems are often rather complicated and we want to avoid unnecessary technicalities—there are enough technicalities in this paper anyway.

Before continuing let us remark that there is an easy way to improve upon the formulation of (2.1) by excluding a number of inequalities that are obviously redundant.

Given $G = (V, E)$ and $r, k, d \in \mathbb{Z}_+^{E_V}$, let us extend the functions r and d to functions operating on sets by setting

$$(3.5) \quad r(W) \equiv \max \{r_{st} \mid s \in W, t \in V \setminus W\} \text{ for } W \subseteq V$$

and

$$(3.6) \quad d(Z, W) \equiv \max \{d_{st} \mid s \in W, t \in V \setminus (Z \cup W), k_{st} \geq |Z|\} \text{ for } Z, W \subseteq V.$$

We call a pair (Z, W) , $Z, W \subseteq V$, *eligible* (with respect to k) if $Z \cap W = \emptyset$ and $|Z| = k_{st}$ for at least one pair of nodes with $s \in W$ and $t \in V \setminus (Z \cup W)$. Then $\text{CON}(G; r, k, d)$ is clearly contained in the solution set of the following system of equations and inequalities.

$$(3.7) \quad \begin{aligned} \text{(i)} \quad & \sum_{i \in W} \sum_{j \in V \setminus W} x_{ij} \geq r(W) && \text{for all } W \subseteq V, \emptyset \neq W \neq V; \\ \text{(ii)} \quad & \sum_{i \in W} \sum_{j \in V \setminus (W \cup Z)} x_{ij} \geq d(Z, W) && \text{for all eligible pairs } (Z, W) \\ & && \text{of subsets of } V; \\ \text{(iii)} \quad & x_{ij} \leq 1 && \text{for all } ij \in E \setminus \text{ES}(G; r, k, d); \\ \text{(iv)} \quad & x_{ij} = 1 && \text{for all } ij \in \text{ES}(G; r, k, d); \\ \text{(v)} \quad & x_{ij} \geq 0 && \text{for all } ij \in E \setminus \text{ES}(G; r, k, d) \text{ with} \\ & && \text{ES}(G; r, k, d) = \text{ES}(G - e; r, k, d). \end{aligned}$$

In the subsequent sections of this paper we will investigate in more detail these and other inequalities for special choices of r, k , and d .

4. Connectivity. We will now consider one of the easiest special cases of our network design problem (2.1). This case, however, will provide further insight and yield a new class of interesting inequalities.

Given a graph $G = (V, E)$ and $W \subseteq V$, the edge set

$$\delta(W) \equiv \{ij \in E \mid i \in W, j \in V \setminus W\}$$

is called the *cut* (induced by W). (We will write $\delta_G(W)$ to make clear—in case of possible ambiguities—with respect to which graph the cut induced by W is considered.) For $W, W' \subseteq V$ with $W \cap W' = \emptyset$ we define $[W : W'] \equiv \{ij \in E \mid i \in W, j \in W'\}$. So $\delta(W) = [W : V \setminus W]$. For $W \subseteq V$ we set $E(W) \equiv \{ij \in E \mid i, j \in W\}$.

In this section we assume that the underlying graph $G = (V, E)$ is connected. As before, multiple edges are allowed, but loops are not. Let $\mathbf{1}$ be the vector (of appropriate dimension) with all components equal to 1. Set

$$(4.1) \quad \text{CON}(G) := \text{CON}(G; \mathbf{1}, 0, 0).$$

In other words, $\text{CON}(G)$ is the convex hull of all feasible solutions of the system

$$(4.2) \quad \begin{aligned} \text{(i)} \quad & x(\delta(W)) \geq 1 && \text{for all } W \subseteq V, \emptyset \neq W \neq V; \\ \text{(ii)} \quad & 0 \leq x_e \leq 1 && \text{for all } e \in E; \\ \text{(iii)} \quad & x_e \in \{0, 1\} && \text{for all } e \in E, \end{aligned}$$

where, from now on, we use the symbol $x(F)$ to abbreviate the sum $\sum_{e \in F} x_e$. Another way to state (4.1) is

$$\text{CON}(G) = \text{conv} \{x^F \in \mathbf{R}^E \mid (V, F) \text{ is a connected subgraph of } G\}.$$

That is why we call $\text{CON}(G)$ the *connected subgraph polytope* of G . It is easy (and well known how) to solve

$$(4.3) \quad \min c^T x, x \in \text{CON}(G).$$

This goes as follows. Let F be the set of edges e with nonpositive weight c_e . If (V, F) is connected, stop. Otherwise contract the components of (V, F) to single nodes and compute a minimum spanning tree T in the resulting graph. $T \cup F$ yields an optimum solution to (4.3). It is also well known that the solution set of the LP-relaxation of (4.2), i.e., the polyhedron defined by (i), (ii) of (4.2), is not integral in general, and that a complete linear description of $\text{CON}(G)$ can be easily derived (see e.g., Cornuéjols, Fonlupt, and Naddef (1985)) from Edmonds' characterization of matroid polytopes—see Edmonds (1970, 1971). We give this transformation here for sake of completeness.

Recall that the bases of the graphic matroid on G are the spanning trees of G , that the bases of the cographic matroid are the complements of spanning trees, and that a set is independent in a matroid if it is contained in a basis. Let r (r^* , respectively) denote the rank function of the graphic (cographic, respectively) matroid on $G = (V, E)$. Then, for $F \subseteq E$, $r(F) = |V| - c_F$, where c_F is the number of components of (V, F) , and

$$(4.4) \quad r^*(F) = |F| + r(E \setminus F) - r(E) = |F| - |V| + 1 + r(E \setminus F).$$

Let $\text{IND}^*(G)$ be the convex hull of the incidence vectors χ^F , where $F \subseteq E$ is independent in the cographic matroid of G , i.e.,

$$(4.5) \quad \text{IND}^*(G) = \text{conv} \{ \chi^F \in \mathbf{R}^E \mid \exists \text{ spanning tree } T \text{ such that } F \subseteq E \setminus T \}.$$

The above definitions imply:

$$(4.6) \quad \text{CON}(G) = \{ \mathbf{1} - y \in \mathbf{R}^E \mid y \in \text{IND}^*(G) \},$$

$$(4.7) \quad \text{IND}^*(G) = \{ \mathbf{1} - x \in \mathbf{R}^E \mid x \in \text{CON}(G) \}.$$

A subset $F \subseteq E$ is called r^* -closed if $r^*(F) < r^*(F \cup \{e\})$ for all $e \in E \setminus F$, and F is called r^* -inseparable if there is no partition F_1, F_2 of F such that $r^*(F) = r^*(F_1) + r^*(F_2)$. (A family $S_1, \dots, S_m$ of subsets of a set S is called a *partition* of S if $S_i \neq \emptyset$, $i = 1, \dots, m$; $S_i \cap S_j = \emptyset$, $1 \leq i < j \leq m$; and $S_1 \cup \dots \cup S_m = S$.) Let $B(G) \subseteq E$ denote the set of bridges of G (a *bridge* is an edge that forms a cut). It follows from Edmonds' results on matroid polytopes that

$$(4.8) \quad \begin{aligned} \text{IND}^*(G) = \{ y \in \mathbf{R}^E \mid & y_e = 0 && \text{for all } e \in B(G); \\ & y_e \geq 0 && \text{for all } e \in E \setminus B(G); \\ & y(F) \leq r^*(F) && \text{for all } F \subseteq E, \text{ with } F \\ & && r^*\text{-closed and } r^*\text{-inseparable} \}. \end{aligned}$$

In fact, the linear description of $\text{IND}^*(G)$ given above is nonredundant. Using relation (4.6) and formula (4.4) we obtain the following nonredundant description of the connected subgraph polytope:

$$(4.9) \quad \begin{aligned} \text{CON}(G) = \{ x \in \mathbf{R}^E \mid & x_e = 1 && \text{for all } e \in B(G); \\ & x_e \leq 1 && \text{for all } e \in E \setminus B(G); \\ & x(F) \geq |V| - 1 - r(E \setminus F) && \text{for all } F \subseteq E \text{ with } F \\ & && r^*\text{-closed and} \\ & && r^*\text{-inseparable} \}. \end{aligned}$$

Observe that the bridges of a graph are exactly the $(G; \mathbf{1}, 0, 0)$ -essential edges and recall that a connected graph is bridgeless if and only if it is 2-edge connected. It is a nice exercise to translate the matroid properties " r^* -closed" and " r^* -inseparable" into the

language of graph theory. The result (ignoring the technical details coming up by considering bridges) is the following theorem.

THEOREM 4.10. *Let $G = (V, E)$ be a 2-edge connected graph. Then $\text{CON}(G)$ is full-dimensional and*

- (i) $\frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p - 1$ for all partitions $V_1, \dots, V_p$ of V , $p \geq 2$, such that each subgraph $(V_i, E(V_i))$ is 2-edge connected and the graph obtained by contracting every V_i , $i = 1, \dots, p$ to a single node is 2-node connected;
- (ii) $x_e \leq 1$ for all $e \in E$;
- (iii) $x_e \geq 0$ for all $e \in E$ such that $(G - e)$ is 2-edge connected

is a complete and nonredundant linear characterization of $\text{CON}(G)$. $\square$

Since we can optimize any linear function over $\text{CON}(G)$ in polynomial time, it follows from the ellipsoid method (see Grötschel, Lovász, and Schrijver (1981, 1988)) that the separation problem for the linear system (i)–(iii) of (4.10) can be solved in polynomial time. In fact, a specialization of Cunningham's algorithm for the separation problem for matroid polytopes (see Cunningham (1984)) yields a combinatorial separation algorithm for this system.

5. Steiner trees. Let $G = (V, E)$ be a connected graph, and let S , $|S| \geq 2$, be a subset of the node set. S is called the set of *terminal* nodes, $V \setminus S$ is called the set of *Steiner nodes*. Define a vector $r^S \in \mathbf{Z}_+^{E_V}$ by setting $r_{st}^S = 1$ for all $s, t \in S$, $s \neq t$ and $r_{st}^S = 0$ else, and let

$$(5.1) \quad \text{CON}(G; S) \equiv \text{CON}(G; r^S, 0, 0).$$

Then $\text{CON}(G; S)$ is the convex hull of all incidence vectors x^F such that all nodes in S belong to the same component of (V, F) . Another way to say this is that $\text{CON}(G; S)$ is the convex hull of all incidence vectors x^F where F contains a Steiner tree of G (with $V \setminus S$ being the set of Steiner nodes). Thus for $c \in \mathbf{R}_+^E$ every optimum vertex solution of

$$(5.2) \quad \min c^T x, \quad x \in \text{CON}(G; S)$$

yields an optimum Steiner tree. The LP-relaxation for (5.2) that is provided by (2.1), respectively, (3.7) and our special choice of r , k , and d has the following constraints:

$$(5.3) \quad \begin{aligned} & \text{(i) } x(\delta(W)) \geq 1 \quad \text{for all } W \subseteq V \text{ such that} \\ & \quad W \cap S \neq \emptyset \text{ and } S \setminus W \neq \emptyset; \\ & \text{(ii) } 0 \leq x_e \leq 1 \quad \text{for all } e \in E. \end{aligned}$$

For $|S| = 2$, say $S = \{s, t\}$, the integral solutions of (5.3) are precisely the incidence vectors of edge sets $F \subseteq E$ that contain an $[s, t]$ -path. In fact, it is easy to derive from any shortest path algorithm that the polyhedron defined by (5.3) is integral; thus in case $|S| = 2$, $\text{CON}(G; S) = \{x \in \mathbf{R}^E \mid x \text{ satisfies (5.3) (i) and (ii)}\}$ holds.

Let us call an edge $e \in E$ a *Steiner bridge* if $G - e$ contains no $[s, t]$ -path for some nodes $s, t \in S$. We denote the set of Steiner bridges of a graph G by $B(G; S)$. It is easy to see that $B(G; S) = ES(G; r^S, 0, 0)$. For $S = V$ the Steiner bridges are just the bridges of G , i.e., $B(G; V) = B(G)$; for $|S| = 2$, say $S = \{s, t\}$, the Steiner bridges are the edges that are on every $[s, t]$ -path. Such edges are called $[s, t]$ -bridges. Let us denote the subgraph of G induced by the node set W by $G[W]$, i.e., $G[W] = (W, E(W))$. Using

this terminology, a nonredundant characterization of the convex hull of the incidence vectors of edge sets containing an $[s, t]$ -path can be derived easily.

THEOREM 5.4. *Let $G = (V, E)$ be a connected graph, let $s, t \in V$ be two different nodes and assume that G contains no $[s, t]$ -bridge. Then the dimension of $\text{CON}(G; \{s, t\})$ is equal to $|E|$ and the following system is a complete and nonredundant characterization of $\text{CON}(G; \{s, t\})$:*

- (i) $x(\delta(W)) \geq 1$ for all cuts $\delta(W)$ such that
 $s \in W, t \in V \setminus W$,
and $G[W]$ and $G[V \setminus W]$ are connected;
- (ii) $x_e \leq 1$ for all $e \in E$;
- (iii) $x_e \geq 0$ for all $e \in E$ such that $G - e$ contains no $[s, t]$ -bridge.

The system (5.3) is in general not a complete description of $\text{CON}(G; S)$ if S is any set of terminal nodes with $|S| \geq 3$ and G a general graph, not even for the other “extreme and simple” case where $S = V$, as Theorem (4.10) shows (note that $\text{CON}(G) = \text{CON}(G; V)$).

A natural way to generalize the system given in (4.10) to the Steiner tree problem is to consider the following system of inequalities:

$$(5.5) \quad \begin{aligned} & \text{(i)} \quad \frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p - 1 \quad \text{for all partitions } V_1, \dots, V_p \text{ of } V, p \geq 2, \\ & \quad \quad \quad \text{such that } |V_i \cap S| \geq 1 \text{ for } i = 1, \dots, p; \\ & \text{(ii)} \quad 0 \leq x_e \leq 1 \quad \text{for all } e \in E. \end{aligned}$$

Clearly, all inequalities of the system (5.5) are valid for $\text{CON}(G; S)$; but—as observed by White, Farber, and Pulleyblank (1985)—they are not sufficient to describe $\text{CON}(G; S)$, not even for graphs as simple as series-parallel graphs.

System (5.5) seems, however, to be a reasonable LP-relaxation of the Steiner tree problem as the following result shows.

THEOREM 5.6. *Let $G = (V, E)$ be a connected graph, let $S \subseteq V$ be a set of terminal nodes and assume that G contains no Steiner bridge. Let $V_1, \dots, V_p, p \geq 2$, be a partition of V such that $V_i \cap S \neq \emptyset$ for $i = 1, \dots, p$. Then*

$$\frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p - 1$$

defines a facet of $\text{CON}(G; S)$ if and only if

- (a) $G[V_i]$ is connected for $i = 1, \dots, p$;
- (b) $G[V_i]$ contains no Steiner bridge with respect to the set $S_i := S \cap V_i$ of terminal nodes for $i = 1, \dots, p$;
- (c) the graph $\hat{G} = (\hat{V}, \hat{E})$ obtained from G by contracting each node set V_i to single node is 2-node connected.

(Comment: If $|S_i| = 1$, no edge of $G[V_i]$ is a Steiner bridge.)

Proof. Suppose one of the graphs $G[V_i]$, say $G[V_1]$, is not connected. Let V'_1 be the node set of a component of $G[V_1]$ such that $(V_1 \setminus V'_1) \cap S \neq \emptyset$. Since G is connected there is a node set $V_j, j \in \{2, \dots, p\}$, say V_2 , such that V'_1 and V_2 are connected by an edge. But then the inequality $\frac{1}{2}(x(\delta(V_1 \setminus V'_1)) + x(\delta(V_2 \cup V'_1)) + \sum_{i=3}^p x(\delta(V_i))) \geq p - 1$ belongs to class (5.5) (ii) and its sum with $x_e \geq 0$ for all $e \in [V'_1 : V_2]$ is equal to $\frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p - 1$. So the latter inequality does not define a facet. We may thus assume that $G[V_i]$ is connected for all i .

Suppose $G[V_i]$, for some $i \in \{1, \dots, p\}$, contains a Steiner bridge, say e , with respect to S_i . We will show that every $D \subseteq E$ with $\chi^D \in \text{CON}(G; S)$ and satisfying $a^T x \equiv \frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p-1$ with equality contains e . This implies that $a^T x \geq p-1$ does not define a facet. Suppose there exists $D \subseteq E \setminus \{e\}$ with $\chi^D \in \text{CON}(G; S)$ and $a^T \chi^D = p-1$. Since e is a Steiner bridge of $G[V_i]$, D must contain a path linking two Steiner nodes contained in different components of $G[V_i] - e$. Thus $D' \equiv D \cup E(V_i)$ satisfies $\chi^{D'} \in \text{CON}(G; S)$ and $a^T \chi^{D'} = p-1$ and contains a cycle C containing e and an edge, say f , of $\delta(V_i)$. But then $D'' \equiv (D' \setminus \{f\}) \cup \{e\}$ satisfies $\chi^{D''} \in \text{CON}(G; S)$ and $a^T \chi^{D''} = p-2$, a contradiction.

Suppose $\hat{G}$ is not 2-node connected. Let $\hat{V} = \{v_1, \dots, v_p\}$, where v_i is the node obtained by contracting V_i , $i = 1, \dots, p$. We may assume that v_1 is a cut-node such that $\{v_2, \dots, v_c\}$ is the node set of one component of $\hat{G} - v_1$. Set $W_1 \equiv V_1 \cup \bigcup_{i=c+1}^p v_i$ and $W_2 \equiv V_1 \cup \bigcup_{i=2}^c v_i$. Then $\frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p-1$ is the sum of the two valid inequalities $\frac{1}{2}(x(\delta(W_1)) + \sum_{i=2}^c x(\delta(V_i))) \geq c-1$ and $\frac{1}{2}(x(\delta(W_2)) + \sum_{i=c+1}^p x(\delta(V_i))) \geq p-c$.

This shows that if one of the conditions (a), (b), (c) is not satisfied then the given inequality does not define a facet of $\text{CON}(G; S)$.

Now let $a^T x \equiv \frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p-1$ be an inequality of type (5.5) (ii) such that (a), (b), and (c) are satisfied. Let $b^T x = \beta$ be an equation such that $F_a \equiv \{x \in \text{CON}(G; S) \mid a^T x = p-1\} \subseteq F_b \equiv \{x \in \text{CON}(G; S) \mid b^T x = \beta\}$ and such that F_b is a facet of $\text{CON}(G; S)$.

Note that, by construction, every spanning tree T of $\hat{G}$ corresponds to a forest (also denoted by T) of G such that, for each subset B of $A \equiv \bigcup_{i=1}^p E(V_i)$, the incidence vector of $B \cup T$ satisfies $a^T x \geq p-1$ with equality.

We first prove that $b_e = 0$ for all $e \in A$. Let $e \in E(V_i)$ for some $i \in \{1, \dots, p\}$. Choose two nodes $s \in S \cap V_i$ and $t \in S \cap (V \setminus V_i)$. Since G contains no Steiner bridge there exists an $[s, t]$ -path P in G not containing e . Choose an edge $f \in P \cap \delta(V_i)$ and construct a spanning tree T of $\hat{G}$ containing f . Set $D \equiv T \cup A$ and $D_e \equiv D \setminus \{e\}$. Since $G[V_j]$ is connected for every j and since e is not a Steiner bridge of $G[V_i]$, by construction, $\chi^D, \chi^{D_e} \in \text{CON}(G; S)$ and $a^T \chi^D = a^T \chi^{D_e} = p-1$. Thus $b^T \chi^D = b^T \chi^{D_e}$ which implies $b_e = 0$.

Let e, f be different elements of $\hat{E} (= E \setminus A)$. Since $\hat{G}$ is 2-node connected there exists a cycle C of $\hat{G}$ containing e and f . Let T be a spanning tree of $\hat{G}$ containing $C \setminus \{e\}$ (but not f). Then $T' \equiv (T \setminus \{e\}) \cup \{f\}$ is also a spanning tree of $\hat{G}$. Set $D \equiv T \cup A$ and $D' \equiv T' \cup A$. Clearly $\chi^D, \chi^{D'} \in \text{CON}(G; S)$ and $0 = b^T \chi^D - b^T \chi^{D'} = b_e - b_f$. This implies that $b^T x$ is a multiple of $a^T x$ which proves that $a^T x \geq p-1$ defines a facet of $\text{CON}(G; S)$. $\square$

Theorems 3.2, 3.3, and 5.6 immediately give the following result.

COROLLARY 5.7. *Let $G = (V, E)$ be a connected graph, $S \subseteq V$ a set of at least two terminal nodes and assume that G contains no Steiner bridge. Then $\dim(\text{CON}(G; S)) = |E|$ and the following system is a nonredundant system of facet-defining inequalities for $\text{CON}(G; S)$:*

- (i) $\frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p-1$ for all partitions $V_1, \dots, V_p$ of V , $p \geq 2$, such that $\hat{G}$ is 2-node connected and, for $i = 1, \dots, p$, $V_i \cap S \neq \emptyset$ and $G[V_i]$ is connected and contains no Steiner bridge with respect to $V_i \cap S$;
- (ii) $x_e \leq 1$ for all $e \in E$;

- (iii) $x_e \geq 0$ for all $e \in E$ such that $G - e$ contains no Steiner bridge.

Note that the nonredundancy parts of Theorems 4.10 and 5.4 follow directly from Corollary (5.7).

Based on the paper of Prodon, Liebling, and Gröflin (1985), Prodon (1985) has generalized the inequality system (5.5) (i) for $\text{CON}(G; S)$ to the system (5.9) defined below. We give a polyhedral proof of the validity of these inequalities.

PROPOSITION 5.8. *Let $G = (V, E)$ be a graph and $S \subseteq V$, $|S| \geq 2$, a set of terminal nodes. Let $\mathcal{F}$ be a set of subsets of V such that*

- (a) $|\mathcal{F}| \geq 1$,
 (b) $U \cap S \neq \emptyset$ for all $U \in \mathcal{F}$,
 (c) $(V \setminus \bigcup_{U \in \mathcal{F}} U) \cap S \neq \emptyset$.

For each edge $e = uv \in E$, set

$$\Delta(\mathcal{F}; e, u, v) \equiv |\{U \in \mathcal{F} \mid u \in U, v \notin U\}|,$$

$$a_e(\mathcal{F}) \equiv \max \{ \Delta(\mathcal{F}; e, u, v), \Delta(\mathcal{F}; e, v, u) \},$$

and define

$$a(\mathcal{F}) \equiv (a_e(\mathcal{F}))_{e \in E} \in \mathbf{R}^E.$$

Then

$$(5.9) \quad a(\mathcal{F})^T x \geq |\mathcal{F}|$$

is valid with respect to $\text{CON}(G; S)$.

Proof. We prove the validity of (5.9) by induction on $|\mathcal{F}|$. For $|\mathcal{F}| = 1$, say $\mathcal{F} = \{W\}$, (5.9) is nothing but the (valid) cut inequality $x(\delta(W)) \geq 1$ already considered in (5.3) (i).

We now assume that (5.9) is valid for $\text{CON}(G; S)$ for all set systems satisfying (a), (b), (c) with at most $p \geq 1$ elements. Let $\mathcal{F} \subseteq 2^V$ be a set system satisfying (a), (b), (c) with $p + 1$ elements. For ease of notation, let us set $\bar{V} \equiv V \setminus (\bigcup_{U \in \mathcal{F}} U)$.

Let I be the set of all (unordered) pairs $\{U, W\}$ with $U, W \in \mathcal{F}$, such that there is an edge $e \in E$ with one endnode in U (or in W) and the other endnode in $W \setminus U$ (or in $U \setminus W$). Moreover, let J be the set of nodesets $U \in \mathcal{F}$ such that there exists an edge $e \in E$ with one endnode in U and the other endnode in $\bar{V}$.

We define new set systems as follows. For $\{U, W\} \in I$, set

$$\mathcal{F}_{UW} \equiv (\mathcal{F} \setminus \{U, W\}) \cup \{U \cup W\},$$

and for $U \in J$, set $\mathcal{F}_U \equiv \mathcal{F} \setminus \{U\}$. Clearly, each of the systems $\mathcal{F}_{UW}$ and $\mathcal{F}_U$ has cardinality p and satisfies (a), (b), and (c). Let

$$b^T x \equiv \sum_{\{U, W\} \in I} a(\mathcal{F}_{UW})^T x + \sum_{U \in J} a(\mathcal{F}_U)^T x.$$

By induction hypothesis, the inequalities of type (5.9) associated with the systems $\mathcal{F}_{UW}$ and the systems $\mathcal{F}_U$ are valid for $\text{CON}(G; S)$. Thus $b^T x \geq (|I| + |J|)p$ is also valid.

We will now prove (componentwise) that $(|I| + |J| - 1)a(\mathcal{F})^T x \geq b^T x$. From this observation we can conclude that $a(\mathcal{F})^T x \geq (|I| + |J|)/(|I| + |J| - 1)p$, and thus validity of $a(\mathcal{F})^T x \geq |\mathcal{F}|$ follows by rounding up the right-hand side.

By definition, for every edge $e \in E$, $a_e(\mathcal{F}) \geq a_e(\mathcal{F}_{UW}) \geq a_e(\mathcal{F}) - 1$ for all $\{U, W\} \in I$, and $a_e(\mathcal{F}) \geq a_e(\mathcal{F}_U) \geq a_e(\mathcal{F}) - 1$ for all $U \in J$; moreover, $a(\mathcal{F}_{UW}) \geq$

0 for all $\{U, W\} \in I$ and $a_e(\mathcal{F}_U) \geq 0$ for all $U \in J$. Thus, for every edge $e \in E$ with $a_e(\mathcal{F}) = 0$, we have $b_e = 0$.

Let $e = uv \in E$ be an edge with $a_e(\mathcal{F}) > 0$.

Case 1. $u, v \notin \bar{V}$.

Case 1.1. $\Delta(\mathcal{F}; e, u, v) > \Delta(\mathcal{F}; e, v, u)$. Set $\mathcal{F}' \equiv \{U' \in \mathcal{F} \mid u \in U', v \notin U'\}$. By definition there are at least $a_e(\mathcal{F})$ sets in $\mathcal{F}'$, and moreover, since $v \notin \bar{V}$ there is at least one set, say W' , in $\mathcal{F} \setminus \mathcal{F}'$ with $v \in W'$. Set $I' \equiv \{\{U', W'\} \mid U' \in \mathcal{F}'\}$. Then $I' \subseteq I$ and $a_e(\mathcal{F}_{U'W'}) = a_e(\mathcal{F}) - 1$ for all $\{U', W'\} \in I'$. Thus

$$\begin{aligned} b_e &= \sum_{\{U, W\} \in I} a_e(\mathcal{F}_{UW}) + \sum_{U \in J} a_e(\mathcal{F}_U) \leq (|I| + |J|)a_e(\mathcal{F}) - |I'| \\ &\leq (|I| + |J| - 1)a_e(\mathcal{F}). \end{aligned}$$

The case $\Delta(\mathcal{F}; e, v, u) > \Delta(\mathcal{F}; e, u, v)$ follows by symmetry.

Case 1.2. $\Delta(\mathcal{F}; e, v, u) = \Delta(\mathcal{F}; e, u, v)$. Let $\mathcal{F}' \equiv \{U' \in \mathcal{F} \mid u \in U', v \notin U'\}$ and $\mathcal{F}'' \equiv \{U'' \in \mathcal{F} \mid u \notin U'', v \in U''\}$. By assumption $|\mathcal{F}'| = |\mathcal{F}''| = a_e(\mathcal{F})$. Set $I' \equiv \{\{U', U''\} \mid U' \in \mathcal{F}', U'' \in \mathcal{F}''\}$. Then $|I'| = a_e(\mathcal{F})^2$ and $a_e(\mathcal{F}_{U'U''}) = a_e(\mathcal{F}) - 1$ for all $\{U', U''\} \in I'$. This implies as before $b_e \leq (|I| + |J|)a_e(\mathcal{F}) - |I'| \leq (|I| + |J| - 1)a_e(\mathcal{F})$.

Case 2. $e \in \delta(\bar{V})$, say $u \in V \setminus \bar{V}$ and $v \in \bar{V}$. Let $\mathcal{F}' \equiv \{U \in \mathcal{F} \mid u \in U\}$. Then by definition $a_e(\mathcal{F}) = \Delta(\mathcal{F}; e, u, v) = |\mathcal{F}'|$ and $\mathcal{F}' \subseteq J$. Obviously, $a_e(\mathcal{F}_U) = a_e(\mathcal{F}) - 1$ for all $U \in \mathcal{F}'$, and hence

$$b_e \leq (|I| + |J|)a_e(\mathcal{F}) - |\mathcal{F}'| \leq (|I| + |J| - 1)a_e(\mathcal{F}).$$

This finishes the proof. $\square$

It is easy to find some necessary conditions for an inequality (5.9) to define a facet of $\text{CON}(G)$. But it is currently unknown which of the inequalities (5.9) defines facets of $\text{CON}(G; S)$.

Given a graph $G = (V, E)$, a node set $S \subseteq V$, and a vector $y \in \mathbf{R}^E$ (we may assume $0 \leq y \leq \mathbf{1}$), it is easy to solve the separation problem for y and the inequality system (5.3) (i) by computing a Gomory–Hu tree for the graph $G = (V, E)$ with the values y_e , $e \in E$, considered as edge capacities. (The Gomory–Hu method for finding a minimum capacity cut is, for instance, described in Hu (1969) or Grötschel, Lovász, and Schrijver (1988).) The Gomory–Hu tree contains an edge with capacity smaller than one whose removal separates two nodes in S if and only if y violates at least one of the inequalities (5.3) (i). This implies (see, e.g., Grötschel, Lovász, and Schrijver (1988)) that the LP-relaxation of (5.2) coming from (5.3) can be solved in polynomial time.

The Gomory–Hu tree can be used to find some violated inequalities of type (5.5) (i) heuristically. But a polynomial time separation procedure for the system (5.5) is not known. The same holds for Prodon's system of inequalities defined in Proposition 5.8.

6. Edge connectivity. By setting $r_{st} \equiv k$ (k a positive integer), and $d_{st} \equiv k_{st} \equiv 0$ for all $s, t \in V$, $s \neq t$ we obtain the following special case of (2.1), respectively (3.7):

$$\begin{aligned} (6.1) \quad & \min c^T x \\ & \text{(i) } x(\delta(W)) \geq k \quad \text{for all } W \subseteq V, \emptyset \neq W \neq V; \\ & \text{(ii) } 0 \leq x_e \leq 1 \quad \text{for all } e \in E; \\ & \text{(iii) } x_e \in \{0, 1\} \quad \text{for all } e \in E. \end{aligned}$$

The feasible solutions of (6.1) are exactly the incidence vectors of all edge sets $C \subseteq E$ such that (V, C) is k -edge connected (i.e., every pair of nodes of G is linked by k edge

disjoint paths). So (6.1) is asking for a minimum cost spanning k -edge connected subgraph of G . To simplify notation in the following, we will just speak of a k -edge connected edge set C and mean that (V, C) is a (spanning) k -edge connected subgraph of G . Let us set

$$\text{ECON}(G; k) \equiv \text{conv} \{ \chi^C \in \mathbf{R}^E \mid (V, C) \text{ is } k\text{-edge connected} \}.$$

$\text{ECON}(G; k)$ is called the *polytope of k -edge connected subgraphs of G* . Clearly, $\text{ECON}(G; k) = \text{CON}(G; k\mathbf{1}, 0, 0)$, and for $k = 1$, we have $\text{ECON}(G; 1) = \text{CON}(G)$ —see § 4. The dominant of this polytope, i.e., the polyhedron $\text{ECON}(G; k) + \mathbf{R}_+^E$, has been considered in Cornuéjols, Fonlupt, and Naddef (1985) who showed, among other results, that $\text{ECON}(G; 2) + \mathbf{R}_+^E = \{x \in \mathbf{R}^E \mid x \geq 0, x(\delta(W)) \geq 2 \text{ for all } W \subseteq V, \emptyset \neq W \neq V\}$ if G is series-parallel.

We will now give a technical characterization of those inequalities of type (6.1) (i) that define facets of $\text{ECON}(G; k)$. To do this we introduce further notation. An edge e of a graph $G = (V, E)$ is called *k -edge-essential* if $G - e$ is not k -edge connected. So the k -edge-essential edges of G are exactly the edges that are $(G; k\mathbf{1}, 0, 0)$ -essential. By Theorem (3.2), $\text{ECON}(G; k)$ has dimension $|E|$ if and only if G is k -edge connected and contains no k -edge-essential edge. By Menger's theorem the latter condition is equivalent to G and is $(k + 1)$ -edge connected.

THEOREM 6.2. *Let $G = (V, E)$ be a $(k + 1)$ -edge connected graph, $k \geq 1$. Then, for $W \subseteq V, \emptyset \neq W \neq V$,*

$$x(\delta(W)) \geq k$$

defines a facet of $\text{ECON}(G; k)$ if and only if

- (a) *for each edge $e \in E \setminus \delta(W)$, there exists a set $C \subseteq \delta(W)$ such that*
 - (a₁) $|C| = k$, and
 - (a₂) $C \cup (E \setminus (\delta(W) \cup \{e\}))$ *is k -edge connected; and*
- (b) *there exist edge sets $C_1, \dots, C_s \subseteq \delta(W)$, where $s = |\delta(W)|$, such that*
 - (b₁) $|C_i| = k, i = 1, \dots, s$,
 - (b₂) $C_i \cup (E \setminus (\delta(W)))$ *is k -edge connected, and*
 - (b₃) *the $s \times s$ -matrix M whose columns are the incidence vectors $\chi^{C_i} \in \mathbf{R}^{\delta(W)}$ is nonsingular.*

Proof. Suppose (a) and (b) are satisfied. Set $a^T x \equiv x(\delta(W))$ and let $b^T x \geq \beta$ define a facet F_b of $\text{ECON}(G; k)$ that contains the face $F_a \equiv \{x \in \text{ECON}(G; k) \mid a^T x = k\}$.

Let $e \in E \setminus \delta(W)$ and let $C_e \equiv C \cup (E \setminus (\delta(W) \cup \{e\}))$ be the k -edge connected subset of E existing by (a). Then $D \equiv C_e \cup \{e\}$ is also k -edge connected and, by (a₁) we have $a^T \chi^{C_e} = a^T \chi^D = k$. Thus, since $F_a \subseteq F_b$, $b^T \chi^{C_e} = b^T \chi^D$ holds. This implies $b_e = 0$. It follows that $b_e = 0$ for all $e \in E \setminus \delta(W)$.

The incidence vectors of the k -edge connected sets $D_i \equiv C_i \cup (E \setminus \delta(W))$, $i = 1, \dots, s$, satisfy $a^T \chi^{D_i} = k$ and hence $b^T \chi^{D_i} = \beta$. Consider the equation $y^T M = \beta \mathbf{1}^T$. Clearly, the vectors $(\beta/k)\mathbf{1}^T$ and $\bar{b}^T$ in $\mathbf{R}^{\delta(W)}$ (where $\bar{b}$ is the vector obtained from b by deleting the components $E \setminus \delta(W)$) are solutions of this equation. Since M is nonsingular, $\bar{b} = (\beta/k)\mathbf{1}$ has to hold. This implies that $a = (k/\beta)\bar{b}$, and thus, $a^T x \geq k$ defines a facet of $\text{ECON}(G; k)$.

If (a) does not hold then there is an edge $e \in E \setminus \delta(W)$ such that, for no subset C of $\delta(W)$ with $|C| = k$, the set $C \cup (E \setminus (\delta(W) \cup \{e\}))$ is k -edge connected. Hence either no k -edge connected edge set satisfies $x(\delta(W)) \geq k$ with equality (and thus this inequality does not define a facet) or every k -edge connected subset $D \subseteq E$ with $|D \cap \delta(W)| = k$ contains e . Therefore $\{x \in \text{ECON}(G; k) \mid x(\delta(W)) = k\} \subseteq \{x \in \text{ECON}(G; k) \mid x_e = 1\}$ and $x(\delta(W)) \geq k$ does not define a facet.

If (b) does not hold, then the set of vertices contained in the face

$$F = \{x \in \text{ECON}(G; k) \mid x(\delta(W)) = k\}$$

is not linearly independent. Hence, as $\text{ECON}(G; k)$ is full-dimensional, F is not a facet of $\text{ECON}(G; k)$. $\square$

Theorem 6.2 is merely of technical interest and does not provide insight into the graph theoretical properties that make an inequality $x(\delta(W)) \geq k$ a facet-defining one. In fact, in contrast to our earlier belief, there seems no easy way to use connectivity properties only to decide whether or not such an inequality defines a facet of $\text{ECON}(G; k)$. This will become clear by the following observations.

PROPOSITION 6.3. *Let $G = (V, E)$ be $(k+1)$ -edge connected, $k \geq 2$, and $W \subseteq V$ with $\emptyset \neq W \neq V$ such that $G[W]$ and $G[V \setminus W]$ are k -edge connected. Then $x(\delta(W)) \geq k$ defines a facet of $\text{ECON}(G; k)$.*

Proof. We first show that (a) of Theorem 6.2 is satisfied. Let $e = uv \in E \setminus \delta(W)$, say $e \in E(W)$. If $G' = G[W] - e$ is k -edge connected then $C \cup (E \setminus (\delta(W) \cup \{e\}))$ is k -edge connected for every set $C \subseteq \delta(W)$ with $|C| = k$. So let us assume that $\lambda(G') = k - 1$. This implies that G' contains a cut of cardinality $k - 1$ that separates u and v . Among all node sets $W_u, W_v \subseteq W$ with $W_u \cap W_v = \emptyset$ such that $u \in W_u, v \in W_v, e \in \delta(W_u), e \in \delta(W_v), |\delta_{G'}(W_u)| = |\delta_{G'}(W_v)| = k - 1$ (note that such sets exist) we choose W_u and W_v such that W_u and W_v have cardinality as small as possible. Since G is $(k+1)$ -edge connected there exists an edge $f \in E$ with one endnode in W_u and the other in $V \setminus W$, and one edge $g \in E$ with one endnode in W_v and the other in $V \setminus W$. Let C be any subset of $\delta(W)$ with $|C| = k$ and $f, g \in C$. (Here $k \geq 2$ is needed.) We claim that $D = C \cup (E \setminus (\delta(W) \cup \{e\}))$ is k -edge connected. Suppose not; then there must be a cut $\delta(Z)$ in $G'' = (V, D)$ with $k - 1$ or fewer edges. It follows from our assumptions that $\delta_{G''}(Z)$ does not separate any two nodes of $V \setminus W$, that it must separate u and v (so we may assume that $u \in Z$) and that $|\delta_{G''}(Z)| = k - 1$. Since the cut cardinality function $|\delta_{G''}|$ is submodular on the subsets of nodes of G'' we obtain $2k - 2 \geq |\delta_{G''}(W_u)| + |\delta_{G''}(Z)| \geq |\delta_{G''}(W_u \cap Z)| + |\delta_{G''}(W \cup Z)|$. As $|\delta_{G''}(X)| \geq k - 1$ for all cuts in G'' we can conclude that $|\delta_{G''}(W_u \cap Z)| = k - 1$, and thus by the choice of W_u , we have $W_u = W_u \cap Z$. But then $f \in \delta_{G''}(Z)$ and thus $|\delta_{G''}(Z)| \geq k$, a contradiction. This proves that (a) of (6.2) is satisfied.

For every $C \subseteq \delta(W)$ with $|C| = k$, $C \cup (E \setminus \delta(W))$ is k -edge connected and the matrix whose columns are the incidence vectors of all possible k -element subsets of $\delta(W)$ has obviously full row rank. Thus a matrix M as required by (b) of (6.2) exists. Hence the theorem follows from (6.2). $\square$

The next example shows that the connectivity requirements on $G[W]$ and $G[V \setminus W]$ cannot be weakened. They are, in a sense, best possible.

Example 6.4. Let $k \geq 1$ and $G_1 = (V_1, E_1)$ be a minimal $(k - 1)$ -edge connected graph (i.e., G_1 is $(k - 1)$ -edge connected and each edge is $(k - 1)$ -edge essential) with at least $k + 1$ nodes of degree $k - 1$. (Such graphs exist for all large enough orders.) Let G be the graph obtained from the disjoint union of G_1 and the complete graph $K_{k+1} = (V_2, E_2)$ of order $k + 1$ by adding all edges that link a node in G_1 to a node in K_{k+1} . G is clearly $(k + 1)$ -edge connected. The inequality $x(\delta(V_1)) \geq k$ does not define a facet of $\text{ECON}(G; k)$, since there is no set $C \subseteq \delta(V_1)$ with $|C| = k$ such that $C \cup E_1 \cup E_2$ is k -edge connected (because at least one node in $(V_1 \cup V_2, C \cup E_1 \cup E_2)$ has degree $k - 1$).

On the other hand a cut inequality may define a facet of $\text{ECON}(G; k)$ even if $G[W]$ and $G[V \setminus W]$ are not highly connected as the next example shows.

Example 6.5. Consider four complete graphs on node sets A, B, C, D with $k \geq 4$ nodes. Link all nodes in $A \cup B$ to all nodes in $C \cup D$, add a set of $\lceil k/2 \rceil + 1$ node disjoint

edges with one endnode in A and one in B , and add a set of $\lceil k/2 \rceil + 1$ node disjoint edges with one endnode in C and the other in D . Let $G = (V, E)$ be the graph obtained this way. G is $(2k + \lceil k/2 \rceil + 1)$ -edge connected. Let $W \equiv A \cup B$. Then the edge (and node) connectivity of $G[W]$ and $G[V \setminus W]$ is $\lceil k/2 \rceil + 1$. Using Theorem (6.2) it is easy to show that $x(\delta(W)) \geq k$ defines a facet of $\text{ECON}(G; k)$.

The above example is an extreme case. The next observation shows that the connectivity of $G[W]$ and $G[V \setminus W]$ cannot be smaller than $\lceil k/2 \rceil + 1$ if the associated inequality defines a facet.

PROPOSITION 6.6. *Let $k \geq 2$, $G = (V, E)$ be a $(k + 1)$ -edge connected graph and $W \subseteq V$, $\emptyset \neq W \neq V$, such that the edge connectivity of $G[W]$ or $G[V \setminus W]$ is not larger than $\lceil k/2 \rceil$. Then $x(\delta(W))$ does not define a facet of $\text{ECON}(G; k)$.*

Proof. We may assume without loss of generality that $G[W]$ is $\lceil k/2 \rceil$ -edge connected but not $(\lceil k/2 \rceil + 1)$ -edge connected. By Menger's theorem $G[W]$ contains an edge e such that $G[W] - e$ is not $\lceil k/2 \rceil$ -edge connected. It is easy to see that each k -edge connected edge set $F \subseteq E$ with $\chi^F(\delta(W)) = k$ contains edge e . Hence by Theorem 6.2, $x(\delta(W)) \geq k$ does not define a facet of $\text{ECON}(G; k)$. $\square$

Propositions 6.3 and 6.6 yield the following result for the cases $k = 2, 3$ which are of particular practical interest.

COROLLARY 6.7. *Let $k \in \{2, 3\}$ and let $G = (V, E)$ be a $(k + 1)$ -edge connected graph. Let $W \subseteq V$, $\emptyset \neq W \neq V$. Then $x(\delta(W)) \geq k$ defines a facet of $\text{ECON}(G; k)$ if and only if $G[W]$ and $G[V \setminus W]$ are k -edge connected.*

Recall that Theorem 4.10 (or Corollary 5.7) yields that, for a 2-edge connected graph G and for a node set $W \subseteq V$, $\emptyset \neq W \neq V$, $x(\delta(W)) \geq 1$ defines a facet of $\text{ECON}(G; 1)$ if and only if $G[W]$ and $G[V \setminus W]$ are 2-edge connected.

Given $G = (V, E)$ and $y \in \mathbf{R}^E$, $0 \leq y \leq \mathbf{1}$, the separation problem for y and (6.1) (i) can be solved in polynomial time by computing a cut $\delta(W^*)$, $\emptyset \neq W^* \neq V$, of minimum capacity $y(\delta(W^*))$ —for instance by the Gomory–Hu method. If $y(\delta(W^*)) < k$ the vector y violates $x(\delta(W^*)) \geq k$, otherwise all inequalities (6.1) (i) are satisfied by y . Hence the LP-relaxation of the minimum cost k -edge connected subgraph problem following from (6.1) (by dropping the integrality constraints (iii)) can be solved in polynomial time.

7. Node connectivity. Parallel edges do not play a role in node connectivity questions. Thus we assume throughout this section that all graphs $G = (V, E)$ considered are simple. Setting in (2.1) $d_{st} = 1$, $r_{st} = k$, and $k_{st} = k - 1$ ($k \in \mathbf{Z}$, $1 \leq k \leq |V| - 1$) for all $s, t \in V$, we obtain the following integer linear program:

$$\begin{aligned}
 (7.1) \quad & \min c^T x \\
 & \text{(i) } x(\delta(W)) \geq k \quad \text{for all } W \subseteq V; \\
 & \text{(ii) } x(\delta_{G-Z}(W)) \geq 1 \quad \text{for all node sets } Z \subseteq V \text{ with} \\
 & \quad |Z| = k - 1 \text{ and all node sets} \\
 & \quad W \subseteq V \setminus Z, \emptyset \neq W \neq V \setminus Z; \\
 & \text{(iii) } 0 \leq x_e \leq 1 \quad \text{for all } e \in E; \\
 & \text{(iv) } x_e \in \{0, 1\} \quad \text{for all } e \in E.
 \end{aligned}$$

Clearly, every optimum solution of (7.1) is a minimum-cost spanning k -node connected subgraph of G . The polytope

$$\text{NCON}(G; k) \equiv \text{conv} \{ \chi^C \in \mathbf{R}^E \mid (V, C) \text{ is } k\text{-node connected} \}$$

is called the *polytope of k -node connected* (or just *k -connected*) *subgraphs* of G . For $k = 1$ this polytope coincides with $\text{ECON}(G; 1)$ and $\text{CON}(G)$. Note that $\text{NCON}(G; k) \subseteq$

ECON ($G; k$). Using the observations of § 4 we can immediately strengthen (7.1) to the following ILP:

$$\begin{aligned}
 (7.2) \quad & \min c^T x \\
 & (i) \quad x(\delta(W)) \geq k \quad \text{for all } W \subseteq V, \emptyset \neq W \neq V; \\
 & (ii) \quad \frac{1}{2} \sum_{i=1}^p x(\delta_{G-Z}(V_i)) \geq p-1 \quad \text{for all } Z \subseteq V \text{ with } |Z| = k-1 \\
 & \quad \text{and all nontrivial partitions } V_1, \dots, V_p \text{ of } V \setminus Z, p \geq 2; \\
 & (iii) \quad 0 \leq x_e \leq 1 \quad \text{for all } e \in E; \\
 & (iv) \quad x_e \in \{0, 1\} \quad \text{for all } e \in E.
 \end{aligned}$$

It follows from our remarks in § 6 that the separation problem for (7.2) (i) and (iii) can be solved in polynomial time. Our remarks in § 4 imply that the separation problem for (7.2) (ii) and (iii) can be solved in polynomial time for every fixed node set Z . Thus, for k fixed, the separation problem for (ii) and hence the LP-relaxation of (7.2) are solvable in polynomial time.

We will now investigate NCON ($G; k$) and find out which of the inequalities of (7.2) induce facets of the polytope of k -node connected subgraphs.

We will call an edge e of a graph $G = (V, E)$ *k-node essential* if $G - e$ is not k -node connected. The k -node-essential edges of G are thus precisely the $(G; k\mathbf{1}, (k-1)\mathbf{1}, \mathbf{1})$ -essential edges of G . By Theorem 3.2, NCON ($G; k$) has dimension $|E|$ if and only if G is k -node connected and contains no k -node-essential edge. In particular, NCON ($G; k$) has dimension $|E|$ if G is $(k+1)$ -node connected.

The results of § 6 concerning the cut inequalities $x(\delta(W)) \geq k$ for ECON ($G; k$) carry over to the case of node connectivity. Only minor modifications in the proofs and examples have to be made. We thus only summarize these observations and give no proofs. The main proof technique is an appropriate modification of Theorem 6.2.

THEOREM 7.3. (a) Let $G = (V, E)$ be a k -node connected graph, $k \geq 2$, without k -node-essential edge and let $W \subseteq V$, $\emptyset \neq W \neq V$, such that $G[W]$ and $G[V \setminus W]$ are k -node connected. Then $x(\delta(W)) \geq k$ defines a facet of NCON ($G; k$), cf. (6.3).

(b) If $x(\delta(W)) \geq k$ defines a facet of NCON ($G; k$), where $G = (V, E)$ is a k -node connected graph, $k \geq 2$, without k -node-essential edge and $W \subseteq V$, $\emptyset \neq W \neq V$, then, for each $e \in E(W)$ and for each $f \in E(V \setminus W)$, $\kappa(G([W] - \{e\}))$ and $\kappa(G([V \setminus W] - \{f\}))$ are at least $\lceil k/2 \rceil$, cf. (6.6).

(c) For every $k \geq 1$, there are $(k+1)$ -node connected graphs $G = (V, E)$ and sets $W \subseteq V$ such that $G[W]$ is $(k-1)$ -node connected, $G[V \setminus W]$ is $(k+1)$ -node connected and such that $x(\delta(W)) \geq k$ does not define a facet of NCON ($G; k$), cf. (6.4).

(d) For every $k \geq 4$, there are $(k+1)$ -node connected graphs $G = (V, E)$ and sets $W \subseteq V$ such that $\kappa(G[W])$ and $\kappa(G[V \setminus W])$ are equal to $\lceil k/2 \rceil + 1$ and such that $x(\delta(W)) \geq k$ defines a facet of NCON ($G; k$), cf. (6.5).

We will now give a technical characterization of those inequalities (7.2) (ii) that define facets of NCON ($G; k$).

THEOREM 7.4. Let $k \geq 1$ and let $G = (V, E)$ be a k -node connected graph without k -node-essential edge. Let $Z \subseteq V$ with $|Z| = k-1$ and let $V_1, \dots, V_p$, $p \geq 2$, be a partition of $V \setminus Z$. Let $E' = E(V_1) \cup \dots \cup E(V_p) \cup E(Z) \cup \delta(Z)$ and let $\bar{G} = (\bar{V}, \bar{E})$ be the graph obtained from G by deleting Z and contracting $V_1, \dots, V_p$. Then

$$\frac{1}{2} \sum_{i=1}^p x(\delta_{G-Z}(V_i)) = x(E \setminus E') = x(\bar{E}) \geq p-1$$

defines a facet of $\text{NCON}(G; k)$ if and only if

- (a) for each edge $e \in E'$ there exists a spanning tree T in $\hat{G}$ such that $T \cup E' \setminus \{e\}$ is k -node connected; and
- (b) there exists a set $T_1, \dots, T_s$ of spanning trees in $\hat{G}$, where $s = |\hat{E}|$, such that:
 - (b₁) $T_i \cup E'$ is k -node connected, and
 - (b₂) the $s \times s$ -matrix M whose columns are the incidence vectors $\chi^{T_i} \in \mathbf{R}^{\hat{E}}$ is nonsingular.

Proof. Let $Z \subseteq V$ with $|Z| = k - 1$ and a partition $V_1, \dots, V_p$ of $V \setminus Z$, $p \geq 2$, be given. Set $a^T x \equiv \frac{1}{2} \sum_{i=1}^p x(\delta_{G-Z}(V_i))$, $F_a \equiv \{x \in \text{NCON}(G; k) \mid a^T x = p - 1\}$, and let us call an edge set $C \subseteq E$ tight if (V, C) is k -node connected and $a^T \chi^C = p - 1$, i.e., if $\chi^C \in F_a$. Note that $E = E' \cup \hat{E}$, and since G is k -node connected and without k -node-essential edge, $\hat{G}$ is 2-edge connected.

Suppose (a) and (b) hold and that $b^T x \geq \beta$ defines a facet F_b of $\text{NCON}(G; k)$ containing F_a . Condition (a) implies that, for each edge $e \in E'$, there exists a spanning tree T in $\hat{G}$ such that $C_e \equiv T \cup E' \setminus \{e\}$ is k -node connected. Since $|C_e \cap \bigcup_{i=1}^p \delta_{G-Z}(V_i)| = p - 1$ we can conclude that C_e is tight, hence, since $F_a \subseteq F_b$, $b^T \chi^{C_e} = \beta$ holds. But, for $C'_e \equiv C_e \cup \{e\}$, we also have $a^T \chi^{C'_e} = p - 1$ and thus $b^T \chi^{C'_e} = \beta$. This implies $b^T \chi^{C_e} = b^T \chi^{C'_e}$ and hence $b_e = 0$ for each $e \in E'$.

$D_i \equiv T_i \cup E'$ is tight for $i = 1, \dots, s$ by (b). The equation $y^T M = \beta \mathbf{1}^T$ is solved by the vectors $(\beta/(p - 1))\mathbf{1}^T$ and $\bar{b}^T$ of $\mathbf{R}^{\hat{E}}$ (where $\bar{b}$ is the vector obtained from b by deleting the components E'). Since M is nonsingular, we have $\bar{b}^T \equiv (\beta/(p - 1))\mathbf{1}^T$. Clearly $\beta/(p - 1) \neq 0$, and thus we can conclude that $a = ((p - 1)/\beta)b$ which implies that F_a defines a facet of $\text{NCON}(G; k)$.

If (a) does not hold then there is either no tight set at all (and hence $a^T x \geq p - 1$ does not define a facet) or there is an edge $e \in E'$ such that each tight set $C \subseteq E$ contains e . This implies that $F_a \subseteq \{x \in \text{NCON}(G; k) \mid x_e = 1\}$ and hence F_a does not define a facet. If (b) does not hold then the set of vertices contained in the face F_a does not span a hyperplane of $\mathbf{R}^E$, and thus, as $\text{NCON}(G; k)$ has dimension $|E|$, F_a is not a facet of $\text{NCON}(G; k)$. $\square$

As in the case of Theorem 6.2 we see no way to translate conditions (a) and (b) of Theorem 7.4 equivalently into nice graph theoretical properties, in particular, into connectivity requirements. An easy consequence of Theorem 7.4 is the following.

THEOREM 7.5. *Let $G = (V, E)$ be a k -node connected graph, $k \geq 1$, without k -node-essential edge. Let $Z \subseteq V$ with $|Z| = k - 1$, and let $V_1, \dots, V_p$, $p \geq 2$, be a partition of $V \setminus Z$. If $G[Z \cup V_i] - e$ is k -node connected for every edge e in $G[V_i \cup Z]$ and for every $i \in \{1, \dots, p\}$ and if $\hat{G} = (\hat{V}, \hat{E})$ is 2-node connected then $\frac{1}{2} \sum_{i=1}^p x(\delta_{G-Z}(V_i)) \geq p - 1$ defines a facet of $\text{NCON}(G; k)$.*

Proof. To prove (b) of Theorem 7.4 we first show that for every spanning tree T of $\hat{G} = (G - Z)/V_1/\dots/V_p$ the edge set $C \equiv T \cup E'$, where $E' \equiv E(V_1) \cup \dots \cup E(V_p) \cup E(Z) \cup \delta(Z)$ is k -node connected. By assumption, for every $i \in \{1, \dots, p\}$ and every two nodes $u, v \in V_i \cup Z$, the graph $G[V_i \cup Z]$ and hence its supergraph (V, C) contains k node disjoint $[u, v]$ -paths. Let $u \in V_i$, $v \in V_j$, $i \neq j$. The edge set $C \setminus (E(Z) \cup \delta(Z))$ contains a $[u, v]$ -path P by construction. Let t_i be the last node of V_i and t_j the first node of V_j that is encountered by going from u to v along P . Since $G[V_i \cup Z]$ is k -connected it contains a $(u, Z \cup \{t_i\})$ -fan and similarly $G[V_j \cup Z]$ contains a $(v, Z \cup \{t_j\})$ -fan. This implies that C contains k node disjoint $[u, v]$ -paths, and thus C is k -node connected.

Let $\bar{M}$ be the matrix whose columns are the incidence vectors of all spanning trees of $\hat{G}$. Using the arguments of the end of the proof of Theorem 5.6, we can easily show that the 2-node connectedness of $\hat{G}$ implies that $\bar{M}$ has full row rank. Hence $\bar{M}$ contains a nonsingular $s \times s$ -submatrix M .

To prove (a) of Theorem 7.4, let $e \in E'$ be an arbitrary edge, say $e \in E(V_1 \cup Z)$. Since e is not k -node-essential with respect to $G[V_1 \cup Z]$, $G[V_1 \cup Z] - e$ is still k -node connected. By assumption and from the arguments used above, it follows that for every spanning tree T of $\hat{G}$, $T \cup E' \setminus \{e\}$ is k -node connected. $\square$

The conditions of Theorem 7.5 are, in particular, satisfied if G is $(k + 1)$ -node connected and if $G[V_i \cup Z]$ is $(k + 1)$ -connected for all $i \in \{1, \dots, p\}$.

Observe that, for $k = 1$, Theorem 7.5 implies that, for any 2-edge connected graph $G = (V, E)$, any partition $V_1, \dots, V_p$ of V , $p \geq 2$, such that $G[V_i]$ is 2-edge connected induces a facet defining inequality $\frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p - 1$ of $\text{NCON}(G; 1) = \text{CON}(G)$. By Theorem 4.10 these are exactly the facet-defining inequalities of $\text{CON}(G)$ of this type. It is easy to see that " $G[V_i]$ is 2-edge connected for $i = 1, \dots, p$ " is a necessary condition for an inequality of type (7.2) (ii) to define a facet of $\text{NCON}(G; k)$. However, this condition is far from being sufficient for $k \geq 2$.

In fact, the connectivity conditions on $G[V_i \cup Z]$ in (7.5) cannot be weakened further. For instance, we can show that for every $k \geq 1$ there exists a $(k + 1)$ -node connected graph $G = (V, E)$ with node sets $Z, V_1, \dots, V_p \subseteq V$ such that $|Z| = k - 1$; $V_1, V_2, \dots, V_p$ is a partition of V and such that $G[Z]$ is a complete graph, $G[Z \cup V_i]$ and $G[V_i]$ are k -node connected for $i = 1, \dots, p$ and the corresponding inequality $\frac{1}{2} \sum_{i=1}^p x(\delta_{G-Z}(V_i)) \geq p - 1$ does not define a facet of $\text{NCON}(G; k)$. We describe this construction for the case where k is even in the following example.

Example 7.6. Let $k = 2r$, $r \geq 1$. We construct a graph $G = (V, E)$ with $5k - 1$ nodes as follows. Let A_i , $i = 1, \dots, 8$ and Z be node sets with the following properties $V = Z \cup \bigcup_{i=1}^8 A_i$, $|Z| = k - 1$ and Z induces a complete subgraph of G . All A_i are stable sets of G and satisfy $|A_i| = r$. Every node of Z is linked to every node of $A_1 \cup A_2 \cup A_5 \cup A_6$ by an edge.

Every node of A_i is linked to every node of A_j ($i < j$) by an edge for $j = i + 1$ ($i = 1, 2, 3, 5, 6, 7$), $i = 1$ and $j = 4$, $i = 5$ and $j = 8$, $i = 3$ and $j = 7$, $i = 4$ and $j = 8$. The scheme of the graph G obtained this way is displayed in Fig. 1.

It is easy to see that for $V_1 = \bigcup_{i=1}^4 A_i$, $V_2 = \bigcup_{i=5}^8 A_i$ the graphs $G[V_1]$, $G[V_2]$, $G[Z \cup V_1]$, $G[Z \cup V_2]$ are k -node connected and that G is $(k + r)$ -node connected. But, for every edge e with one endnode in A_7 and the other in A_8 , or one endnode in A_3 and

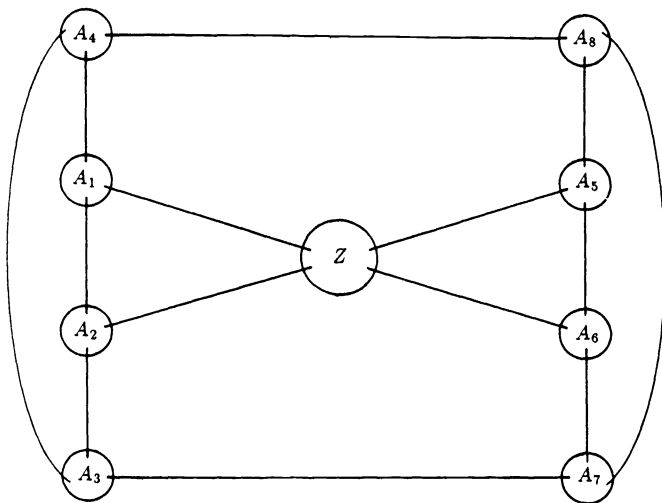


FIG. 1

the other in A_4 , there is no spanning tree T in $\hat{G} = (G - Z)/V_1/V_2$ (i.e., T is just an edge) such that $T \cup E(V_1) \cup E(V_2) \cup E(Z) \cup \delta(Z) \setminus \{e\}$ is k -node connected. So by Theorem 7.4 the corresponding inequality does not define a facet of $\text{NCON}(G; k)$.

We will now give, for every $k \geq 2$, an example where no two nodes in $G[Z]$ are adjacent and where $\kappa(G[Z \cup V_i]) = 1$, yet the corresponding inequality defines a facet of $\text{NCON}(G; k)$.

Example 7.7. Let $k \geq 2$. We construct a graph $G = (V, E)$ with $p + k - 1$ nodes, where $p \geq k + 1$ as follows. Let $Z \subseteq V$ with $|Z| = k - 1$ be a node set such that no two nodes in Z are adjacent. Let $V' \equiv \{v_1, \dots, v_p\}$ be further $p \geq k + 1$ nodes and set $V_i \equiv \{v_i\}$, $i = 1, \dots, p$. The subgraph of G induced by V' is complete and every node of V' is linked to every node of Z by an edge. Thus $G[Z \cup V_i]$ is a star and $\kappa(G[Z \cup V_i]) = 1$. It is easy to see that G is $(k + 1)$ -connected and that properties (a) and (b) of Theorem 7.4 are satisfied.

The results of this section show that dropping the integrality stipulations (iv) from (7.2) yields a reasonable (and, for small fixed k , computationally tractable) LP-relaxation for the problem of finding a spanning k -connected subgraph of minimum cost.

8. Node and edge connectivity mixed: A model used in practice. We will now introduce a model that describes the current situation in the area of building fiber optical networks faithfully—see Monma and Shallcross (1989) for a detailed overview of the approaches to design “survivable” communication networks. This model contains a mixture of certain node and edge connectivity requirements. As far as we can see, graphs with these kinds of “mixed connectivity” properties have not received much attention in the graph theory literature nor have the related optimization problems been considered seriously in combinatorial optimization.

As before we begin with the graph $G = (V, E)$ that describes the possible direct connections between the given locations of switches. (Recall that two nodes $s, t \in V$ are called (locally) k -connected or (locally) k -edge connected if G contains k $[s, t]$ -paths that are node-disjoint or edge-disjoint, respectively.) In the network design area it is common to classify the locations by “type.” So, for each node $s \in V$ we introduce two connectivity parameters, denoted by r_s and k_s ; r_s is called the *edge connectivity type*, k_s the *node connectivity type*. As usual we have a cost c_e for all $e \in E$. We are looking for a subset C of the edge set E of minimum cost $c(C)$ such that for every two nodes $s, t \in V$, $s \neq t$, (V, C) contains $\min\{r_s, r_t\}$ edge-disjoint $[s, t]$ -paths and $\min\{k_s, k_t\}$ node-disjoint $[s, t]$ -paths.

The conditions stated above require that the “cable network” (V, C) is locally $\min\{r_s, r_t\}$ -edge connected and locally $\min\{k_s, k_t\}$ -connected. Since local k -connectedness implies local k -edge connectedness we may assume that

$$(8.1) \quad r_s \geq k_s \quad \text{for all } s \in V.$$

Setting, for all $s, t \in V$, $s \neq t$, $r_{st} \equiv \min\{r_s, r_t\}$, $d_{st} \equiv 1$, and $k_{st} \equiv \min\{k_s, k_t\} - 1$, we see that the problem defined above can be viewed as a special case of (2.1).

In telephone network applications of the type considered here, we typically have

$$(8.2) \quad r_s, k_s \in \{0, 1, 2\} \quad \text{for all } s \in V.$$

(But we also know of a communication network application with $k_s \in \{0, 1, \dots, 5\}$.) The nodes $s \in V$ with $r_s = k_s = 0$ are the Steiner nodes. They are not required to be in the fiber optical network but they may be used to construct the network. The nodes (respectively, offices or locations) s with $k_s = 2$ are sometimes called “special offices.” They frequently carry high loads of communication traffic. Their failure—without the

possibility of rerouting—would be fatal to the system and result in considerable losses (financially and in customer good-will).

Using (2.1) and (7.2) we can formulate the mixed connectivity problem described above as an integer linear program as follows:

$$\begin{aligned}
 (8.3) \quad & \min c^T x \\
 & (i) \quad x(\delta(W)) \geq \min \{r_s, r_t\} \quad \text{for all } s, t \in V, s \neq t, \text{ and} \\
 & \quad \text{for all } W \subseteq V, s \in W, t \notin W; \\
 & (ii) \quad \frac{1}{2} \sum_{i=1}^p x(\delta(V_i)) \geq p - 1 \quad \text{for all } s, t \in V, s \neq t \text{ and for all} \\
 & \quad Z \subseteq V \setminus \{s, t\}, |Z| = \min \{k_s, k_t\} - 1, \\
 & \quad \text{and all nontrivial partitions} \\
 & \quad V_1, \dots, V_p \text{ of } V \setminus Z, p \geq 2; \\
 & (iii) \quad 0 \leq x_e \leq 1 \quad \text{for all } e \in E; \\
 & (iv) \quad x_e \in \{0, 1\} \quad \text{for all } e \in E.
 \end{aligned}$$

Needless to say, (8.3) belongs to the class of NP-hard optimization problems; in fact, this is true even in the case where k and r satisfy (8.1) and (8.2), since the Steiner tree problem is a special case.

As before, to address optimization issues it is natural to introduce a polytope associated with the integral solutions of (8.3). So, let $G = (V, E)$ be a graph and $k \in \mathbf{R}^V$, $r \in \mathbf{R}^V$ be two vectors of nonnegative integers. Then

$$\text{CON}_{r,k}(G) \equiv \text{conv} \{x \in \mathbf{R}^E \mid x \text{ satisfies (8.3)(i), } \dots, \text{(iv)}\}$$

is the convex hull of the incidence vectors x^C of edge sets $C \subseteq E$ such that for every two nodes $s, t \in V, s \neq t$, the subgraph (V, C) is locally $\min \{k_s, k_t\}$ -connected and locally $\min \{r_s, r_t\}$ -edge connected.

As mentioned before, the separation problems for the inequality systems of (8.3) can be solved in polynomial time, for r_s and k_s small, with not so small degrees of the polynomials of the running time functions, however. To get practically efficient cutting plane algorithms for the solution of (8.3) we have to use heuristic separation routines in addition.

The theoretical work presented here on (partial) characterizations of facets will be used to design fast heuristic separation algorithms. Our work on polyhedral properties of $\text{CON}_{r,k}(G)$, as well as on computational aspects of the LP-relaxation that follows from (8.3), is still in progress and will be reported in a forthcoming paper.

REFERENCES

- D. BIENSTOCK, E. F. BRICKELL, AND C. L. MONMA (1987), *Properties of k -connected networks*, Bell Communications Research, internal memorandum.
- R. G. BLAND, D. GOLDFARB, AND M. J. TODD (1981), *The ellipsoid method: A survey*, Oper. Res., 29, pp. 1039–1097.
- B. BOLLOBÁS (1978), *Extremal Graph Theory*, Academic Press, London.
- N. CHRISTOFIDES AND C. A. WHITLOCK (1981), *Network synthesis with connectivity constraints: A survey*, in Operational Research '81, J. P. Brans, ed., pp. 705–723.
- G. CORNUÉJOLS, J. FONLUPT, AND D. NADDEF (1985), *The traveling salesman problem on a graph and some related integer polyhedra*, Math. Programming, 33, pp. 1–27.
- W. H. CUNNINGHAM (1984), *Testing membership in matroid polyhedra*, J. Combin. Theory B, 36, pp. 161–188.

- J. EDMONDS (1970), *Submodular functions, matroids, and certain polyhedra*, in *Combinatorial Structures and Their Applications*, R. Guy et al., eds., Gordon and Breach, New York, pp. 69–87.
- (1971), *Matroids and the greedy algorithm*, *Math. Programming*, 1, pp. 127–136.
- R. E. GOMORY AND T. C. HU (1961), *Multi-terminal network flows*, *J. Soc. Indust. Appl. Math.*, 9, pp. 551–570.
- R. L. GRAHAM AND P. HELL (1985), *On the history of the minimum spanning tree problem*, *Ann. Hist. Comput.*, 7, pp. 41–57.
- M. GRÖTSCHEL, L. LOVÁSZ, AND A. SCHRIJVER (1981), *The ellipsoid method and its consequences in combinatorial optimization*, *Combinatorica*, 1, pp. 169–197.
- (1988), *Geometric Algorithms and Combinatorial Optimization*, Springer-Verlag, Heidelberg.
- T. C. HU (1969), *Integer Programming and Network Flows*, Addison-Wesley, Reading, MA.
- F. LING AND T. KAMEDA (1987), *Complexity of graph connectivity functions*, Working Paper, School of Computing Science, Simon Fraser University, Burnaby, British Columbia, Canada.
- C. L. MONMA, B. S. MUNSON, AND W. R. PULLEYBLANK (1985), *Minimum-weight two-connected spanning networks*, Bell Communications Research, internal memorandum.
- C. L. MONMA AND D. F. SHALLCROSS (1989), *Methods for designing survivable communication networks*, *Oper. Res.*, 38, pp. 531–541.
- A. PRODON (1985), *A polyhedron for Steiner trees in series parallel graphs*, Technical Report, Department of Mathematics, EPF Lausanne.
- A. PRODON, T. M. LIEBLING, AND H. GRÖFLIN (1985), *Steiner's problem on two-trees*, Technical Report RO 850314, Department of Mathematics, EPF Lausanne.
- K. WHITE, M. FARBER, AND W. R. PULLEYBLANK (1985), *Steiner trees, connected domination and strongly chordal graphs*, *Networks*, 15, pp. 109–124.

VERTEX TRANSITIVITY AND SUPER LINE CONNECTEDNESS*

YAHYA OULD HAMIDOUNE† AND RALPH TINDELL‡

Abstract. A connected graph X is said to be super line connected if every minimum edge cut is the set of edges incident with some vertex. It is proved that a connected Abelian Cayley graph is super line connected unless it is a cycle or the product of complete graphs, one of size 2 and one of size $m \geq 3$. This result generalizes previous characterizations for circulants by Boesch and Wang and for hypercubes by Boesch, Lee, Wang, and Yang. A simple algebraic characterization for strongly connected Cayley digraphs that are not super arc connected and a structural characterization for the Abelian case are given. In a previous work, one of the authors presented a characterization of the connected vertex-transitive graphs that are not super line connected. The final result of the present paper generalizes this result to directed graphs.

Key words. connectivity, Cayley graphs, vertex-transitive graphs, super line connectivity

AMS(MOS) subject classifications. 05C40, 05C25, 05C20, 05C75

1. Introduction. All graphs and digraphs are assumed to be finite and without loops or multiple arcs. We shall identify an undirected graph X with its symmetric digraph X^* , obtained by replacing each edge e of X with two opposite arcs with the same endpoints as e . Proper subset inclusion is denoted $\subset$ and general subset inclusion by $\subseteq$. By abuse of notation, we identify the elements of $\mathbf{Z}_m$ with the natural numbers between 0 and $m - 1$. The *Cayley digraph* $\text{Cay}(G, S)$ determined by a group G has vertex set G and arc set all pairs of the form (a, as) with $a \in G$ and $s \in S$. The arcs of $\text{Cay}(G, S)$ are viewed as being *colored* by the elements of S , and an arc of the form (a, as) is called an *s-arc* of the Cayley digraph. A Cayley digraph $\text{Cay}(G, S)$ is strongly connected if and only if it is connected, which is true if and only if S generates G . It is easy to see that $\text{Cay}(G, S)$ is a symmetric digraph (i.e., an undirected graph) if and only if $S = S^{-1}$. It is also easy to see that $\text{Cay}(G, S)$ is a symmetric cycle if and only if S is a generating set of the form $\{a, a^{-1}\}$.

Let $X = (V, E)$ be a connected graph and W a proper subset of V . The set of edges joining W to $V \setminus W$ will be denoted by $\omega(W)$. The *line-connectivity* of X is then defined by $\lambda(X) = \min \{|\omega(W)| : \emptyset \subset W \subset V\}$. If $|\omega(W)| = \lambda(X)$, then the set $\omega(W)$ is called a *minimum edge cut* of X . In connection with studies of network reliability, Bauer, Boesch, Suffel, and Tindell [1] defined a connected graph $X = (V, E)$ to be *super line connected* if every minimum edge cut of X is the set of edges incident with some vertex. In the same paper, they proved that a circulant, which is to say a Cayley graph of the form $\text{Cay}(\mathbf{Z}_n, S)$, is super line connected if S contains 1. Boesch and Wang [3] later showed that the only circulants that are not super line connected are the cycles and the graphs isomorphic to $\text{Cay}(\mathbf{Z}_{2m}, \{2, 4, \dots, 2m - 2, m\})$ for some odd integer m . It should be noted that the latter graphs are isomorphic to the Cartesian product of K_m and K_2 , which consists of two complete graphs joined by a perfect matching. Recently Yang et al. [8] proved that the hypercube $\text{Cay}(\mathbf{Z}_r^n, \{(1, 0, \dots, 0), \dots, (0, 0, \dots, 1)\})$ is super line connected unless $n = r = 2$. Generalizing both results, we show that an Abelian Cayley graph is super line connected unless it is a cycle or isomorphic to $K_m \times K_2$ for

* Received by the editors November 8, 1989; accepted for publication (in revised form) February 7, 1990.

† Université Pierre et Marie Curie, U.E.R. Combinatoire, 4 Place Jussieu, 75230 Paris, Cedex, France.

‡ Department of Electrical Engineering and Computer Science, Stevens Institute of Technology, Hoboken, New Jersey 07030. This research was partially supported by the Laboratoire de Recherche en Informatique, Université de Paris Sud, Orsay, France.

some integer $m \geq 3$. We note that the latter exceptional graphs are circulants only if m is odd (see below).

Recall that a digraph X is *vertex-transitive* if for every pair u, v of vertices of X there is an automorphism φ of X with $\varphi(u) = v$. Arc-transitivity is defined similarly with arcs in place of vertices. We note that since left multiplication by a fixed member of a group G yields an automorphism of the Cayley digraph $\text{Cay}(G, S)$, all such digraphs are vertex transitive. We say that an undirected graph X is *1-transitive* if its symmetric digraph is arc-transitive. Mader [6] proved that a connected vertex-transitive graph has line-connectivity equal to its degree. However, there are vertex-transitive graphs that are not super line connected. A characterization of such graphs has been given by Tindell ([7], referenced in [2]).

THEOREM 1 (Tindell). *If X is a connected vertex-transitive graph of degree $\delta \geq 3$ that is not super line connected, then*

- 1) *X is the union of pairwise vertex disjoint complete subgraphs of size δ , called super atoms of X , together with a collection of independent edges;*
- 2) *there is an integer m such that the number of independent edges joining any two super atoms of X is either 0 or m ; and*
- 3) *the graph Y obtained by contracting each super atom to a vertex and eliminating multiple edges and self-loops is a 1-transitive graph.*

Moreover, every connected 1-transitive graph Y and integer $m \geq 1$, with $m \geq 2$ if Y is a cycle, determines in this fashion a nonsuper line connected vertex-transitive graph.

We note here an interesting consequence of Theorem 1 that was not observed in the original work. In the terminology of Theorem 1, each vertex of the graph X is incident with exactly one of the independent edges. It then follows that the number of vertices of X is exactly twice the number of independent edges, and hence that X has even order. This observation is summarized in the following. Recall that C_n denotes the n -cycle.

THEOREM 2. *If $X \neq C_n$ is a connected vertex-transitive graph of odd order, then X is super line connected.*

In this paper, we generalize Theorem 1 to a characterization of strongly connected vertex-transitive digraphs, which are not super arc connected (defined below). In the special case of Cayley digraphs, we show that if $\text{Cay}(G, S) \neq C_n$ is strongly connected then $\text{Cay}(G, S)$ is not super arc connected if and only if $S = (H \setminus \{1\}) \cup \{t\}$ for some nontrivial subgroup H of G and element t of $G \setminus H$. When H is a normal subgroup of G , an explicit construction can be given for the Cayley digraph X , which obviously covers the case of Abelian Cayley digraphs.

2. Preliminaries. Let $X = (V, E)$ be a digraph. The outdegree and the indegree of vertex u in X will be denoted $d_X^+(u)$ and $d_X^-(u)$, respectively. If all vertices have the same outdegree, then this number is called the outdegree of X and denoted $d^+(X)$. If A is a subset of V and $u \in A$, we will, in an abuse of notation, denote by $d_A^+(u)$ the outdegree of u in the subdigraph of X induced by A . If W is a proper nonempty subgraph of V , then the set of arcs joining W to $V \setminus W$ (in the directed sense) will be denoted by $\omega^+(W)$. The set of arcs joining $V \setminus W$ to W , which is to say $\omega^+(V \setminus W)$, will be denoted $\omega^-(W)$. More precisely,

$$\omega^+(W) = \{(u, v) : u \in W \text{ and } v \in V \setminus W\}$$

$$\omega^-(W) = \{(u, v) : u \in V \setminus W \text{ and } v \in W\}.$$

The *arc-connectivity* $\lambda(X)$ of X is then defined to be $\min \{|\omega^+(W)| : \emptyset \subset W \subset V\}$ if X is strongly connected and 0 otherwise.

Let $X = (V, E)$ be a strongly connected digraph and let W be a proper, nonempty subset W of V . If $|\omega^+(W)| = \lambda(X)$, then W is called a *positive fragment*, $V \setminus W$ is called a *negative fragment*, and $\omega^+(W)$ is called a *minimum arc cut* of X . We note here a simple property of fragments.

LEMMA 1. *Let $X = (V, E)$ be a strongly connected digraph and let A and B be positive (negative) fragments of X . If $A \cup B \neq V$ and $A \cap B \neq \emptyset$, then $A \cap B$ and $A \cup B$ are positive (negative) fragments of X .*

Lemma 1 is an immediate consequence of the inequality

$$|\omega^+(A \cap B)| + |\omega^-(V \setminus (A \cup B))| \leq |\omega^+(A)| + |\omega^+(B)|,$$

which may easily be verified by the reader.

A strongly connected digraph X is said to be *super arc connected* if every minimum arc cut is either the set of outarcs or the set of inarcs of some vertex. We note that a variant of this extension of the super arc connected concept to digraphs was introduced in [4]. It is clear that the digraph definition coincides with the definition for undirected graphs if we identify such graphs with their symmetric digraphs.

We now turn to a digraph version of a definition that was crucial in the characterization [7] of nonsuper line connected vertex-transitive graphs. To avoid technicalities, we assume that X is vertex-transitive. One consequence of this assumption is that a subset W of the vertex set V of X is a positive fragment of X if and only if W is a negative fragment of X . Thus henceforth, we will drop the prefixes “positive” and “negative” in referring to fragments. A fragment with cardinality at least 2 will be referred to as a *nontrivial* fragment. A *super atom* of X is a nontrivial fragment with the smallest possible cardinality. It is obvious from definitions that a strongly connected, vertex-transitive digraph is super arc connected if and only if every super atom of X is the complement of a vertex. We note that Hamidoune [5] has shown that strongly connected vertex-transitive digraphs have arc connectivity equal to their outdegree.

LEMMA 2. *Let $X = (V, E)$ be a strongly connected, vertex-transitive digraph that is not super arc connected and is not isomorphic to a symmetric cycle. If A and B are super atoms of X with $A \cap B \neq \emptyset$, then $A = B$.*

Proof. First note that $d^+(X)$ cannot be 1 because then X would be a circuit, which is super arc connected. We prove the lemma by assuming that $A \neq B$ and deriving a contradiction. Since $|A| = |B| \leq |V|/2$ and $A \cap B \neq \emptyset$, $A \cup B \neq V$. Also, since neither A nor B is a subset of the other, $A \cup (V \setminus B) \neq V$ and $B \cup (V \setminus A) \neq V$. Thus by Lemma 1, the sets $A \cap B$, $A \setminus B$, and $B \setminus A$ are fragments. By the definition of super atoms, each of these sets must have cardinality 1. Therefore, $A = \{u, v\}$ and $B = \{u, w\}$ for some vertices u, v, w of X . Thus $d_A^+(v) = d_A^-(u) \leq 1$ and $d_A^+(u) = d_A^-(v) \leq 1$. We now have

$$\begin{aligned} |\omega^+(A)| &= d_X^+(v) - d_A^+(v) + d_X^+(u) - d_A^+(u) \\ &\geq 2d^+(X) - 2. \end{aligned}$$

Since $|\omega^+(A)| = \lambda(X) = d^+(X)$, we conclude that $d^+(X) = 2$ and that $d_A^+(v) = d_A^+(u) = 1$. Applying the preceding argument with B in place of A , we also obtain $d_B^+(v) = d_B^+(u) = 1$. Thus, the arcs incident with u are (u, v) , (v, u) , (u, w) and (w, u) , and it follows that the vertex-transitive digraph X is isomorphic to a symmetric cycle. This contradiction establishes the lemma.

The following corollary is an easy consequence of Lemma 1.

COROLLARY 1. *Let $X = (V, E)$ be a strongly connected, vertex-transitive digraph that is not super arc connected. Then each super atom induces a vertex-transitive subdigraph of X .*

3. Local characterization theorem. In this section we prove a “local” characterization theorem and apply this theorem to Cayley digraphs. A *clique* in a digraph X is a complete symmetric subdigraph of X .

LEMMA 3. *Let $X = (V, E)$ be a strongly connected, vertex-transitive digraph that is not super arc connected. Then $A \subset V$ is a super atom of X if and only if A induces a clique of X and $|A| = d^+(X)$.*

Proof. Since cliques and directed cycles are super arc connected, we may assume that $1 < d^+(X) < n - 1$. First suppose A is a super atom of X . Then $|A|(d^+(X) - d^+(A)) = \lambda(X) = d^+(X)$ and thus $(|A| - 1)(d^+(X) - d^+(A)) = d^+(A) \leq |A| - 1$. But then we have $d^+(A) = |A| - 1$, so that A is a clique, and $d^+(X) = |A|$. Next suppose A is a clique of X with $|A| = d^+(X)$. Then each vertex of A has exactly one outarc in X whose terminal vertex is not in A . Therefore $|\omega^+(A)| = |A| = d^+(X) = \lambda(X)$, so that A is a fragment of X . Since we showed in the first part of this proof that the super atoms of X have cardinality $d^+(X)$, it then follows that A is a super atom.

THEOREM 3 (Local characterization). *Let $X = (V, E)$ be a strongly connected, vertex-transitive digraph that is neither a clique nor a directed cycle. Then X is not super arc connected if and only if X contains a clique of size $d^+(X)$.*

Proof. The only if part of this theorem is immediate from Lemma 3. Thus, suppose that X contains a clique with vertex set A and that $|A| = d^+(X)$. Since X is neither a directed cycle nor a clique, we have $1 < d^+(X) = |A| < n - 1$. Moreover, $|\omega^+(A)| = |A| = d^+(X) = \lambda(X)$. It then follows that A is a nontrivial fragment of X with $|V \setminus A| > 1$, and hence that X is not super connected.

THEOREM 4 (Algebraic characterization). *Let G be a group and S be a generating set that is not of the form $\{s, s^{-1}\}$. Then $\text{Cay}(G, S)$ is not super arc connected if and only if $S = (H \setminus \{1\}) \cup \{t\}$ for some nontrivial subgroup H of G and some element t of $G \setminus H$.*

Proof. The if part of the theorem may be verified by showing that $\omega^+(H) = |H| = |S|$, which is straightforward and omitted. Thus, assume that $\text{Cay}(G, S)$ is not super arc connected. Since the Cayley digraph is vertex transitive, we know from Theorem 3 that the identity 1 lies in a clique with vertex set H of size $d^+(X)$. By Lemma 3, H is a super atom. If $h \in H$, then hH is a super atom with $H \cap hH \neq \emptyset$. Since $\text{Cay}(G, S)$ is not a symmetric cycle we have, by Lemma 2, that $hH = H$. Therefore, H is a subgroup of G . Since H induces a clique, $(1, h)$ is an arc of the Cayley graph for any $h \in H \setminus \{1\}$ so that $H \setminus \{1\} \subset S$. Since each vertex of H has exactly one outarc with terminal vertex not in H , $S = (H \setminus \{1\}) \cup \{t\}$ for some element $t \in G \setminus H$.

COROLLARY 2. *A connected Abelian Cayley graph that is not super line connected is either a cycle or isomorphic to $K_m \times K_2$ for some $m \geq 2$.*

Proof. Let $\text{Cay}(G, S)$ be a connected symmetric Cayley digraph with G Abelian. Then $S = -S$ and, by Theorem 4, $S = (H \setminus \{0\}) \cup \{t\}$ for some nontrivial subgroup H of G and some $t \in G \setminus H$. It then follows that $2t = 0$ and hence H has index 2. Thus, the Cayley graph consists of the two cliques H and $t + H$ together with a perfect matching between the two cliques and hence is isomorphic to $K_m \times K_2$, where $m = |H| \geq 2$. It is simple to verify that $K_m \times K_2$ is isomorphic to the Abelian Cayley graph $\text{Cay}(\mathbb{Z}_m \times \mathbb{Z}_2, \mathbb{Z}_m \times \{0\} \cup \{(0, 1)\})$, which completes the proof.

The Boesch–Wang [3] characterization of nonsuper line connected circulants follows from Corollary 2 and the following simple proposition.

PROPOSITION 1. *If $K_m \times K_2$ is isomorphic to a circulant then m is odd.*

Proof. Suppose $K_m \times K_2$ is isomorphic to $\text{Cay}(\mathbb{Z}_{2m}, S)$ for some generating set S . By Theorem 4, $S = (H \setminus \{0\}) \cup \{t\}$ with H a nontrivial subgroup of $\mathbb{Z}_{2m}$ and t an

element of $\mathbb{Z}_{2m} \setminus H$. Since the Cayley digraph must be symmetric, t must have order 2 and hence $t = m$. Moreover, H is a subgroup of index 2 and thus $H = 2\mathbb{Z}_{2m}$. Since $t = m$ cannot be in H , we conclude that m is odd.

In the next section, we provide a specific structural description for the Cayley digraphs of the form $\text{Cay}(G, (H \setminus \{1\}) \cup \{t\})$ when H is a normal subgroup of G and $t \in G \setminus H$. The digraphs that occur for these cases are then used in the structural characterization theorem for general vertex-transitive digraphs that are not super arc connected.

4. General vertex-transitive digraphs. Throughout this section we assume that $X = (V, E)$ is a strongly connected, vertex-transitive digraph that is not isomorphic to a symmetric cycle and is not super arc connected.

It is immediate from Theorem 3 and Lemma 3 that the super atoms partition the vertex set of X , that each super atom induces a clique of size $d^+(X)$, and that each vertex has exactly one in-arc and one out-arc that lies in no atom. The arcs of X that have both endpoints in the same super atom will be called *atomic arcs* of X ; all other arcs will be called *nonatomic arcs*. Since automorphisms of X must preserve super atoms, they must also preserve nonatomic arcs. If we delete all atomic arcs from X , the resulting digraph is a disjoint union of directed cycles, called *fundamental loops* of X . Obviously, automorphisms of X must carry fundamental loops onto fundamental loops. The set of super atoms with nonempty intersection with a fundamental loop C will be called the *orbit* of C and the cardinality of the orbit of C will be called the *period* of C . Again, automorphisms of X must preserve orbits. We will denote by A_u the super atom containing vertex u .

LEMMA 4. *If (u, v) and (x, y) are arcs of a fundamental loop C and $A_u = A_x$, then $A_v = A_y$.*

Proof. If we choose an automorphism φ of X with $\varphi(u) = v$, then $\varphi(C) = C$ and hence φ induces an automorphism of cycle C . Since φ maps u to its immediate successor v on C , it must map x to its immediate successor y . But then $A_y = \varphi(A_x) = \varphi(A_u) = A_v$.

Let us say that two fundamental loops C, C' are *parallel* if for any two super atoms A and B of X , there is an arc of C from A to B if and only if there is an arc of C' from A to B . Note that a fundamental loop is parallel to itself. We define the *multiplicity* of a fundamental loop C to be the number of fundamental loops parallel with C . The next lemma is now apparent.

LEMMA 5. *There are positive integers p, r, m such that*

- 1) *All fundamental loops of X have period p .*
- 2) *All fundamental loops of X have multiplicity m .*
- 3) *For any fundamental loop C and super atoms A and B of X , the number of arcs of C initiating in A and terminating in B is either r or 0 .*

Moreover, $p \geq 2$ and each fundamental loop has length rp .

The integers p, r, m of Lemma 5 will be called *period*, *return count*, and *multiplicity* of the digraph X . Note that if all fundamental loops of X are parallel, then X has exactly p super atoms and each fundamental loop induces the same cyclic order on the super atoms. Using this observation, the following lemma is easily deduced.

LEMMA 6. *Let $X \neq C_n$ be a strongly connected vertex-transitive digraph that is not super arc connected, and let p, r, m be the period, return count, and multiplicity, respectively, of X . Then all the fundamental loops of X are parallel if and only if $mr \geq 2$ and X is isomorphic to the digraph $X_{p,r,m}$ with vertex set*

$$\{1, 2, \dots, rp\} \times \{1, 2, \dots, m\}$$

and arc set

$$\{((x_1, y_1), (x_2, y_2)) : [x_1 \equiv x_2 \pmod{p}] \text{ or } [y_1 = y_2 \text{ and } x_2 \equiv x_1 + 1 \pmod{p}]\}.$$

In the previous lemma mr is the cardinality of a super atom and hence must be at least 2. It should be noted that the digraph $X_{p,r,m}$ as given in Lemma 6 is isomorphic to the abelian Cayley digraph

$$\text{Cay}(\mathbf{Z}_{rp} \times \mathbf{Z}_m, \{(ip, j) : 0 \leq i < r, 0 \leq j < m\} \cup \{(1, 0)\}).$$

The next theorem, while stated for Abelian Cayley digraphs, is valid for any Cayley digraph of the form $\text{Cay}(G, S)$ with $S = (H \setminus \{1\}) \cup \{t\}$, $t \in G \setminus H$ and H a normal subgroup of G .

THEOREM 5. *A strongly connected Abelian Cayley digraph is not super arc connected if and only if it is a symmetric cycle or is isomorphic to the digraph $X_{p,r,m}$ for some positive integers m, r, p with $p \geq 2$ and $mr \geq 2$.*

Proof. By Lemma 6 we need only show that all fundamental loops in $\text{Cay}(G, S)$ are parallel if G is Abelian and $S = (H \setminus \{0\}) \cup \{t\}$ for some nontrivial subgraph H of G and element t of $G \setminus H$. First note that if h is an element of the super atom H of 0, then its unique nonatomic outarc $(h, h + t)$ terminates in the coset $t + H = H + t$. It is obvious that the super atoms are just the (left) cosets of H . Thus all nonatomic arcs initiating in the super atom H terminate in the super atom $t + H$. It then follows that all fundamental loops are parallel, completing the proof.

We now turn to the characterization of strongly connected vertex-transitive digraphs that are not super arc connected. First note that in the case of graphs, which is to say symmetric digraphs, fundamental loops must have length 2. This implies that the digraph has period 2 and return count 1. Thus the only remaining variable is the multiplicity, which becomes the number of undirected edges connecting two “adjacent” super atom cliques. The next key fact in the characterization in the undirected case is that contracting the super atoms to a point and removing self-loops and multiple edges results in a 1-transitive graph. Thus, we need something to play the role for directed graphs that 1-transitive graphs play for undirected graphs. Unfortunately, the digraph obtained by contracting super atoms loses the integrity of the system of automorphism invariant fundamental loops, so it does not appear suitably by itself. Our attempt to capture this structure led us to a notion of “oriented” hypergraph.

A p -uniform oriented hypergraph is a pair $Y = (V, F)$ where V is a nonempty set and F is a set of ordered p -tuples of elements of V such that if $f = (f_1, \dots, f_p)$ is in F , then (1) $f_i \neq f_j$ when $i \neq j$ and (2) every cyclic permutation of f is in F . The elements of F are called the *faces* of Y . If vertex u appears as a component of face f of Y , we say that f is *incident* on u . Two faces are said to be *equivalent* if one is a cyclic permutation of the other. The *degree* of a vertex is the cardinality of the set of (cyclic permutation) equivalence classes of faces incident with the vertex. If φ is a function from V to itself, we use the same name for the function on p -tuples of elements of V given in the usual way: $\varphi((f_1, \dots, f_p)) = (\varphi(f_1), \dots, \varphi(f_p))$. An *automorphism* of Y is a bijection φ of V such that for any p -tuple f over V , f is in F if and only if $\varphi(f)$ is in F . We say that Y is *1-transitive* if for any vertices u, v and any faces f, g with f incident on u and g incident on v , there is an automorphism φ of Y with $\varphi(u) = v$ and $\varphi(f)$ a cyclic permutation of g . Clearly if Y is a 1-transitive p -uniform oriented hypergraph then all vertices have the same degree d in Y ; we then refer to d as the *degree* of Y . The *1-skeleton* of Y is the digraph Y_1 with vertex set V and arcs all pairs (u, v) such that there is a face f of Y such that $f_1 = u$ and $f_2 = v$. We say that Y is *strongly connected* if its 1-skeleton is strongly connected. It is interesting to note that Y_1 is strongly connected if and only if it is connected.

It is easy to see that a 2-uniform oriented hypergraph and its 1-skeleton are in fact the same. Moreover, such a hypergraph is nothing more than a symmetric digraph, which is to say an undirected graph, and the notions of 1-transitivity coincide in this case. With

this interpretation, the characterization theorem we present here will, when restricted to 2-uniform oriented hypergraphs, produce the characterization theorem given in [7] for undirected graphs.

Let $X = (V, E)$ be a strongly connected vertex-transitive digraph that is not super arc connected, and let p, r, m be the period, return count, and multiplicity, respectively, of X . The *fundamental loop hypergraph* of X is the pr -uniform oriented hypergraph with vertex set the set of super atoms of X and face set the set of all pr -tuples $(A_{u_1}, \dots, A_{u_{pr}})$ such that $(u_1, u_2), (u_2, u_3), \dots, (u_{pr-1}, u_{pr})$ are arcs of some fundamental loop C of X . Since automorphisms of X preserve super atoms and fundamental loops, each such automorphism induces in the obvious way an automorphism of the fundamental loop hypergraph of X . It is thus a simple matter to show that the fundamental loop hypergraph of X is 1-transitive.

We now describe how to construct a strongly connected vertex-transitive digraph X when given positive integers $p \geq 2, r, m$ and a strongly connected, 1-transitive, pr -uniform oriented hypergraph Y of degree d . When $rm \geq 2$, X will fail to be super arc connected and Y will be the fundamental loop hypergraph of X . First we choose a system $F_{=}$ of distinct representatives for the (cyclic permutation) equivalence classes of F . Let L be the digraph $X_{p,r,m}$ defined in Lemma 6. If $rm = 1$, then L is a directed cycle of length p . If $rm > 1$ then, by Lemma 6, L is a strongly connected, non super arc connected digraph with period p , return count r , and multiplicity m , and in which all fundamental arcs are parallel. Note that, in both cases, L has exactly p maximal cliques of size $d^+(X)$. For each face f in $F_{=}$ we choose a copy L_f of L , with the copies associated with different faces disjoint. For each face f of Y , the fundamental loops of L_f induce a cyclic order on the maximal cliques of L_f ; it should be noted that each loop "traverses" the maximal cliques in this order exactly r times. Since the face f obviously induces a cyclic order on the vertices of f , we can label each maximal clique of L_f with a vertex of f in such a way that the cyclic order of the maximal cliques is compatible with the cyclic order on their labels. For each vertex u of Y , we have exactly d disjoint complete symmetric digraphs of order rm labeled u , because each vertex lies on exactly d faces from $F_{=}$; we next add the necessary arcs so that their union is a complete symmetric digraph (which will obviously have order drm). The resulting digraph will be denoted $\rho(Y, r, m)$. The final theorem of the paper now admits a straightforward proof, which we omit.

THEOREM 6. *Let X be a strongly connected vertex-transitive digraph that is not a cycle. Then X is not super arc connected if and only if $X = \rho(Y, r, m)$ for some strongly connected, 1-transitive pr -uniform oriented hypergraph Y of degree d and some integers r, m with $p > 1$ and $rm > 1$.*

REFERENCES

- [1] D. BAUER, F. BOESCH, C. SUFFEL, AND R. TINDELL, *Connectivity extremal problems and the design of reliable probabilistic networks*, The Theory and Application of Graphics, G. Chartrand, Y. Alevi, D. Goldsmith, L. Lesniak-Foster, and D. Lick, eds., John Wiley, New York, 1981, pp. 45–54.
- [2] F. BOESCH AND R. TINDELL, *Connectivity and symmetry in graphs*, Graphs and Applications, F. Harary and J. Maybee, eds., John Wiley, New York, 1985, pp. 53–67.
- [3] F. BOESCH AND J. WANG, *Super line connectivity properties of circulant graphs*, SIAM J. Algebraic Discrete Methods, 7 (1986), pp. 89–98.
- [4] J. FÀBREGA AND M. A. FIOL, *Maximally connected digraphs*, J. Graph Theory, 13 (1989), pp. 657–668.
- [5] Y. O. HAMIDOUNE, *Quelques problèmes de connexité dans les graphes orientés*, J. Combin. Theory Ser. B, 30 (1981), pp. 1–10.
- [6] W. MADER, *Minimale n -fach kantenzusammen hängen den Graphen*, Math. Ann., 23 (1971), pp. 21–28.
- [7] R. TINDELL, *Edge connectivity properties of symmetric graphs*, preprint, Stevens Institute of Technology, Hoboken, N.J., 1982.
- [8] C. S. YANG, J. F. WANG, J. Y. LEE, AND F. T. BOESCH, *Reliability properties of the hypercube network*, Comput. Math. Appl., 14 (1987), pp. 541–548.

THE DOMATIC NUMBER PROBLEM IN INTERVAL GRAPHS*

TUNG-LIN LU†, PEI-HSIN HO†, AND GERARD J. CHANG†

Abstract. A set of vertices D is a dominating set of a graph $G = (V, E)$ if every vertex in $V - D$ is adjacent to a vertex in D . The domatic number $d(G)$ of a graph $G = (V, E)$ is the maximum number k such that V can be partitioned into k disjoint dominating sets $D_1, \dots, D_k$. The main purpose of this paper is to give linear algorithms for the domatic number problem in interval graphs. This paper also proves that $d(G) = \delta(G) + 1$ for any interval graph G , where $\delta(G)$ is the minimum degree of a vertex in G .

Key words. dominating set, domatic number, interval graph, degree, linear algorithm, NP-complete

AMS(MOS) subject classifications. 05C70, 68R10

1. Introduction. A set of vertices D is a *dominating set* of a graph $G = (V, E)$ if every vertex in $V - D$ is adjacent to a vertex in D . The *domatic number* $d(G)$ of a graph $G = (V, E)$ is the maximum number k such that V can be partitioned into k disjoint dominating sets $D_1, \dots, D_k$. The domination set problem and its variations have been extensively studied; however, the domatic number problem is much less well known.

Lower bounds and upper bounds for the domatic number were studied in [4]–[6], [9]–[11], and [13]. In particular, [6] showed that $d(G) \leq \delta(G) + 1$ for any graph G , where $\delta(G)$ is the minimum degree of a vertex in G . G is *domatically full* if $d(G) = \delta(G) + 1$. Cockayne and Hedetniemi [6] determined $d(G)$ for some special classes of graphs; consequently, K_n , $\bar{K}_n$, C_{3n} , trees and maximal outerplanar graphs are domatically full.

The domatic number problem is NP-complete for general graphs [7] and circular-arc graphs [2]. The problem is solved in $O(n^2 \log n)$ time for proper circular-arc graphs [2], $O(n^{2.5})$ time for interval graphs [1] and $O(n \log n)$ time for proper interval graphs [1].

The main purpose of this paper is to give linear algorithms for the domatic number problem in interval graphs. As a by-product we also prove that interval graphs are domatically full.

An *interval family* is a set of intervals on the real line. An interval family is *proper* if no interval is properly contained within another interval. A graph is a (*proper*) *interval graph* if there is a one to one correspondence between the vertices of the graph and the intervals of a (proper) interval family such that two vertices are joined by an edge if and only if their corresponding intervals overlap.

Interval graphs have been extensively studied and used as models for many real world problems. In particular, they have applications in archaeology, genetics, ecology, psychology, traffic control, computer scheduling, storage information retrieval, and electronic circuit design (see [8], [12]).

Booth and Lueker [3] gave a linear algorithm for deciding whether a given graph is an interval graph and constructing, in the affirmative case, the required interval family. In this paper we begin with the assumption that G is known to be an interval graph, and a corresponding interval family is given.

* Received by the editors June 30, 1989; accepted for publication (in revised form) March 6, 1990. This research was supported by the National Science Council of the Republic of China under grant NSC-77-0208-M009-21.

† Department of Applied Mathematics, National Chiao Tung University, Hsinchu 30050, Taiwan, Republic of China.

2. Notation and assumptions. Suppose $I = \{1, \dots, n\}$ is the interval family for an interval graph G produced by the linear algorithm in [3], where interval i is equal to $[a_i, b_i]$ for $i = 1, \dots, n$. a_i is the *left endpoint* of interval i and b_i the *right endpoint*. Without loss of generality we may assume that $\{a_1, \dots, a_n, b_1, \dots, b_n\} = \{1, 2, \dots, 2n\}$.

For the sake of simplicity we will denote an interval graph G as $G(I)$ and deal with intervals instead of vertices. In this way, the *closed neighborhood* $N[i]$ of an interval i is the set of all intervals that overlap with interval i . A dominating set for $G(I)$ corresponds to a subset S of intervals in I such that every interval in I overlaps with at least one interval in S .

For each interval i $\text{next}(i)$ is the interval j such that b_j is as small as possible but satisfying $b_i < a_j$; $\text{next}(i)$ is null if no such j exists.

3. The algorithms. In this section we will give two efficient algorithms for the domatic number problem in interval graphs $G(I)$. For technical reasons, we first augment I with two “dummy” intervals 0 and $n + 1$ such that $a_0 = -1$, $b_0 = 0$, $a_{n+1} = 2n + 1$ and $b_{n+1} = 2n + 2$. We then construct an acyclic directed graph H as follows. The nodes of H correspond to the intervals in $I' \equiv I \cup \{0, n + 1\}$. There is a directed arc (i, j) in H if and only if $j \in N[\text{next}(i)]$. Note that if (i, j) is an arc in H , then $b_i < b_j$. This guarantees that H is acyclic. Since $N[n + 1] = \{n + 1\}$, $(i, n + 1)$ is an arc in H if and only if $\text{next}(i) = n + 1$.

LEMMA 3.1. *Any directed path from node 0 to node $n + 1$ in H corresponds to a dominating set for $G(I)$.*

Proof. Suppose $\langle i_0, i_1, \dots, i_r \rangle$ is a directed path from node 0 to node $n + 1$ in H . By the definition of an arc in H , $b_{i_0} < b_{i_1} < \dots < b_{i_r}$. For any interval $j \in I$, choose an index s such that $b_{i_{s-1}} < a_j < b_{i_s}$. Suppose intervals j and i_s do not overlap. Then $b_{i_{s-1}} < a_j < b_j < a_{i_s} < b_{i_s}$. Let $k = \text{next}(i_{s-1})$. By the definition of function next , $b_k \leq b_j$ and so intervals k and i_s do not overlap. That implies $i_s \notin N[\text{next}(i_{s-1})]$, on contradicting that (i_{s-1}, i_s) is an arc in H . Therefore $j \in N[i_s]$ and so $\{i_1, \dots, i_{r-1}\}$ is a dominating set for $G(I)$. $\square$

A dominating set of $G(I)$ does not necessary correspond to a directed path from node 0 to node $n + 1$ in H . So we cannot conclude immediately, as in [1], that the domatic number of $G(I)$ is equal to the maximum number of disjoint paths from node 0 to node $n + 1$ in H . In fact our definition of directed graph H is different from that in [1]. The way our algorithms work is by means of the following duality relation.

LEMMA 3.2 [6] (weak duality inequality). $d(G) \leq \delta(G) + 1$ for any graph G .

The main idea of our algorithms is to find $\delta + 1$ disjoint dominating sets in $G(I)$, or equivalently $\delta + 1$ disjoint paths from node 0 to node $n + 1$ in H . We will present two algorithms for this purpose. The first algorithm finds the dominating sets one by one. The second algorithm finds all dominating sets simultaneously. Section 4 implements these algorithms and shows that their running times are linear.

ALGORITHM D1

```

initially all intervals are unlabeled;  $k \leftarrow 0$ ;
loop
   $i \leftarrow 0$ ;
  while ( $\text{next}(i) \neq n + 1$ ) do
     $j \leftarrow \text{next}(i)$ ;
    if  $N[j]$  has no unlabeled intervals then STOP;
    choose an unlabeled interval  $h \in N[j]$  with largest left endpoint;
    label  $h$  by  $k + 1$ ;

```


$i \leftarrow h;$
end while;
 $k \leftarrow k + 1;$
forever.

Suppose k^* is the final k when Algorithm D1 stops. Let D_i be the set of all intervals labeled by i . By Lemma 3.1, we have k^* disjoint dominating sets $D_1, \dots, D_{k^*}$.

LEMMA 3.3. *There exists an interval j such that $|N[j]| = k^*$.*

Proof. When Algorithm D1 stops there exists some interval j' such that all intervals in $N[j']$ are labeled by integers between 1 and k^* . Let j be such an interval with largest left endpoint.

Suppose $N[j]$ contains two distinct intervals p and q of the same label k . Assume intervals in D_k are labeled in the order $\dots p = p_1, p_2, \dots, p_m = q, \dots$. By the definition of function next,

$$(3.1) \quad b_p = b_{p_1} < a_{\text{next}(p_1)} < b_{p_2} < a_{\text{next}(p_2)} < \dots < b_{p_{m-1}} < a_{\text{next}(p_{m-1})} < b_{p_m} = b_q.$$

Also $a_j < b_p$ since $p \in N[j]$. Let $r = \text{next}(p_{m-1})$. Then $a_j < a_r$. By the choice of j , $N[r]$ has an interval s which is unlabeled or is labeled by $k^* + 1$. Suppose $a_s > a_q$. Since $s, q \in N[r]$, by Algorithm D1, interval s would be labeled by k before interval q being labeled by k . So $a_s < a_q$. Also $a_q < b_j$ since $q \in N[j]$. Then $a_s < b_j$. On the other hand,

$$a_j < b_p < a_r < b_s.$$

The first inequality follows from that $p \in N[j]$, the second is part of (3.1), and the third from $s \in N[r]$. Both $a_s < b_j$ and $a_j < b_s$ imply that $s \in N[j]$, a contradiction to the assumption that all intervals in $N[j]$ are labeled by integers between 1 and k^* but that s is not. Hence all intervals in $N[j]$ have distinct labels, i.e., $|N[j]| = k^*$. $\square$

By Lemmas 3.2 and 3.3 we have

$$\delta + 1 = \min_{i \in I} |N[i]| \leq |N[j]| = k^* \leq d(G(I)) \leq \delta + 1,$$

and so in fact the above inequalities are equalities.

THEOREM 3.4 (strong duality theorem). $d(G) = \delta(G) + 1$ for any interval graph G .

THEOREM 3.5. *Algorithm D1 works for solving the domatic number problem in interval graphs.*

The second algorithm follows from the fact that the outdegree of each node i in H such that $(i, n + 1)$ is not an arc in H is $|N[\text{next}(i)]| \geq \delta + 1$. We will describe our algorithm in terms of H here and implement it in terms of intervals in the next section.

ALGORITHM D2

initially all nodes in H are unlabeled;
 find a topological sort $i_0 = 0, i_1, i_2, \dots, i_n, i_{n+1} = n + 1$ for the nodes of H
 (i.e., (i_p, i_q) is an arc in H implies $p < q$);
 label the first r intervals (according to the topological sort)
 in $N[\text{next}(0)]$ by $1, \dots, r$ respectively;
for $r \leftarrow 1$ to n do
 if $(i_r, n + 1)$ is not an arc in H and i_r is labeled by k then
 (*) choose an unlabeled node j with (i_r, j) is an arc in H and label j by k ;
end for.

Since $i_0 = 0, i_1, i_2, \dots, i_n, i_{n+1} = n + 1$ is a topological sort for the nodes of H , the fact that each node of H such that $(i, n + 1)$ is not an arc in H has outdegree at least

$\delta + 1$ implies that each subgraph H_r of H induced by $\{i_r, i_{r+1}, \dots, i_{n+1}\}$ has the same property. So in Step(*) of Algorithm D2, we can always find such a node j . For each k between 1 and $\delta + 1$, $\{0, n + 1\}$ together with all nodes labeled by k form a directed path from node 0 to node $n + 1$. The algorithm does produce $\delta + 1$ disjoint paths from node 0 to node $n + 1$. So we have another way to verify Theorem 3.4 and solve the domatic number problem.

THEOREM 3.6. *Algorithm D2 works for solving the domatic number problem in interval graphs.*

4. Implementation. This section gives two implementations for Algorithm D1 and one implementation for Algorithm D2. The implementations show that the algorithms are linear.

We assume $\{a_1, \dots, a_n, b_1, \dots, b_n\} = \{1, 2, \dots, 2n\}$. We also need the information that each $p \in [1, 2n]$ is equal to a_i or b_i for some interval $i \in I$. This can be done by a simple do loop. We can use one array to indicate that p is a left or a right endpoint and another array to indicate that p is the endpoint of an interval $i \in I$.

To implement our algorithms, we first must produce function next. This can be done in $O(n)$ time as follows. The algorithm scans the endpoints from $2n$ backward to -1 . In the algorithm, s is the interval with smallest right endpoint among the intervals whose left endpoints have been scanned.

```

/*Calculate function next*/
s ← n + 1;
for p ← 2n to -1 step -1 do
  case 1: p = ai for some interval i ∈ I'
    if bi < bs then s ← i;
  case 2: p = bi for some interval i ∈ I'
    next(i) ← s;
end for.

```

Having calculated function next, we now present the first implementation of Algorithm D1. The main difficulty in Algorithm D1 is how to “choose an unlabeled interval $h \in N[i]$ with largest left endpoint.” We use a stack to store candidates of such intervals such that an interval with larger left endpoint is closer to the top of the stack.

```

/*First implementation of Algorithm D1*/
label(i) ← 0 for all intervals i ∈ I; k ← 0;
loop
  stack S ← ∅; h ← 0;
  for p ← 1 to 2n do
    j ← next(h);
    case 1: p = ai for some interval i ∈ I.
      if label(i) = 0 then push i into S;
    case 2: p = bi for some interval i ∈ I.
      if i = j then
        while (S ≠ ∅) do
          pop h from S;
          if bh > aj then
            [label(h) ← k + 1;
             if next(h) = n + 1 then goto (**) else goto (*)]
          end while;
        STOP the algorithm;
      end if;

```

```

(*) end for;
(**)  $k \leftarrow k + 1$ ;
forever.

```

As in the proof after Lemma 3.3, there are $\delta + 2$ iterations in the loop. Each iteration needs $O(n)$ steps. So this is an $O(\delta n + n) = O(|E| + |V|)$ time implementation of Algorithm D1.

As a second implementation, we only have to produce “sorted” closed neighborhoods $N[i]$ for all intervals $i \in I$. Once we have sorted closed neighborhoods, Algorithm D1 is easily implemented. We now give an $O(|E| + |V|)$ algorithm for producing the closed neighborhood of all intervals in I as follows.

```

/*Each closed neighborhood  $N[i]$  is sorted according to left endpoints*/
doubly linked list  $L \leftarrow \phi$ ;
for  $p \leftarrow 1$  to  $2n$  do
  case 1:  $p = a_i$  for some interval  $i \in I$ 
     $N[i] \leftarrow L$ ;
     $L \leftarrow L + i$ ; /*remember the address of  $i$  in  $L$ */
    for each  $j \in L$  do  $N[j] \leftarrow N[j] + i$ ;
  case 2:  $p = b_i$  for some interval  $i \in I$ 
    delete  $i$  from  $L$ ;
end for.

```

Finally, we shall implement Algorithm D2 in terms of intervals rather than the directed graph H . Note that if (i, j) is an arc in H , then $b_i < b_j$. A simple topological sort of the nodes of H is to sort intervals in I' according to their right endpoints.

```

/*Implementation of algorithm D2*/
label( $i$ )  $\leftarrow 0$  for all intervals  $i \in I$ ;
 $p \leftarrow 1$ ;
for  $k \leftarrow 1$  to  $\delta + 1$  do
  if  $p = a_i$  for some interval  $i$  which intersects interval next(0) then label( $i$ )  $\leftarrow k$ ;
   $p \leftarrow p + 1$ ;
end for;
for  $p \leftarrow 1$  to  $2n$  do
  if  $p = b_i$  for some interval  $i \in I$  then
    if label( $i$ )  $\neq 0$  and next( $i$ )  $\neq n + 1$ 
      then [get  $j \in N[\text{next}(i)]$  with label( $j$ ) = 0; label( $j$ )  $\leftarrow$  label( $i$ )]
  end for.

```

In the above implementation, each closed neighborhood is considered as an unsorted single linked list. To get some interval j in $N[\text{next}(i)]$ we simply get the first element of the list and delete it from the list. If its label is nonzero, we continue to get the first element from the remaining list until an interval with label zero is found.

5. Proper interval graphs. In the case where $G(I)$ is a proper interval graph, assume $a_1 < a_2 < \dots < a_n$. It is easy to see that each closed neighborhood $N[i]$ is a consecutive set, i.e., $N[i] = \{j, j + 1, \dots, j + k\}$ for some j and k . Since each $|N[i]| \geq \delta + 1$, we can construct $\delta + 1$ disjoint dominating sets $D_k = \{i \in I : i \equiv k \pmod{\delta + 1}\}$, $k = 1, \dots, \delta + 1$. This is a much simpler method than that in [1].

6. Conclusion. The main results of this paper give two linear algorithms for the domatic number problem on interval graphs. As a by-product we also show that interval

graphs are domatically full. A much simpler method for the problem in proper interval graphs is also discussed. We suspect that similar results can be obtained for the problem in strongly chordal graphs or even for chordal graphs.

REFERENCES

- [1] A. A. BERTOSSI, *On the domatic number of interval graphs*, Inform. Proc. Lett., 28 (1988), pp. 275–280.
- [2] M. A. BONUCCELLI, *Dominating sets and domatic number of circular arc graphs*, Disc. Appl. Math., 12 (1985), pp. 203–213.
- [3] K. BOOTH AND G. S. LUEKER, *Testing for consecutive ones property, interval graphs and graph planarity using PQ-tree algorithms*, J. Comput. System Sci., 13 (1976), pp. 335–379.
- [4] E. J. COCKAYNE, *Domination of undirected graphs—a survey*, Lecture Notes in Math. 642, Springer-Verlag, Berlin, New York, (1978), pp. 141–147.
- [5] E. J. COCKAYNE AND S. T. HEDETNIEMI, *Optimal domination in graphs*, IEEE Trans. Circuits Systems, CAS-22 (1975), pp. 41–44.
- [6] ———, *Towards a theory of domination in graphs*, Networks, 7 (1977), pp. 247–261.
- [7] M. R. GAREY AND D. S. JOHNSON, *Computer and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, CA, 1979.
- [8] M. C. GOLUMBIC, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York, 1980.
- [9] F. JAEGAR AND C. PAYAN, *Relations du type Nordhans–Gaddum pour le nombre d’absorption d’un graphe simple*, C. R. Acad. Sci. Paris, Ser. A, 274 (1972), pp. 728–730.
- [10] R. LASKAR AND H. B. WALIKAR, *On domination related concepts in graph theory*, Lecture Notes in Math. 885, Springer-Verlag, Berlin, New York, (1980), pp. 308–320.
- [11] O. ORE, *Theory of Graphs*, Amer. Math. Soc. Colloq. Publ. 38, Providence, RI, 1962.
- [12] F. S. ROBERTS, *Graph Theory and its Applications to Problems of Society*, CBMS-NSF Regional Conference Series 29, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1978.
- [13] P. J. SLATER, *R-domination in graphs*, J. ACM, 23 (1976), pp. 446–450.

POLYNOMIAL ALGORITHMS FOR FINDING CYCLES AND PATHS IN BIPARTITE TOURNAMENTS*

Y. MANOUSSAKIS† AND Z. TUZA‡

Abstract. Efficient algorithms for finding Hamiltonian cycles, Hamiltonian paths, and cycles through two given vertices in bipartite tournaments are given.

Key words. polynomial algorithm, bipartite tournament, Hamiltonian cycle

AMS(MOS) subject classifications. 68C25, 68E10

1. Introduction. A bipartite tournament is a complete oriented bipartite graph. Here, by using known results on perfect matchings, we obtain $O(n^{2.5})$ algorithms for finding Hamiltonian cycles and Hamiltonian paths and an $O(n^3)$ algorithm for finding cycles through two specified vertices in a bipartite tournament. These problems are known to be NP-complete for digraphs (see [2], [4]) and linear for tournaments (see [6]).

Formally, throughout this paper $B(X, Y, E)$ denotes a bipartite tournament of order n with bipartition (X, Y) , vertex-set $V(B)$ and arc-set $E(B)$. Put $e = |E(B)|$, $|X| = a$ and $|Y| = b$. In the following we assume that $a \leq b$. When $a = b$, we say that B is balanced and when $b = a + 1$, we say that B is almost balanced. If x and y are vertices of B , then x dominates y if the arc (x, y) is present. Moreover, we define $\Gamma^-(x)$, $\Gamma^+(x)$ to be the vertices of B which, respectively, dominate, and are dominated by, the vertex x . The indegree, outdegree and degree of x are defined as $|\Gamma^-(x)|$, $|\Gamma^+(x)|$ and $|\Gamma^-(x)| + |\Gamma^+(x)|$, respectively, and are denoted $d^-(x)$, $d^+(x)$, and $d(x)$, respectively. The length of a path is the number of its arcs. A bipartite tournament B is strong, if for each two vertices x and y there is a path from x to y in B .

A matching M_{XY} from X into Y is a set of arcs of B within origin in X and end terminus in Y , such that any two arcs in M_{XY} are nonadjacent. An $(1, 1)$ -factor, or more simply a factor F , of B is a spanning subgraph such that $d^-(x) = 1 = d^+(x)$, for any vertex x in F . A factor consists of cycles $C_1, C_2, \dots, C_m$, where $m \geq 1$. It is well known that a bipartite tournament admits a factor F , if and only if it is balanced and has a matching M_{XY} and a matching M_{YX} , each of cardinality $|X|$.

In § 2 we give a method for finding Hamiltonian cycles in bipartite tournaments containing exactly two pairwise vertex-disjoint cycles.

In § 3, we give the main results of this work.

2. Preliminary results. Throughout this section, we suppose that B is strong and that it contains two pairwise vertex-disjoint cycles C_1 and C_2 covering all its vertices. Put $C_1: x'_1 \rightarrow y'_1 \rightarrow \dots \rightarrow x'_s \rightarrow y'_s \rightarrow x'_1$, $s \geq 2$, and $C_2: x_1 \rightarrow y_1 \rightarrow \dots \rightarrow x_t \rightarrow y_t \rightarrow x_1$, $t \geq 2$, $x'_i, x_i \in X$ and $y'_i, y_i \in Y$. Clearly $|V(C_1)| = 2s$, $|V(C_2)| = 2t$ and $2s + 2t = n$.

The results (Lemmas 2.1–2.4) of this section will be useful in order to validate the algorithms of § 3.

LEMMA 2.1. *If there exists a vertex x in C_1 such that either $\Gamma^+(x) \cap C_2 = \emptyset$ or $\Gamma^-(x) \cap C_2 = \emptyset$, then we may find a Hamiltonian cycle in B in at most $O(2s \cdot 2t)$ steps.*

* Received by the editors June 15, 1988; accepted for publication (in revised form) February 2, 1990.

† University of Orsay, L.R.I., Bât. 490, 91405 Orsay, France. This research was partially supported by PRC Math-Info.

‡ Computer and Automation Institute, Hungarian Academy of Sciences, H-1111 Budapest, Kende u. 13-17, Hungary.

Proof. Assume without loss of generality that there exists a vertex x in C_1 such that $\Gamma^-(x) \cap C_2 = \emptyset$, in other words, that x dominates any vertex of the other bipartition in C_2 . If the successor of x on C_1 is dominated by a vertex of C_2 , then we can easily find a Hamiltonian cycle in B . If, on the other hand, the successor of x dominates any vertex of C_2 , we replace x by its successor and repeat the same argument. It follows that, if we are unable to find a Hamiltonian cycle in B , then any vertex in C_1 dominates any vertex of the other bipartition in C_2 , a contradiction to the hypothesis that B is strong.

Clearly, in order to complete the above argument, we need no more operations than the number of arcs between C_1 and C_2 , so the proof of the lemma is complete.

LEMMA 2.2. *Assume that y'_s dominates x'_2 . We may find either a Hamiltonian cycle or else two vertex-disjoint cycles C'_1 and C'_2 such that $V(C'_1) = V(C_1) - \{x'_1, y'_1\}$, $V(C'_2) = V(C_2) \cup \{x'_1, y'_1\}$ and $V(C'_1) \cup V(C'_2) = V(B)$ in B within at most $O(2t)$ operations.*

Proof. Choose two vertices x_j and y_i in C_2 , $i \geq j$, (i, j are taken modulo t) such that both the arcs (x'_1, y_i) and (x_j, y'_1) are present. Let P denote the path $x_j \rightarrow \cdots \rightarrow y_i$ on C_2 . If x_j precedes y_i on C_2 , we can easily find a Hamiltonian cycle of B . Consequently, assume that there are some (in fact, no less than two) intermediate vertices on P . Now, if x_i dominates y'_1 , the cycle $x_1 \rightarrow y_1 \rightarrow \cdots \rightarrow x_i \rightarrow y'_1 \rightarrow x'_2 \rightarrow y'_2 \rightarrow \cdots \rightarrow x'_1 \rightarrow y_i \rightarrow \cdots \rightarrow x_1$ is Hamiltonian. Assume that y'_1 dominates x_i . If y_{i-1} dominates x'_1 , then we may put $C'_2 : x_1 \rightarrow y_1 \rightarrow \cdots \rightarrow y_{i-1} \rightarrow x'_1 \rightarrow y'_1 \rightarrow x_i \rightarrow y_i \rightarrow \cdots \rightarrow x_1$ and $C'_1 : y'_s \rightarrow x'_2 \rightarrow \cdots \rightarrow x'_s \rightarrow y'_s$, so assume that x'_1 dominates y_{i-1} . Replace now the path P by the path $x_j \rightarrow \cdots \rightarrow y_{i-1}$ and then repeat all the above arguments. Continuing in this way, we can finally see that, if we are unable to find the cycles C'_1 and C'_2 , as described in the lemma, then the vertex x'_1 dominates y_j . Consequently, in this final case, we can easily find a Hamiltonian cycle as required. Notice that the number of operations needed to complete the argument is proportional to $O(2t)$, so the proof is complete.

LEMMA 2.3. *We may find a Hamiltonian cycle in B within at most $O(2s, 2t)$ steps, if either (i) or (ii) below holds.*

(i) *The length of C_1 is four or*

(ii) *the length of C_1 is no less than six and moreover y'_s dominates x'_2 , x'_s dominates y'_2 , y'_{s-1} dominates x'_3 and so on.*

Proof. Put $C_2 : x_1 \rightarrow y_1 \rightarrow \cdots \rightarrow x_t \rightarrow y_t \rightarrow x_1$, where $t + s = n$, $x_i \in X$, and $y_i \in Y$. If for some vertex x'_i (respectively, y'_i) we have either $\Gamma^+(x'_i) \cap C_2 = \emptyset$ or $\Gamma^-(x'_i) \cap C_2 = \emptyset$ (respectively, either $\Gamma^+(y'_i) \cap C_2 = \emptyset$ or $\Gamma^-(y'_i) \cap C_2 = \emptyset$), then we may find a Hamiltonian cycle by Lemma 2.1, so assume that it is not the case. Now choose two vertices x_j and y_i in C_2 such that both the arcs (x'_1, y_i) and (x_j, y'_1) are present and subject to this requirement; the path $P : x_j \rightarrow \cdots \rightarrow y_i$ on C_2 is the minimum possible (for finding P , fix x_j and then find y_i such that $\Gamma^+(x'_1) \cap \{y_p, j \leq p \leq i-1\} = \emptyset$, then fix y_i and find x_j such that $\Gamma^-(y'_1) \cap \{x_p, j+1 \leq p \leq i\} = \emptyset$). If x_j precedes y_i on C_2 , we can easily find a Hamiltonian cycle in B . Consequently, in what follows, assume that there are some intermediate vertices on P .

Proof of (i). In this case, put $C_1 : x'_1 \rightarrow y'_1 \rightarrow x'_2 \rightarrow y'_2 \rightarrow x'_1$. Since P is of minimum length, we have $\Gamma^+(x'_1) \cap V(P) = \{y_i\}$ and $\Gamma^-(y'_1) \cap V(P) = \{x_j\}$. Now, if the arc (y'_2, x_i) is present, the cycle $y_{i-1} \rightarrow x'_1 \rightarrow y'_1 \rightarrow x'_2 \rightarrow y'_2 \rightarrow x_i \rightarrow y_i \rightarrow \cdots \rightarrow y_{i-1}$ is Hamiltonian and if the arc (y_{i-1}, x'_2) is present, then the cycle $y_{i-1} \rightarrow x'_2 \rightarrow y'_2 \rightarrow x'_1 \rightarrow y'_1 \rightarrow x_i \rightarrow y_i \rightarrow \cdots \rightarrow y_{i-1}$ is, once more, Hamiltonian. Consequently, in what follows, assume that both the arcs (x_i, y'_2) and (x'_2, y_{i-1}) are in $E(B)$. If $i-1 = j$, then the cycle $x_j \rightarrow y'_1 \rightarrow x'_2 \rightarrow y'_{i-1} \rightarrow x_i \rightarrow y'_2 \rightarrow x'_1 \rightarrow y_i \rightarrow \cdots \rightarrow x_j$ is Hamiltonian. If, on the other hand, $i-1 < j$, then by repeating all the above arguments we can see

that the arc (y'_2, x'_{i-1}) is present and therefore, the cycle $y_{i-2} \rightarrow x' \rightarrow y'_1 \rightarrow x'_2 \rightarrow y'_2 \rightarrow x_{i-1} \rightarrow \cdots \rightarrow y_{i-2}$ is, once more, Hamiltonian. This completes the proof of (i).

Proof of (ii). Apply repeatedly Lemma 2.2 by renaming appropriately the vertices of C'_1 and C'_2 in each repetition until we find either a Hamiltonian cycle or a cycle C'_1 of length four. In this final case, we use (i), if necessary, to complete the proof.

Clearly (i) or (ii) may be given with at most $O(2s \cdot 2t)$ operations, since each arc between C_1 and C_2 is counted constant times, so the proof of Lemma 3.3 is complete.

LEMMA 2.4. *We may find a Hamiltonian cycle in B within $O(2s \cdot 2t)$ steps.*

Proof. The following algorithm finds a Hamiltonian cycle in B .

Step 1. If there exists a vertex x'_i (respectively, y'_i) in C_1 such that either $\Gamma^+(x'_i) \cap C_2 = \emptyset$ or $\Gamma^-(x'_i) \cap C_2 = \emptyset$ (respectively, either $\Gamma^+(y'_i) \cap C_2 = \emptyset$ or $\Gamma^-(y'_i) \cap C_2 = \emptyset$), then find a Hamiltonian cycle by Lemma 2.1. Otherwise go to 2.

Step 2. If the length of C_1 is four, then find a Hamiltonian cycle by Lemma 2.3. Otherwise go to Step 3.

Step 3. Test if the arc (x'_2, y'_s) is present. If not, go to Step 4. Otherwise, find a vertex x'_{s-i} such that i is maximum and subject to this requirement (i) x'_{s-i} dominates y'_{i+2} but any other vertex x'_{s-j} , $j < i$, is dominated by y'_{j+2} and (ii) y'_{s-j} is dominated by x'_{j+2} . If x'_{s-i} does not exist, try to find a vertex y'_{s-i} such that it dominates x'_{i+2} but any other vertex x'_{s-j} , $j < i$, is dominated by y'_{j+2} and y'_{s-j} is dominated by x'_{j+2} . If neither x'_{s-i} nor y'_{s-i} exist, then we apply Lemma 2.3 to find a Hamiltonian cycle of B in $O(e)$ time. Otherwise, we put $C_1 : x'_{s-i} \rightarrow y'_{i+2} \rightarrow x'_{i+3} \rightarrow \cdots \rightarrow x'_{s-i-1} \rightarrow y'_{s-i-1} \rightarrow x'_{s-i}$ and $C_3 : y'_{s-i} \rightarrow x'_{s-i+1} \rightarrow \cdots \rightarrow x'_s \rightarrow y'_s \rightarrow x'_1 \rightarrow \cdots \rightarrow x'_{i+2} \rightarrow y'_{s-i}$. Then find a Hamiltonian cycle, C_2 say, in the bipartite tournament induced by $V(C_1) \cup V(C_2)$, by applying Lemma 2.3. Moreover, rename appropriately the vertices of C_1 and C_2 and then go to 2.

Step 4. Replace the cycles C_1 and C_2 by the cycles C'_1 and C'_2 given in Lemma 2.3, rename the vertices of C'_1 modulo 3 and then, if the arc (y'_s, x'_2) is not present, go to Step 3; otherwise, repeat the same arguments until a cycle C'_1 of length four is found. In this final case, we complete the proof by Lemma 2.3 (i).

3. Algorithms. Algorithm 3.1 below finds Hamiltonian cycles (if any) in bipartite tournaments in $O(n^{2.5})$ steps. It was inspired by the following result of [5]; however, our constructive proof is quite different.

THEOREM 3.1 (Häggkvist and Manoussakis). *A bipartite tournament B is Hamiltonian if and only if both the conditions below hold:*

- (i) B is strong and
- (ii) B has a factor.

ALGORITHM 3.1

INPUT: A bipartite tournament B .

We suppose that B is balanced, since otherwise it has no Hamiltonian cycles.

Step 1. Find a maximum M_{XY} and a maximum M_{YX} matching in B by using a known algorithm (see [3], [7]). If either $|M_{XY}| \leq |X| - 1$ or $|M_{YX}| \leq |X| - 1$ then stop; B has no Hamiltonian cycle by Theorem 3.1.

Step 2. Test if B is strong (see [7]). If it is not the case then stop; it has no Hamiltonian cycle by Theorem 3.1.

Step 3. Form a factor F , by considering M_{XY} and M_{YX} found in Step 1. Now let $C_1, C_2, \dots, C_m$, $m \geq 1$, be the cycles of F . If $m = 1$, we have finished, and if $m = 2$, we can complete the algorithm by Lemma 2.4. Assume $m \geq 3$. Now let B' denote the digraph obtained from B as follows: Replace any cycle C_i by a new vertex c_i and then add an arc (c_i, c_j) , $i \neq j$, $i, j = 1, 2, \dots, m$, in B' , if and only if there exists at least one arc from the cycle C_i to the cycle C_j in B . Clearly B' is either a tournament or a semicomplete digraph, i.e., a digraph that contains a spanning subgraph which is a tournament. Now, if B' is a tournament go to 4, otherwise go to 5.

Step 4. Clearly B' is strong and therefore Hamiltonian by a well-known theorem of Camion [1]. Find now a Hamiltonian cycle, $c_1 \rightarrow c_2 \rightarrow \dots \rightarrow c_m \rightarrow c_1$ say, in B' by using an algorithm given in [6], and then by using this cycle find a Hamiltonian cycle in B by taking appropriate arcs from C_1 to C_2 , from C_2 to C_3 and so on.

Step 5. Take two vertices c_i and c_j such that both the arcs (c_i, c_j) , and (c_j, c_i) are present in B' . Then put $V(C_i) \leftarrow V(C_i) \cup V(C_j)$ by applying Lemma 2.4 in the bipartite tournament induced by $V(C_i) \cup V(C_j)$. Now, we find a new digraph B' on $m - 1$ vertices and then repeat all the above arguments. If, in a given instant, we are unable to find two vertices c_i and c_j such that both the arcs (c_i, c_j) , and (c_j, c_i) are present, either we have found a Hamiltonian cycle, hence we have finished or else the resulting graph B' is a tournament, in which case we go back to Step 4.

OUTPUT. Either a Hamiltonian cycle of B or else an answer that such a cycle does not exist in B .

Time analysis. To find perfect matchings in Step 1 needs no more than $O(n^{2.5})$ operations (see [3], [7]). To test if B is strong in Step 2 requires at most $O(n^2)$ operations (see [7]). Finding Hamiltonian cycles of B' , in Step 4, needs $O(n^2)$ operations (see [6]). The time required for Step 5 is proportional to $O(n^2)$, since each edge is counted constant times by Lemma 2.3. Consequently, the operations needed for the whole algorithm are at most $O(n^{2.5})$, as required.

Remark. Algorithm 3.1 can also be used for finding Hamiltonian cycles in semicomplete bipartite digraphs, i.e., bipartite digraphs that contain a spanning subgraph which is a bipartite tournament. On the other hand, this algorithm has been implemented by six students of ENST (Ecole Nationale Supérieure des Télécommunications, Paris) on an IBM-PC computer with satisfactory results. Namely, it seems that once the factor has been found, the algorithm runs with a very high speed. For example, for any bipartite tournament with 150 vertices, the Hamiltonian cycle was constructed from a found factor in less than three seconds.

Algorithm 3.2, given later, finds Hamiltonian paths in bipartite tournaments in at most $O(n^{2.5})$ steps. In order to validate that we need the following two results of [5].

LEMMA 3.2 (Häggkvist and Manoussakis). *Let B be a bipartite tournament that admits a factor. Then B is not strong if and only if there exists a factor F consisting of cycles $C_1, C_2, \dots, C_m$, $m \geq 2$, such that there are arcs from a cycle C_i to a cycle C_j if and only if $i < j$.*

THEOREM 3.3 (Häggkvist and Manoussakis). *Any bipartite tournament B has a Hamiltonian path if and only if one of the following conditions hold:*

- (i) B admits a factor.
- (ii) B admits an “almost factor,” that is, a spanning subgraph that differs from a factor by the direction of exactly one arc.
- (iii) B is almost balanced, that is, $|Y| - |X| = 1$, and moreover has a matching M_{XY} and a matching M_{YX} , each of cardinality $|X|$.

In view of Algorithm 3.2, we must prove the following lemma.

LEMMA 3.4. *Let B be a nonstrong balanced bipartite tournament. We may find a factor F in B , with cycles $C_1, \dots, C_m$, such that m is the minimum possible, with at most $O(n^{2.5})$ operations.*

Proof. Find perfect matchings M_{XY} and M_{YX} and then form a factor F . Now let $C_1, C_2, \dots, C_m$, $m \geq 2$ be the cycles of F . If $m = 2$, we have finished. Assume $m \geq 3$. Now let B' denote the digraph obtained from B as follows: Replace any cycle C_i by a new vertex c_i and then add an arc (c_i, c_j) , $i \neq j$, $i, j = 1, 2, \dots, m$, in B' , if and only if there exists at least one arc from the cycle C_i to the cycle C_j in B . Clearly B' is either a tournament or a semicomplete digraph. Now, by using known algorithms (see [7]), find all strong components, $B'_1, \dots, B'_p$ say, of B' and then apply in each B'_i the algorithm of [6] in order to find a Hamiltonian cycle C'_i corresponding to this component. Next, by using appropriate arcs find the cycles $C_1, \dots, C_p$ in B and form a factor F . Clearly the factor found in this way is the desired. Notice that finding F requires $O(n^{2.5})$ operations, so the proof is complete.

The following $O(n^{2.5})$ algorithm finds Hamiltonian paths in bipartite tournaments.

ALGORITHM 3.2

INPUT: A bipartite tournament B .

We may suppose $|Y| - |X| \leq 1$; otherwise, B has no Hamiltonian path.

Step 1. Find maximum M_{XY} and M_{YX} matchings. If either B is balanced and $|M_{XY}| + |M_{YX}| \leq 2|X| - 2$ or if $|Y| - |X| = 1$ and $|M_{XY}| + |M_{YX}| \leq 2|X| - 1$, then B has no Hamiltonian paths by Theorem 3.3.

Step 2. Take the M_{XY} and M_{YX} matchings found in Step 1. Now let B' be the bipartite tournament obtained from B as follows: If B is balanced, then if $|M_{XY}| + |M_{YX}| = 2|X|$, put $B' = B$; else (by assuming $|M_{YX}| = |X| - 1$) take $V(B') = V(B)$ and $E(B') = E(B) - \{(x, y)\} \cup \{(y, x)\}$, where x and y are the vertices not saturated by M_{YX} . On the other hand, if B is almost balanced and $|M_{XY}| + |M_{YX}| = 2|X|$, then add a new vertex x and the set of arcs $\{(z, x), z \text{ is the vertex not saturated by } M_{YX}\} \cup \{(x, w) \mid w \text{ in } Y - z\}$. Clearly B' has a factor F . Let $C_1, \dots, C_m$ be the cycles of F .

Step 3. Test if B' , found in Step 2, is strong. If it is the case, then find a Hamiltonian cycle in B' by using Algorithm 3.1 and then, by using this result, find a Hamiltonian path in B . Otherwise go to Step 4.

Step 4. Take the factor F of B' , found in Lemma 3.4, with m minimum. Now, find the cycle C_m of F such that there are only incoming arcs in C_m and then let B'' denote the bipartite tournament obtained from B' as follows: Add four new vertices x, y, z, w , and the set of arcs $\{(x, y), (y, z), (z, w), (w, x)\} \cup \{(z, q), (q, x) \mid q \text{ in } X \cap V(C_m)\} \cup \{(w, s), (y, s) \mid s \text{ in } Y \cap V(C_i), i = 1, 2, \dots, m\} \cup \{(z, q), (x, q) \mid q \text{ in } X \cap V(C_i), i \neq m\}$. Clearly B' is strong and has a factor, so it is Hamiltonian by Theorem 3.1. Now, find a Hamiltonian cycle in the bipartite tournament B'' by using Algorithm 3.1. Next, by using this Hamiltonian cycle, order the cycles $C_1, \dots, C_m$ of F , in such a way that no vertex in C_i dominates a vertex in C_j , if and only if $j > i$ in B . Finally, by using appropriate arcs of B and taking care of the modifications given in Step 2, find a Hamiltonian path in B .

OUTPUT: Either a Hamiltonian path of B or else an answer that such a path does not exist in B .

Time analysis. To find maximum matchings in Step 1 needs at most $O(n^{2.5})$ operations by using known algorithms (see [7]). To find B' in Step 2 requires $O(n)$ time.

To find a Hamiltonian cycle of B' needs $O(n^{2.5})$ operations by Algorithm 3.1. To find a factor F of B' with m minimum requires $O(n^2)$ operations by Lemma 3.4. Finally, to find a Hamiltonian cycle in B'' and then a Hamiltonian path in B requires at most $O(n^2)$ operations. It follows that the necessary number of steps for completing the algorithm is at most $O(n^{2.5})$, which is the desired result.

We conclude this paper with the following $O(n^3)$ algorithm for finding cycles through two given vertices in bipartite tournaments.

ALGORITHM 3.3

INPUT: Two vertices x and y in a bipartite tournament B .

We split the algorithm into the following two cases.

First case. The vertices x and y are not in the same bipartition of B .

Assume without loss of generality that x is in X , y is in Y , and that x dominates y . Now it suffices to find a path from y to x .

Second case. Both the vertices x and y are in the same bipartition of B .

Assume without loss of generality that both x and y are in X . Put $Y_1 = \Gamma^-(y)$, $Y_2 = \Gamma^+(y)$, $Y'_1 = \Gamma^-(x)$ and $Y'_2 = \Gamma^+(x)$. Clearly $Y_1 \cup Y_2 = Y = Y'_1 \cup Y'_2$. We distinguish between three subcases (I), (II), and (III).

(I) $Y_1 \cap Y'_2 \neq \emptyset$ and $Y_2 \cap Y'_1 \neq \emptyset$. A cycle of length four is easily found.

(II) $Y'_1 = Y'_2$ and $Y_2 = Y'_2$. It is easy to see that there exists a cycle through x and y if and only if there are two vertex-disjoint paths, say P_1 and P_2 , from Y_2 to Y_1 , each of length 3. Now we delete all the arcs from Y_1 to X and from X to Y_2 and then we find two vertices w and z of X such that $d^+(z) \geq 2$, $d^-(z) \geq 2$, $d^+(w) \geq 2$ and $d^-(w) \geq 2$. If found, then the paths P_1 and P_2 may be easily obtained. Otherwise, if either $d^+(z) = 0$ or $d^-(z) = 0$ holds, then we delete any vertex z of X . Now see that the paths P_1 and P_2 may be found, unless either all arcs from Y_2 to X (respectively, from X to Y_1) are incident to the same vertex of Y_2 (respectively, of Y_1) or else the cardinality of X is exactly three. This completes the proof of this case.

(III) $Y_2 \supset Y'_2$ and $Y'_1 \supset Y_1$. Put $Y_0 = Y_2 \cap Y'_1$. It is easy to see that, if there exists a cycle through x and y , then this cycle must contain a vertex z of Y_0 and both the arcs (y, z) and (z, x) . Now, we choose a vertex z in Y_0 , in all possible cases, and decide if there is a path from Y'_2 to Y_1 in the graph induced by $X \cup Y - \{x, y, z\}$.

OUTPUT: Either a cycle through x and y or else an answer that such a cycle does not exist in B .

Time analysis. To find a path from y to x in the first case requires time proportional to $O(n^2)$. Subcase (I) may be done in linear time. The time needed for (II) is no more than $O(n^2)$. Finally, case (III) may be done in $O(n^3)$ time. Consequently, the whole algorithm requires $O(n^3)$ operations, which is the desired conclusion.

Remark. Algorithm 3.3 can be used for finding cycles through two given vertices in semicomplete bipartite digraphs. This algorithm has been implemented by two students of SUPELEC (Ecole Supérieure d'Electricité) on a VAX/3100 computer. The obtained results are very satisfactory for any bipartite tournament with at most 4500 vertices.

Acknowledgments. We wish to thank the students J. Belt, S. Benhaim, J. F. Bustarret, J. M. Guez, F. Lin, J. M. Scemama, E. Tourangin, and R. Vergnet who kindly accepted to implement Algorithms 3.1 and 3.3.

REFERENCES

- [1] P. CAMION, *Chemins et circuits hamiltoniens des graphes complets*, C.R. Acad. Sci. Paris, Sér, 249 (1959), pp. 2151–2152.
- [2] S. FORTUNE, J. HOPCROFT, AND J. WYLLIE, *The directed subgraph homeomorphism problem*, Theoret. Comput. Sci., 10 (1980), pp. 111–121.
- [3] J. EDMONDS AND R. M. KARP, *Theoretical improvements in algorithmic efficiency for network flow problems*, J. ACM, 19 (1972), pp. 248–264.
- [4] M. GAREY AND D. JOHNSON, *Computers and Intractability—A guide to the Theory of NP-Completeness*, W. H. Freeman, New York, 1979.
- [5] R. HÄGGKVIST AND Y. MANOUSSAKIS, *Cycles and paths in bipartite tournaments with spanning configurations*, Combinatorica, 9 (1989), pp. 51–56.
- [6] Y. MANOUSSAKIS, *A linear time algorithm for finding Hamiltonian cycles in tournaments*, Technical Report No. 488, University of Paris-XI, 1989, Discrete Mathematics, to appear.
- [7] R. E. TARJAN, *Testing graph connectivity*, in Proceedings of the Sixth Annual ACM Symposium on Theory of Computing, Association for Computer Machinery, New York, 1974, pp. 185–193.

CYCLE LENGTHS AND GRAPH ORIENTATIONS*

OLIVER PRETZEL† AND DALE YOUNGS†

Abstract. Those functions defined on the cycles of a finite graph G that can be the flow difference of an orientation of G are characterized. The result is applied to give a proof of Bandelt's Theorem stating that all (Hasse-) diagram orientations of a planar graph G have zero flow difference on all cycles if and only if G has a cycle basis of 4-cycles.

Key words. graph, orientation, circuit, flow difference, diagram, planar graph, cycle basis

AMS(MOS) subject classifications. 05C20, 05C38, 06A10

1. Introduction. In his book [5] Ore posed the problem of characterizing those finite graphs that can be oriented as (Hasse-) diagrams of ordered sets. It has been shown by Nešetřil and Rödl [4] that the corresponding decision problem is NP complete. However, the question of determining, at least theoretically, the possible orientation types of an undirected graph is still important and was stressed as one of the outstanding open problems of orientability in the survey [8].

We begin by introducing informally some of the terminology that is rigorously defined in the next section. Let G be a finite graph. To any cycle C we can assign one of two reference directions ("clockwise and anti-clockwise"). For clarity we will call a cycle with a reference direction a circuit. If C is a circuit, then the "opposite" circuit is denoted by $-C$. When we add circuits, we must respect their reference directions. For instance if C and D are the circuits in the figure below, then $C + D$ contains their common edge e twice and has length 8, while $C - D$ is the "usual" sum, which does not contain e and has length 6. In this paper, as in papers [6], [7] of the first author, additivity is *always taken in the above sense*.

Strictly speaking, the difference $C - D$ in this figure is not equal to the usual cycle sum, but differs from it by a "trivial circuit," one that simply traces the edges of a path to and fro.

An orientation R of the graph G partitions the edges of any circuit C into the forward edges C^+ for which the orientation coincides with the reference direction of C and the backward edges C^- . The flow difference $d_R(C)$, defined as $|C^+| - |C^-|$, was introduced in [7], where it was shown that for two acyclic orientations R and S of G , S can be obtained from R by "pushing down" maximal vertices if and only if for all (simple) circuits C , $d_R(C) = d_S(C)$. In the same paper this result was generalized to orientations admitting monotone cycles. Liu and Rival [2] characterize acyclic orientations related by pushing down maximal vertices in terms of cut-sets of edges and suggest the term "inversions" for such reorientations. We will extend their usage to orientations admitting monotone cycles and call a reorientation S of an oriented graph (G, R) an inversion if $d_R(C) = d_S(C)$ for all circuits.

A monotone circuit in an orientation R is characterized by $d_R(C) = \pm|C|$ and a "bypass" circuit C which has exactly one forward or exactly one backward edge is characterized by $|d_R(C)| = |C| - 2$. As an orientation R is a diagram orientation if it has

* Received by the editors February 15, 1989; accepted for publication (in revised form) February 9, 1990.

† Mathematics Department, Imperial College, London SW7 2BZ, United Kingdom.

neither monotone cycles nor bypasses, the flow difference function determines whether or not an orientation is a diagram orientation.

Our main purpose in this paper is to characterize those functions on circuits that are flow differences of orientations. We prove that certain obvious necessary conditions for a function to be a flow difference are in fact sufficient.

THEOREM 1. *Let G be a finite graph and f a function defined on the circuits of G . Then there is an orientation R of G such that $f(C) = d_R(C)$ for all circuits if and only if f satisfies the following conditions:*

- (i) *Reducibility:* $f(C) = 0$ for all trivial circuits C .
- (ii) *Additivity:* $f(C + D) = f(C) + f(D)$ for all pairs of circuits C and D .
- (iii) *Evenness:* $f(C) \equiv |C| \pmod{2}$ for all circuits C .
- (iv) *Length-boundedness:* $|f(C)| \leq |C|$ for all nontrivial circuits C .

Just as there exist cycle bases for conventional cycles there exist circuit bases for circuits. In fact most natural cycle bases can be made into circuit bases by giving their members arbitrary reference directions. Conditions (i) and (ii) imply that f is determined by its values on a circuit basis, and condition (iii) will hold for all circuits if it holds on a circuit basis. However, condition (iv) must be verified on all (simple) circuits because length is not an additive function on circuits.

The theorem can be immediately extended to characterize flow differences of acyclic and diagram orientations. All that needs to be done is to change the length bound to $|C| - 2$ or $|C| - 4$, respectively.

COROLLARY. *A finite graph G can be oriented as a diagram if and only if there exists a reducible, additive, even function f on the circuits of G such that $|f(C)| \leq |C| - 4$ for all simple circuits.*

Nešetřil and Rödl's NP-completeness result implies that the condition of the Corollary cannot be checked efficiently and that the theorem does not lead to an efficient algorithm for finding the flow difference functions on a graph. Indeed, to apply the corollary we must be able to determine appropriate lower bounds on the lengths of all simple cycles. That clearly requires intimate knowledge of the graph.

Nevertheless, such bounds can be determined for certain types of graphs. As an example, we conclude the paper with a proof of Bandelt's unpublished result characterizing planar graphs for which every diagram orientation is balanced; that is, for any circuit $|C^+| = |C^-|$, or $d_R(C) = 0$. We call a bipartite graph balanced if all its diagram orientations are balanced. Bandelt's original proof of this theorem makes use of powerful topological results, whereas our proof uses only Theorem 1 and the fact that the face cycles of a plane graph contain a cycle basis.

THEOREM 2 (Bandelt). *Let G be a planar graph. Then G is balanced if and only if it has a circuit basis consisting of 4-cycles.*

We remark that in a further paper [10] we shall give a different, more direct, but more intricate, proof of this theorem together with a series of examples to show that it is false for nonplanar graphs. In that paper we also intend to describe in more detail the relationship between circuit bases and conventional cycle bases.

2. Notation and definitions. In all that follows G will be a finite graph. It will be necessary to distinguish oriented concepts from their ordinary counterparts and, in particular, we will need terms to describe paths and cycles traversed in a particular direction. Our terminology is consistent with that of [6], but because our proofs involve contraction we must allow loops and multiple edges.

A walk W in G from x to y is a sequence of alternate vertices and edges $W: x = v_0 e_1 v_1 \cdots e_n v_n = y$, such that the end points of e_i are v_{i-1} and v_i . For links (i.e., nonloops)

e_i the end points determine the direction of traverse; for a loop this must be specified separately. The number n that we denote by $|W|$ is the length of the walk. The inverse walk $-W$ from y to x is obtained by reversing the sequence. If e_i and e_j are the same edge we will write $e_i = e_j$ only if the directions of traverse match. Otherwise, we will write $e_i = -e_j$. If $x = y$, the walk is called a *circuit*. We do not distinguish the circuit $W: v_0e_1 \cdots e_nv_0$ from the circuit $v_1e_2 \cdots e_nv_0e_1v_1$, but we do distinguish W and $-W$.

A walk is *reduced* if it does not traverse an edge to and fro, that is if $e_i \neq -e_j$ for any $i, j = 1, \dots, n$. A walk is *simple* if $e_i \neq \pm e_j$ for any $i, j = 1, \dots, n$, $i \neq j$, and $v_i \neq v_j$ for any $i, j = 0, \dots, n$, $i \neq j$, except possibly $v_0 = v_n$. A circuit in which the edges can be paired off as opposite pairs is called *trivial*. Formally, W is trivial if there exists a bijection φ from $\{1, \dots, n\}$ to itself such that $e_{\varphi(i)} = -e_i$.

Walks W and U can be concatenated if the last vertex x of U is the first vertex of W . We denote the concatenation by $U + W$. Circuits may be concatenated if they have a vertex in common. Any circuit is a sum of reduced circuits and trivial circuits. When representing a reduced circuit as a sum, we delete any trivial circuits that occur. It is in this sense that in Fig. 1, $C - D$ is the outer circuit of length 6. Note that in the figure $C + D$ is reduced but not simple.

A circuit basis is a minimal set of circuits $C_1, \dots, C_m$ such that every reduced circuit can be written as a sum $\sum a_i C_i$. Most natural methods of constructing cycle bases can be adapted to give circuit bases by choosing a reference direction for each cycle. For example, the fundamental circuits of a spanning tree form a circuit basis and for a plane graph the set of face circuits can be made into a circuit basis by removing any one circuit.

If the graph G is given an orientation R , then we can distinguish between *forward* and *backward* edges of the walk W . The edge e_i is a forward edge if R directs it in the direction it is traversed by W . We denote the set of forward edges by W_R^+ and the set of backward edges by W_R^- . If all the edges of W lie in W_R^+ we call W an R -forward walk. If W is a circuit and W or $-W$ is R -forward, W will be called *monotone*. It is well known and obvious that R makes G into a diagram of an ordered set if $|C_R^+| \geq 2$ for all nontrivial circuits C .

For any walk W , the *flow difference* of W is defined by

$$d_R(W) = |W_R^+| - |W_R^-|.$$

The following proposition states the basic properties of the flow difference. The proofs are straightforward and left to the reader.

PROPOSITION 1. *Let G be a finite graph with orientation R . Then*

- (i) *Reducibility: For any trivial circuit C , $d_R(C) = 0$.*
- (ii) *Additivity: For any walks U and W for which the sum is defined $d_R(U + W) = d_R(U) + d_R(W)$.*
- (iii) *Evenness: For all walks W , $d_R(W) \equiv |W| \pmod{2}$.*
- (iv) *Length-boundedness: For all walks W , $|d_R(W)| \leq |W|$.*
- (v) *R is acyclic if and only if for all simple circuits C , $|d_R(C)| \leq |C| - 2$.*
- (vi) *R is a diagram orientation if and only if for all simple circuits C , $|d_R(C)| \leq |C| - 4$.*

Reducibility implies that for any walk W , the flow difference of the trivial circuit $W - W$ is 0. From additivity it follows that $d_R(-W) = -d_R(W)$. Note that since trivial circuits have zero flow difference our convention of ignoring them in adding circuits does not disturb the validity of (ii). Properties (i) and (ii) imply that the flow differences of all circuits are determined by the values on a circuit basis.

Given two orientations S and R defined on the same graph G , we call S an *inversion* of R if $d_S(C) = d_R(C)$ for all circuits C . From Proposition 1 it follows that this condition

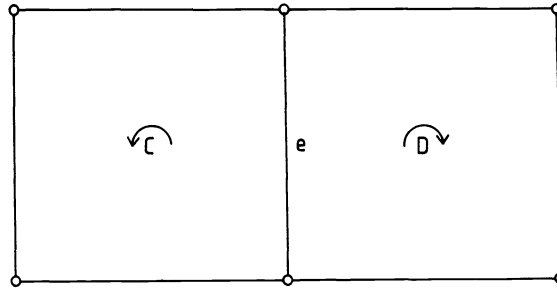


FIG. 1

need only be tested on a circuit basis. The main purpose of [7] was to characterize inversions. So in a sense this paper is complementary to that one. We will use some of the results of [7] and for the reader's convenience we recall the necessary definitions and theorems.

Given a graph G with an orientation R , a vertex is called maximal if it is not the tail of any edge. In general, orientations need not have maximal vertices, but if G is finite and R is acyclic, then such vertices always exist. The operation of *pushing down* a maximal vertex v , introduced by Mosesian [3], reverses the direction of all edges incident with v . It is immediate that the orientation produced by pushing down a maximal vertex of R is an inversion of R .

Theorem 2 of [7] states that any inversion of an acyclic orientation of a finite graph can be obtained by a sequence of such pushdown operations. Theorem 2A of that paper extends the result to general orientations by permitting the additional operation of contracting edges and later reinstating them with their original directions.

Because it is used in the next section, we state here and give a direct proof of a proposition that is a direct corollary of Theorem 2A of [7], and that was first proved for acyclic orientations by Mosesian [3].

PROPOSITION 2. *Let G be a finite connected graph with an orientation R , and let v be a vertex of G not lying on any R -monotone circuit. Then there exists an inversion S of R with v as its only maximal vertex.*

Proof. First contract all edges of G that lie on R -monotone circuits. That produces a graph G' with an acyclic orientation R' . Note that by assumption no edge incident with v has been contracted. Now we apply Mosesian's argument: if there is a maximal vertex $u \neq v$ in G' , push it down. Repeat this until there are no such vertices left. The process will either result in an orientation S' of G' with v as its sole maximal vertex, or it will continue indefinitely.

The latter possibility does not occur because of the following sample fact. If x and y are adjacent, then any pushdown of either x or y reverses the edge xy . Hence between any two pushdowns of x there must be a pushdown of y . Now v is never pushed down. So its neighbours can only be pushed down once, and their neighbours can therefore only be pushed down twice. In general, if the distance from v to u is d , then u can only be pushed down d times.

Having found the orientation S' of G' with v as sole maximal vertex we now reinstate the contracted edges with their original orientations. This gives an orientation S of G . As any cycle of G contracts to a cycle of G' and S' is an inversion of R' , it follows that S and R have the same flow differences on all circuits. Thus S is an inversion of R . Finally, if two vertices are identified u and w in the contraction, then there are forward

walks from u to w and w to u along contracted edges. Hence neither vertex is maximal in S . Thus v is the only maximal vertex of S as required. $\square$

3. The characterization of flow differences. Our aim in this section is to prove Theorem 1. As Proposition 1 above shows that the flow difference of an orientation has the properties stated in the theorem, we need only show that a reducible, additive, even length-bounded function f on circuits is the flow difference of an orientation. The main effort in the proof is concerned with analysing how the flow difference $d_S(W)$ can differ from $d_R(W)$ if S is an inversion of R and W is an arbitrary walk. Recall that for an inversion $d_S(C) = d_R(C)$ for all circuits. For simple walks that are not circuits we show that by applying inversions the flow difference can be increased or decreased by 2, unless there is a cycle whose flow difference prohibits change in the desired direction. To be more precise, the flow difference of W can be increased (or decreased) by 2 unless there is a circuit $C = W - V$ in which W contains all the forward edges (respectively, all the backward edges).

First we consider the case where W consists of a single edge.

LEMMA 1. *Let G be a graph with an orientation R and let e be an edge of G . Then either there exists an inversion S of R that reverses the direction of e , or e lies in an R -monotone circuit.*

Proof. Suppose that e does not lie on a monotone circuit. Then it follows directly from Proposition 2 above that the tail of e can be made into a maximal vertex of G of an inversion S of R . S reverses e . $\square$

Now we extend this result to simple walks of any length. We will state and prove the lemma in a form in which the flow difference is increased by 2. The dual statement follows by symmetry.

LEMMA 2. *Let G be a graph and W a simple walk in G from u to v , where $u \neq v$, and let R be an orientation of G . Then one of the following two statements holds:*

- (a) *there exists an inversion S of R such that $d_S(W) = d_R(W) + 2$.*
- (b) *there exists an R -forward walk V from u to v .*

Proof. We remark that if statement (b) holds, the requirement that for an inversion S of R , $d_S(W - V) = d_R(W - V) = d_R(W) - |V|$ prohibits any increase in the flow difference $d_R(W)$.

The proof is first by induction on the number of edges of G , the statement being trivial for graphs without cycles, and second on the number of backward edges of W , the statement (b) holding trivially if W is a forward walk.

Choose an edge e of W and call its head and tail in the orientation R h and t , respectively. Take e to be in W_R^+ if possible. Let G' be the graph obtained from G by contracting and deleting e . Clearly, R induces an orientation R' of G' and W contracts to a shorter walk W' in G' .

It is possible that the images u' and v' in G' of u and v coincide, but suppose for the moment that they are distinct. Then W' satisfies the conditions of the lemma. Any inversion of R' lifts to an inversion of R in which e has its original orientation. Hence if (a) holds for W' in G' it also holds for W in G . Thus we may assume that (a) does not hold for W' and so by induction hypothesis there exists a forward walk V' from u' to v' in G' .

If $u' = v'$, we let V' be the “empty” walk with no edges from u' to itself. The argument now proceeds identically whether or not $u' = v'$.

There are three ways in which V' may produce an R -forward walk in G .

Case 1(a). V' lifts to a walk V in G .

Case 1(b). V' lifts to two walks V_1 from u to t , and V_2 from h to v .

Case 1(c). V' lifts to two walks V_1 and V_2 that have a vertex in common.

We may exclude these cases, for which statement (b) obviously holds, and are then left with the possibility that V_1 and V_2 are vertex disjoint walks, V_1 from u to h and V_2 from t to v .

The walk W also splits into $W_1 + e + W_2$, but the relation between the end points of W_i and V_j depends on the orientation of e (see Fig. 2). We therefore distinguish two further cases.

Case 2. The edge e lies in W_R^- .

In this case W is a backward walk because e is chosen in W_R^+ if that is not empty. As V_1 and V_2 are forward walks the circuits $V_1 - W_1$ and $V_2 - W_2$ contain only forward edges in R and these edges will not change direction in any inversion of R .

Now by Lemma 1, either there exists an inversion S of R reversing e or there is a forward walk U from h to t . In the former case S preserves the direction of all edges in monotone circuits and thus of all the other edges of W . Hence

$$d_S(W) = d_R(W) + 2,$$

so that (a) holds.

In the latter case the walk $V_1 + U + V_2$ satisfies (b). Thus one of (a) or (b) holds.

Case 3. The edge e lies in W_R^+ .

In this case the walk W_1 is a walk from u to t and W_2 starts at h . If W_1 is a forward walk, then $W_1 + V_2$ satisfies (b). So we may assume that W_1 contains at least one backward edge. Then, since V_1 is a forward walk, the walk $V_1 + W_2$ must contain fewer backward edges than W . Hence by the induction hypothesis, either (b) holds (in which case we are finished) or there is an inversion S of R such that

$$d_S(V_1 + W_2) = d_R(V_1 + W_2) + 2.$$

Since $d_R(V_1)$ cannot be enlarged, it follows that $d_S(W_2) > d_R(W_2)$.

We will show that S reverses exactly one backward edge in each of W_1 and W_2 and also e , and that hence $d_S(W) = d_R(W) + 2$.

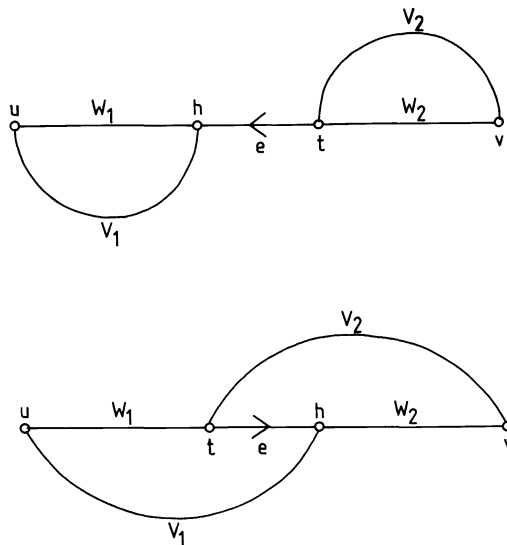


FIG. 2

Consider the circuit $K = e + W_2 - V_2$. Since V_2 lies entirely in K_R^- , we have

$$d_R(W_2) = d_R(K) + |V_2| - 1.$$

Similarly,

$$d_S(W_2) = d_S(K) + d_S(V_2) \pm 1,$$

where the sign is $+$ if S reverses e and $-$ otherwise. Now $d_S(V_2) \leq |V_2|$ and S is an inversion of R so $d_S(K) = d_R(K)$. By hypothesis $d_S(W_2) > d_R(W_2)$. Hence

- (i) $d_S(V_2) = |V_2|$ and V_2 is an S -forward walk;
- (ii) S reverses e ;
- (iii) $d_S(W_2) = d_R(W_2) + 2$.

Now we show that S preserves the orientation of V_1 also. As $d_R(V_1) + d_R(W_2) + 2 = d_S(V_1) + d_S(W_2) = d_S(V_1) + d_R(W_2) + 2$, it follows that $d_S(V_1) = d_R(V_1) = |V_1|$ and V_1 is also an S -forward walk.

It remains to show that $d_S(W_1) = d_R(W_1) + 2$. Consider the circuit $L = W_1 + e - V_1$. We have

$$d_S(W_1) = d_S(L) + |V_1| + 1.$$

Similarly,

$$d_R(W_1) = d_R(L) + |V_1| - 1.$$

Again the flow difference of L under S is the same as that under R . Thus

$$d_S(W_1) = d_R(W_1) + 2.$$

Finally,

$$d_S(W) = d_S(W_1) + d_S(W_2) - 1 = d_R(W_1) + d_R(W_2) + 3 = d_R(W) + 2.$$

Thus (a) holds, and we have shown that in all circumstances one of (a) and (b) holds.

We are now able to prove Theorem 1.

Proof of Theorem 1. As remarked above, it is immediate that the flow difference of an orientation is reducible, additive, even, and length-bounded. We will prove the converse. We assume we are given a reducible, additive, even and length-bounded function f on the circuits of G and will show that there exists an orientation of G having f as its flow difference.

The proof is by induction on the number of edges of G . If G has no cycles, then the statement of the theorem is vacuous. Remove an edge e of G to obtain a graph G' and let R' be an orientation of G' such that $d_{R'}(C) = f(C)$ for all circuits not involving e . If e does not lie on a cycle, then R' can be extended to G by giving e any orientation.

Since every circuit splits into a concatenation of simple circuits and trivial circuits, when we construct the orientation R it is sufficient to verify that $f(C) = d_R(C)$ for simple circuits. For any simple circuit $C = e + U$, define the deficiency of C as follows:

$$\text{def}_{R'}(C) = f(C) - d_{R'}(U).$$

The orientation R' can be extended to an orientation R with $d_R(C) = f(C)$ if and only if $\text{def}_{R'}(C) = \pm 1$.

The first step in the argument is showing that any simple circuits involving e have the same deficiency, so that it will be sufficient to ensure that a single such circuit has deficiency ± 1 .

Let C and D be circuits $e + U$ and $e + V$ involving e and walks U and V avoiding e . Suppose that $\text{def}_{R'}(C) = j$ and $\text{def}_{R'}(D) = k$. Then $C - D = U - V$ is a circuit avoiding e and

$$f(C) - f(D) = f(C - D) = d_{R'}(C - D) = d_{R'}(U) - d_{R'}(V) = f(C) - f(D) + j - k.$$

Hence $j = k$.

The second step in the argument is to find an inversion S' of R' so that for a fixed simple circuit $C = e + U$, $\text{def}_{S'}(C) = \pm 1$.

Note that $d_{R'}(U) \equiv |U| = |C| - 1 \equiv f(C) - 1 \pmod{2}$. Hence $\text{def}(C)$ is necessarily odd. Suppose that $\text{def}_{R'}(C) < -1$ (the argument for the case $\text{def}_{R'}(C) > 1$ is symmetrical). We apply Lemma 2 to U (in the opposite case we would apply it to $-U$).

If statement (b) of the Lemma holds there is a forward walk W in G' with the same end points as U . Let K be the circuit $W - U$, and let L be the circuit $e + W$. Then $L = C + K$ and

$$f(L) = f(C) + f(K) > d_{R'}(U) + 1 + d_{R'}(W) - d_{R'}(U) = 1 + |W| = |L|,$$

contradicting the hypothesis of the theorem.

Therefore, statement (a) of the lemma holds and there exists an inversion S' of R' such that $d_{S'}(U) = d_{R'}(U) + 2$. As S' is an inversion of R' , $d_{S'}(X) = d_{R'}(X)$ for all circuits X not involving e .

We can repeat this argument until $d_{S'}(U) = f(C) - 1$. Then giving e a forward direction extends S' to the required orientation R of G . $\square$

4. Balanced graphs. We call a bipartite graph G balanced if for every diagram orientation R of G , $d_R(C) = 0$ for all circuits C . That means that every circuit has equally many forward and backward edges. It has already been remarked that a diagram orientation is characterized by the condition that

$$d_R(C) \leq |C| - 4,$$

for all nontrivial circuits C . Thus if C is a 4-cycle, then necessarily $d_R(C) = 0$. It follows immediately that if G has a circuit basis consisting of 4-cycles, then G is balanced. As an application of Theorem 1 we will show that the converse holds for planar graphs. This result is due to Bandelt [1]. For a more detailed discussion of balanced graphs we refer the reader to our forthcoming paper [10]. These graphs are used to construct a large variety of graphs that cannot be oriented as diagrams in [9].

THEOREM 3. *Let G be a planar graph. Then G is balanced if and only if it has a circuit basis in which each member has length 4.*

Proof. We will exploit the fact that if the face cycles of a plane graph G are oriented consistently (see Fig. 3) they have sum 0. They can be made into a circuit basis by removing any one of them, but it is easier not to do this and to note that a function f defined on all the face circuits can be extended to a reducible, additive function on all circuits if and only if the sum of the values $f(C)$ over all face circuits is 0.

As remarked just before the theorem, the sufficiency of the condition is immediate. To prove its necessity, we suppose that G does not have a circuit basis of 4-cycles and has been embedded in the plane. We assume that the face cycles have been directed consistently to form face circuits.

We will show that G has an unbalanced diagram orientation. The proof is again an inductive argument. We search for a 4-cycle that is not a face cycle, and if we find one we derive our orientation inductively by considering its interior and exterior separately. The induction start occurs when there is no such 4-cycle.

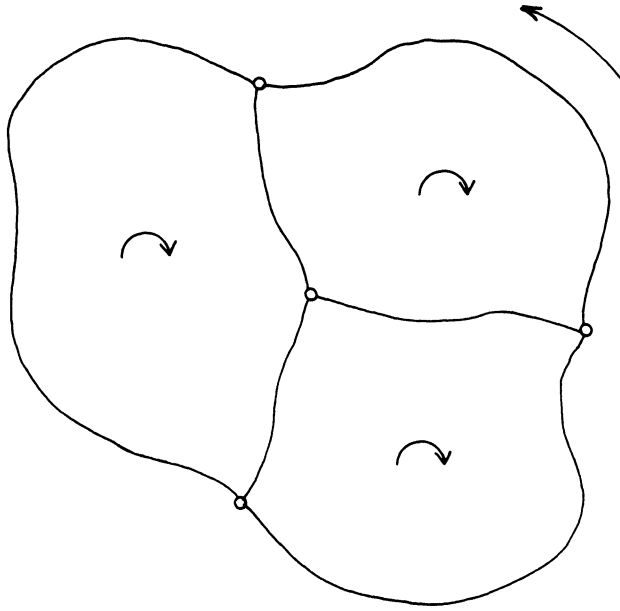


FIG. 3

Induction start. All the 4-cycles of G are face cycles. In that case as G does not have a circuit basis of 4-cycles, at least two face cycles X and Y are not 4-cycles. Define f on the face circuits by $f(X) = -f(Y) = 2$ and $f(K) = 0$ for all other face circuits. Then f can be extended to a reducible, additive, even function on circuits. Now for any simple circuit C , $f(C)$ is (up to sign) the sum of the values $f(K)$ where K ranges over the face circuits on the interior of C . Hence $|f(C)| = 0$ or $|f(C)| = 2$. If C is not a face circuit, or $C = X$ or $C = Y$, then $|C| \geq 6$ by assumption. Thus $|f(C)| \leq |C| - 4$ for all circuits. By Theorem 1, $f(C)$ is the flow difference of a diagram orientation R of G , and since $f(X) \neq 0$, R is unbalanced.

Induction step. G contains a circuit C of length 4 that is not a face circuit or its negative. We define the graphs Int and Ext to be the subgraphs of G consisting of C and its interior C and its exterior, respectively. Int and Ext both have fewer edges than G . Each face cycle of G lies in exactly one of Int or Ext . So every circuit is a sum of circuits of Int and Ext . Hence one of these two graphs, say Int , fails to have a circuit basis of 4-cycles. By induction hypothesis Int has an unbalanced diagram orientation R . Define f on the face cycles of G by

$$\begin{aligned} f(K) &= d_R(K) \text{ if } K \text{ lies in Int;} \\ f(K) &= 0 \text{ otherwise.} \end{aligned}$$

The sum of the face circuits of G lying inside C is $\pm C$ and $d_R(C) = 0$. Hence the sum of $f(K)$ over all the face circuits of G is 0 and it can be extended to a reducible, additive, even function on the circuits of G . It remains to show that this function satisfies the length-boundedness condition.

Let K be a simple circuit of G . Reversing K if necessary, we can ensure that K is the sum of the face circuits on its interior. Thus $K = J + L$ where J is the sum of the face circuits inside K and C , and L is the sum of the face circuits inside K but outside

C . J and L may be sums of disjoint simple circuits, but that does not affect the argument. Now

$$f(K) = f(J) + f(L) = d_R(J).$$

So it will suffice to show that $|K| \geq |J|$.

All the edges of J and L are edges of K except possibly certain edges of C that are common to both J and L . If $L = 0$, there is nothing to show. Otherwise $|L| \geq 4$ and J and L have at most four edges in common, so

$$|K| \geq |J| + |L| - 4 \geq |J|.$$

Therefore $|f(K)| = |d_R(J)| \leq |J| - 4 \leq |K| - 4$. Hence by Theorem 1, $f(K) = d_S(K)$ for a diagram orientation S of G . Restricted to Int , S is an inversion of R that is unbalanced. Hence S is unbalanced and the theorem is proved. $\square$

REFERENCES

- [1] H-J. BANDELT, unpublished manuscript, 1988.
- [2] W. P. LIU AND I. RIVAL, *Inversions, cuts and orientations*, preprint, Department of Computer Science, University of Ottawa, 1988.
- [3] K. M. MOSESIAN, *Certain theorems on strongly basable graphs*, Akad. Nauk. Armian. SSR Dokl., 55 (1972), pp. 83–86. (In Russian.)
- [4] J. NEŠETŘIL AND V. RÖDL, *Complexity of diagrams*, Order, 3 (1987), pp. 321–330.
- [5] O. ORE, *Theory of graphs*, Amer. Math. Soc. Colloq. Pub., 34 (1962).
- [6] O. PRETZEL, *On graphs that can be oriented as diagrams of ordered sets*, Order, 2 (1985), pp. 25–40.
- [7] ———, *On reorienting graphs by pushing down maximal vertices*, Order, 3 (1986), pp. 135–153.
- [8] ———, *Orientations and reorientations of graphs*, Contemp. Math., 57 (1986), pp. 103–125.
- [9] O. PRETZEL AND D. YOUNGS, *Null graphs and non-covering graphs*, preprint, Mathematics Dept., Imperial College, London, 1988.
- [10] ———, *On cycle bases of balanced graphs*, in preparation.

A MINIMAX ARC THEOREM FOR REDUCIBLE FLOW GRAPHS*

VIJAYA RAMACHANDRAN†

Abstract. A conjecture of Frank and Gyárfás is established by proving that the cardinality of a minimum feedback arc set in a reducible flow graph is equal to the cardinality of a maximum collection of arc disjoint cycles.

Key words. graph theory, flow graphs, minimax theorem, network flow

AMS(MOS) subject classifications. 68Q20, 68R05, 68R10

1. Introduction. A feedback arc set (FAS) in a directed graph is a set of arcs whose removal breaks all cycles in the graph. A minimum feedback arc set is an FAS of minimum cardinality.

Let X be a maximum cardinality set of arc-disjoint cycles in a directed graph G and let Y be a minimum FAS for G . Then clearly $|X| \leq |Y|$.

We prove that if $G = (V, A, r)$ is a reducible flow graph (rfg), then $|X| = |Y|$, i.e., the class of rfg's satisfies a minimax arc theorem. It was known before that the class of directed planar graphs satisfies a minimax arc theorem: this is a consequence of the Lucchesi–Younger theorem [Lu Yo], [Lu], [Lo]. It was also known before that the class of rfg's satisfies a minimax vertex theorem [FrGa]; the minimax arc theorem for rfg's is stated as a conjecture in [FrGa].

Our proof is based on the algorithm in [Ra] to find a minimum FAS in an rfg. In the notation of [Ra], we establish that there exists a maximum flow F in $G_{mm}(r)$ that saturates all of the newly added arcs in $G_{mm}(r)$, such that the flow induced by it in each $G_{mm}(i)$ is itself a maximum flow for $G_{mm}(i)$. We then decompose F into a collection of $c = |F|$ paths with the property that a portion of each path traces a path in the DAG of G from the head of a back arc to its tail. This in turn gives us a collection of c arc-disjoint cycles in G , thus establishing the minimax theorem.

Our proof is constructive, and it gives an algorithm to find a maximum collection of arc-disjoint cycles in an rfg in the same time bound as the algorithm in [Ra] to find a minimum FAS in an rfg, i.e., in $O(\min(mn^{5/3}, m^2))$, where m is the number of arcs in the rfg and n is the number of vertices.

2. Graph-theoretic definitions and properties of rfg's. A directed graph $G = (V, A)$ consists of a finite set of vertices (or nodes) V and a set of arc A which is a subset of $V \times V$. An arc $a = (v_1, v_2)$ is an *incoming* arc to v_2 and an *outgoing* arc from v_1 . Vertex v_1 is a *predecessor* of v_2 and v_2 is a *successor* of v_1 ; v_1 is the *tail* of a and v_2 is its *head*. Given a directed graph $G = (V, A)$ and a set of arcs C , we will sometimes use the notation $C \cap G$ to denote the set $C \cap A$. An *arc-weighted* (vertex-weighted) directed graph is a directed graph with a real value on each arc (vertex).

A *directed path* p in G from vertex u to vertex v is a sequence of arcs $a_1, \dots, a_r$ in A such that $a_i = (w_i, w_{i+1})$, $i = 1, \dots, r$ with $w_1 = u$ and $w_{r+1} = v$. The path p passes

* Received by the editors March 7, 1988; accepted for publication October 25, 1989.

† International Computer Science Institute, Berkeley, California and Coordinated Science Laboratory, University of Illinois, Urbana, Illinois. Current address, Department of Computer Sciences, The University of Texas at Austin, Austin, Texas 78712. This work was supported in part by Joint Services Electronics Program grant N00014-84-C-0149 and by the International Computer Science Institute, Berkeley, California.

through each w_i , $i = 1, \dots, r + 1$. A directed path p from u to v is a *cycle* if $u = v$. A directed graph G is *acyclic* if there is no cycle in G . A *rooted directed graph* or a *flow graph* $G = (V, A, r)$ is a directed graph with a distinguished vertex r such that there is a directed path in G from r to every vertex v in $V - \{r\}$.

A *rooted tree* is a flow graph $T = (V, A, r)$ in which every vertex in $V - \{r\}$ has exactly one incoming arc. If (u, v) is the unique incoming arc to v then u is the *parent* of v , and v is a *child* of u . A *leaf* is a vertex in a tree with no outgoing arc. The *height* of a vertex v in a tree is the length of a longest path from v to a leaf. The *height of a tree* is the height of its root. A *forest* is a collection of trees.

Let $G = (V, A, r)$ be a rooted DAG. A vertex u is a descendant of vertex v if either $u = v$ or there is a directed path from v to u in G . The vertex u is a *proper descendant* of v if $u \neq v$ and u is a descendant of v . A vertex v is a (*proper*) *ancestor* of vertex u if u is a (proper) descendant of v .

Let $G = (V, A)$ be an arc-weighted directed graph. A set $F \subseteq A$ is a *feedback arc set* (FAS) for G if $G' = (V, A - F)$ is acyclic. The set F is a *minimum* FAS if the sum of the weights of arcs in F is minimum. Analogous definitions hold for a *feedback vertex set*.

Let $G = (V, A)$ be a directed graph, and let $V' \subseteq V$. The *subgraph of G induced by V'* is the graph $G_s(V') = (V', A')$, where $A' = A \cap V'XV'$. The graph $G - V'$ is the subgraph of G induced on $V - V'$.

A *reducible (flow) graph* (or rfg) is a rooted directed graph for which the rooted depth first search DAG [Ta] is unique. Thus, the arcs in a reducible graph can be partitioned in a unique way into two sets as the DAG or *forward* arcs and the *back* arcs. See [HeUl], [AhUl] for other characterizations of reducible flow graphs.

If the root of an rfg $G = (V, A, r)$ is not the head of a back arc, then G can be decomposed into a collection of arc-disjoint rfg's, such that one rfg in the collection is acyclic, and the root of each remaining rfg is the head of a back arc. Any cycle in G will appear as a cycle in one of the rfg's in the collection and vice versa. Hence for the rest of the paper we will assume, without loss of generality, that the input rfg has a back arc that is incoming to the root.

Let $G = (V, A, r)$ be a reducible graph and let $b = (u, v)$ be a back arc in G . Then b *spans* vertex w (or w is in the *span* of b) if there exists a path from v to u in the DAG of G that passes through w . Given two vertices $u, v \in V$, vertex u *dominates* vertex v if every path from r to v passes through u (note that u dominates itself).

LEMMA 2.1 [HeUl]. *Let $G = (V, A, r)$ be a reducible graph and let $b = (u, v)$ be a back arc in G . Then v dominates every vertex w in the span of b .*

It is well known [AhUl] that the dominator relation can be represented in the form of a tree rooted at r , the root of the flow graph G . This tree is called the *dominator tree* T of G . The descendants of a vertex v in T are the vertices dominated by v in G . A vertex v' is *immediately dominated* by v if it is a child of v in T .

Given a set $V' \subseteq V$ the *dominator forest* $F_{V'}$ for V' represents the dominator relation restricted to the set V' . Let $V_h = \{v \in V \mid v \text{ is the head of a back arc in } G\}$. Since r is the head of a back arc in G it is easy to see that F_{V_h} is a tree; we call it the *head dominator tree* of G and denote it by T_h .

A *flow network* $G = (V, A, s, t, C)$ is an arc-weighted directed graph with vertex set V and arc set A , where s and t are vertices in V called the *source* and *sink*, respectively, and C is the *capacity function* on the arcs which specifies the arc weights, which are always nonnegative. The *maximum flow* problem asks for a *flow* of maximum value from s to t (see [Ev], [FoFu], [PaSt], [Ta] for definition of a flow). A *cut* C separating

s and t is a set of arcs that breaks all paths from s to t . The *capacity* of C is the sum of the capacities of arcs in C . A *minimum cut* separating s and t is a cut of minimum capacity. It is well-known that the value of maximum flow is equal to the value of a minimum cut [FoFu].

Let $G = (V, A, r)$ be an arc-weighted reducible graph and let v be the head of a back arc in G . Let $b_1 = (u_1, v_1), \dots, b_r = (u_r, v_r)$ be the back arcs in G whose heads are dominated by v . The *dominated back arc vertex set* of v is the set $V_v = \{v' \in V \mid v' \text{ lies on a DAG path from } v \text{ to some } u_i, i = 1, \dots, r\}$. It is easy to see that v dominates all vertices in V_v .

DEFINITION 2.1. Let $G = (V, A, r)$ be an arc-weighted reducible graph with non-negative arc weights, and let v be the head of a back arc in G ; hence $G_s(V_v) = (V_v, A_v)$ is the subgraph of G induced by the dominated back arc vertex set of v . The *maximum flow network of G with respect to head v* is a flow network $G_m(v)$ formed by splitting each head h in $G_s(V_v)$ into h and h' (see Fig. 1). All DAG arcs entering or leaving the original head h will enter or leave the newly formed h ; all back arcs entering the original h will enter h' . There will be an arc of infinite capacity from h' to a new vertex t . All other arcs will inherit their capacities from their weights in G . We will interpret v as the source and t as the sink of $G_m(v)$.

Note that the set of arcs in $G_s(V_v)$ is exactly the set of arcs of finite capacity in $G_m(v)$.

DEFINITION 2.2. Let $G = (V, A, r)$ be an arc-weighted reducible graph. We define $G_{mm}(v)$, the *mincost maximum flow network with respect to head v* inductively as follows:

- a. If v dominates no other head in G then $G_{mm}(v) = G_m(v)$.
- b. Let $v_1, \dots, v_r$ be the heads immediately dominated by v in G and let the capacity of minimum cut in $G_{mm}(v_i)$ be c_i , $i = 1, \dots, r$. Then $G_{mm}(v) = (V, A)$ where V is the same as the vertex set for $G_m(v)$ and $A = \{\text{arcs in } G_m(v)\} \cup \{\text{arcs in } G_{mm}(v_i), i = 1, \dots, r\} \cup F_v$, where $F_v = \{f_{vv_i} = (v, v_i) \mid i = 1, \dots, r, \text{ with capacity of } f_{vv_i} \text{ equal to } c_i\}$.

We call F_v the *mincost-arc set for head v* ; if j is a head immediately dominated by head i then f_{ij} is the *mincost arc from head i to head j* .

Figure 2 gives an example of $G_{mm}(v)$.

It is established in [Ra] that the following algorithm determines the cardinality of a minimum FAS in a reducible graph G .

Minimum FAS Algorithm for Reducible Flow Graphs

Input: A reducible graph $G = (V, A, r)$.

Output: The cardinality of a minimum FAS for G .

begin

1. Preprocess G : Label the heads of back arcs in G in postorder. Derive the head dominator tree T_h for G . Introduce a pointer from each vertex i in T_h (except r) to its parent h_i . Let the number of heads be h .
2. For $i = 1, \dots, h$ process head i :
 - a. Find the capacity of minimum cut, c_i , in $G_m(i)$.
 - b. If $i \neq h$ then introduce an arc of weight c_i from h_i to i in G . (Note that G changes during the execution of the algorithm so that $G_m(i)$ is the same as $G_{mm}(i)$ if G were unchanged.)
3. Output c_h as the cardinality of minimum FAS for G .

end.

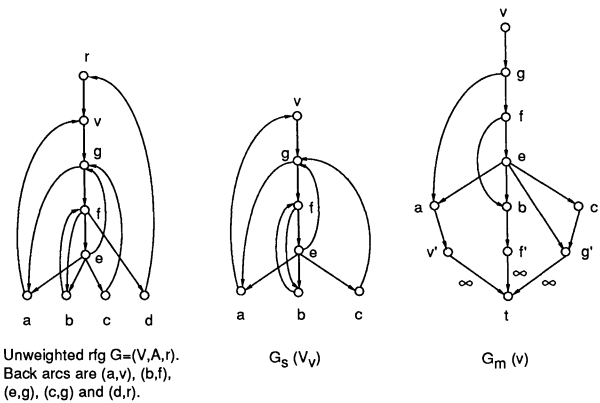


FIG. 1. Constructing $G_m(v)$ from $G = (V, A, r)$.

We will refer to the arc introduced in step 2b from h_i to i as a *newly added arc*. It is shown in [Ra] that this algorithm can be implemented to run in $O(\min(mn^{5/3}, m^2))$ time, where m is the number of arcs in G , and n is the number of vertices.

In the next section we give a specific way of implementing the above algorithm, which gives a maximum flow for $G_{mm}(r)$ with certain properties that allow us to construct a collection of c_h arc disjoint cycles in G . This in turn establishes the minimax theorem, and allows us to construct a maximum collection of arc-disjoint cycles in $O(\min(m^2, mn^{5/3}))$ time.

3. The Minimax Theorem. We describe our implementation of the minimum FAS algorithm. In step 2a of the algorithm, let the children of head i in T_h be $i_1, \dots, i_r$. Let $F_{i_1}, \dots, F_{i_r}$ be the maximum flow on $G_m(i_1), \dots, G_m(i_r)$, respectively, found by the algorithm. During the i th execution of step 2, we extend these flows to a feasible flow F for $G_m(i)$ by saturating the newly added arcs from i to $i_j, j = 1, \dots, r$. We then find a maximum flow F_i for $G_m(i)$ by successively finding a special type of augmenting path

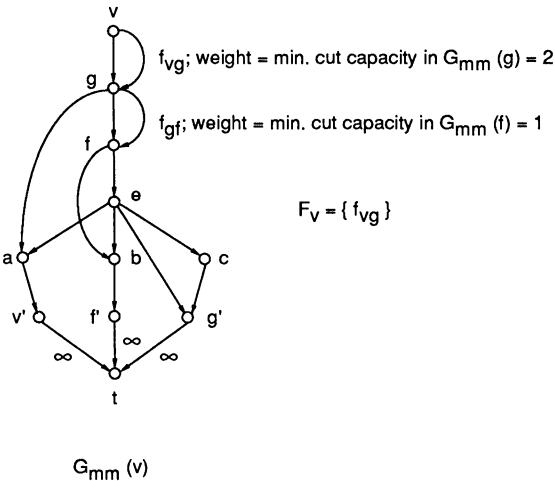


FIG. 2. The mincost maximum flow network with respect to head v for the graph $G = (V, A, r)$ of Fig. 1.

from i to t , and pushing flow along it until no augmenting path can be found (see [Ev], [FoFu], [PaSt], [Ta] for definition and properties of augmenting paths). At this point a maximum flow and a minimum cut for $G_m(i)$ are obtained [FoFu].

Our special augmenting path will be constructed as a shortest path in the residual graph, with newly added arcs having weight n , and the original arcs from G and the arcs into t having weight 1 (arcs will have the same weight, whether they appear in the forward or reverse direction in the residual graph). Here, n is the number of vertices in G .

We will establish later that every augmenting path constructed by this method contains only arcs of weight 1. Thus we can find each such path by a breadth first search in the corresponding residual graph with the newly added arcs deleted. Thus, each augmenting path can be found in $O(m)$ time, where m is the number of arcs in G . Hence, by the analysis in [Ra], the time complexity of this algorithm is $O(m^2)$. By using Dinic's algorithm for unit-capacity networks (see, e.g., [Ta]) on residual graphs with newly added arcs deleted, we can obtain a time bound for the algorithm as $O(mn^{5/3})$. Hence, this implementation of the minimum FAS algorithm has time complexity $O(\min(m^2, mn^{5/3}))$, which is the same complexity as the one in [Ra].

LEMMA 3.1. *Let $\mathbf{F}$ be a feasible flow for $G_{mm}(v)$, and let $v', v'_1, \dots, v'_r$ be the vertices adjacent to sink t in $G_{mm}(v)$. Let the sum of the flows on incoming arcs to v' be f , and the sum of the flows on incoming arcs to each v'_i be f_i . Then the flow $\mathbf{F}$ can be decomposed into $|F|$ paths from s to t , of which f_i pass through v'_i and f pass through v' , such that the number of paths through any arc is exactly equal to the value of the flow on that arc.*

Proof. The proof is obvious. $\square$

LEMMA 3.2. *Any augmenting path constructed by the algorithm does not push flow backward on any of the arcs incident on the v' .*

Proof. There would be a shorter path that avoids using this arc. $\square$

LEMMA 3.3. *Let $e_1, \dots, e_r$ be the arcs entering v' in $G_{mm}(v)$, and let $e_1, \dots, e_j, j \leq r$ be the arcs that are saturated by the maximum flow $\mathbf{F}$, found by the algorithm for $G_{mm}(v)$. Then, for every ancestor u of v in T_h , the sequence of flows constructed by the algorithm for $G_{mm}(u)$ leaves unchanged the saturated arcs incident on v' as $e_1, \dots, e_j$.*

Proof. By Lemma 3.2 there is no flow pushed backward along any of the $e_i, i = 1, \dots, j$. Hence it only remains to show that there is no flow pushed forward on any of the remaining e_i .

Assume for contradiction that $\mathbf{F}$ is a feasible flow constructed by the algorithm for $G_{mm}(u)$, where u dominates v , and $\mathbf{F}$ saturates some arc, say e_{j+1} , in addition to the previously saturated arcs $e_1, \dots, e_j$. Now consider the decomposition of $\mathbf{F}$ into $|F|$ paths as in Lemma 3.1. There are $j+1$ paths of flow from u to v' . Since v dominates v' , all of these paths pass through v . But this implies that there is a feasible flow in $G_{mm}(v)$ with value $j+1$, contradicting the fact that the maximum flow value in $G_{mm}(v)$ is j . Hence there can be no flow pushed forward on any of the remaining e_i . $\square$

LEMMA 3.4. *The flow constructed by the algorithm after each augmentation saturates all of the newly added arcs in the current network.*

Proof. Suppose not. Then let $\mathbf{F}$ be a flow constructed for $G_{mm}(v)$, which is the first flow constructed by the algorithm in which some newly added arc, say (u, u_i) is left unsaturated. Let capacity of (u, u_i) be c . Let $\mathbf{F}^-$ be the flow constructed by the algorithm just before $\mathbf{F}$ and let p be the augmenting path that formed $\mathbf{F}$ from $\mathbf{F}^-$. Thus p traverses the newly added arc (u, u_i) in the reverse direction.

Now consider $G_{mm}(u_i)$. The maximum flow value in $G_{mm}(u_i)$ is c , and by Lemma 3.3, there are c units of flow incoming to u'_i in $\mathbf{F}$. By Lemma 3.1, this flow can be

decomposed into c paths of flow from v to u'_i . By the dominance relation, each of these paths passes through u to u_i and then to u'_i . Hence $\mathbf{F}$ induces c sub-paths of flow from u to u_i . We can redirect this flow to saturate arc (u, u_i) and thus obtain a new feasible flow $\mathbf{F}'$, with $|\mathbf{F}'| = |\mathbf{F}|$, which saturates (u, u_i) . But this implies that there is an augmenting path p' which uses one less newly added arc than p , contradicting the fact that p is a shortest augmenting path in the residual graph with weight n on newly added arcs and weight 1 on the remaining arcs. $\square$

LEMMA 3.5. *Let $\mathbf{F}$ be a flow constructed by the algorithm for $G_{mm}(v)$. Then $\mathbf{F}$ can be decomposed into $|\mathbf{F}|$ paths as in Lemma 3.1, with the additional property that each path to v' contains only the original arcs in G , and each path to v'_i consists of a sequence of newly added arcs up to v_i , followed by a sequence of original arcs of G from v_i up to v'_i .*

Proof. The proof is obtained by induction on the height α of v in the head dominator tree T_h .

Base. When $\alpha = 0$ there are no newly added arcs in $G_{mm}(v)$ and the result follows from Lemma 3.1.

Induction step. Assume that the result is true up to height $\alpha - 1$ and let the height of v in T_h be α . Let $\mathbf{F}$ be a feasible flow constructed by the algorithm for $G_{mm}(v)$. By Lemma 3.4 this flow saturates all of the newly added arcs in $G_{mm}(v)$.

Let the children of v in T_h be $v_1, \dots, v_r$. Let $\mathbf{F}_i$, $i = 1, \dots, r$ be the flow induced in $G_{mm}(v_i)$ by $\mathbf{F}$. By Lemma 3.3, each $\mathbf{F}_i$ is a maximum flow for $G_{mm}(v_i)$. Hence we obtain $\sum_{i=1}^r |\mathbf{F}_i|$ units of flow from v to those u' that are dominated by the v_i (i.e., all of the primed vertices except v'), by including the flow on the saturated newly added arcs (v, v_i) , $i = 1, \dots, r$. By the induction hypothesis, each such path can be decomposed as required by the lemma. The remaining paths, which are the paths to v' , can only contain original arcs from G . $\square$

THEOREM 3.1. *Let c be the cardinality of a minimum feedback arc set in an rfg $G = (V, A, r)$. Then there exist c arc-disjoint cycles in G .*

Proof. Let $\mathbf{F}$ be the maximum flow constructed by the algorithm for $G_{mm}(r)$. Hence $|\mathbf{F}| = c$ by the result in [Ra]. By Lemma 3.5, this flow can be decomposed into c paths, such that each path entering u' consists of a sequence of newly added arcs to u , followed by a sequence of original arcs from u to u' . Thus, embedded in $\mathbf{F}$ are $|\mathbf{F}|$ paths from the heads of back arcs to their primed images, each containing only arcs from G . These paths are arc disjoint, since the capacity of the original arcs from G in $G_{mm}(r)$ is 1. Thus all of these paths are arc-disjoint, and each traces out a cycle in G . Hence there are c arc-disjoint cycles in G . $\square$

By Lemma 3.4, each augmenting path constructed by the algorithm contains only arcs of weight 1. Hence, as mentioned at the beginning of the section, the algorithm finds a maximum flow $\mathbf{F}$ for $G_{mm}(r)$ in $O(\min(m^2, mn^{5/3}))$ time that saturates all newly added arcs. We can now extract the maximum collection of arc disjoint cycles described in Theorem 3.1 by processing the heads of back arcs in postorder on T_h , and extracting the cycles from head u to u' in $G_{mm}(u)$ by pushing flow backward from u' to u . This only takes additional linear time.

REFERENCES

- [AhUl] A. V. AHO AND J. D. ULLMAN, *Principles of Compiler Design*, Addison Wesley, New York, 1977.
- [Ev] S. EVEN, *Graph Algorithms*, Computer Science Press, Potomac, MD, 1979.

- [FoFu] L. R. FORD, JR. AND D. R. FULKERSON, *Flows in Networks*, Princeton University Press, Princeton, NJ, 1962.
- [FrGa] A. FRANK AND A. GYARFAS, *Directed graphs and computer programs*, in *Problemes Combinatoires et Theorie des Graphes*, Colloque Internationaux C.N.R.S., 260 (1976), pp. 157–158.
- [HeUl] M. S. HECHT AND J. D. ULLMAN, *Characterization of reducible flow graphs*, JACM, 21 (1974), pp. 167–175.
- [Lo] L. LOVASZ, *On two minimax theorems in graph*, J. Combin. Theory (B), 21 (1976), pp. 96–103.
- [Lu] C. L. LUCCHESI, *A minimax equality for directed graphs*, Ph.D. thesis, University of Waterloo, Waterloo, Ontario, Canada, 1976.
- [LuYo] C. L. LUCCHESI AND D. H. YOUNGER, *A minimax theorem for directed graphs*, J. London Math. Soc., 17 (1978), pp. 369–374.
- [PaSt] C. H. PAPADIMITRIOU AND K. STEIGLITZ, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, NJ, 1982.
- [Ra] V. RAMACHANDRAN, *Finding a minimum feedback arc set in reducible flow graphs*, J. Algorithms, 9 (1988), pp. 299–313.
- [Ta] R. E. TARJAN, *Data Structures and Network Algorithms*, CBMS-NSF Regional Conference Series 44, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1983.

GENERATING PERMUTATIONS WITH k -DIFFERENCES*

CARLA D. SAVAGE†

Abstract. Given (n, k) with $n \geq k \geq 2$ and $k \neq 3$, it is shown how to generate all permutations of n objects (each exactly once) so that successive permutations differ in exactly k positions, as do the first and last permutations. This solution generalizes known results for the specific cases where $k = 2$ and $k = n$. It is shown that it is possible, when $k = 3$, to generate all even (odd) permutations of n objects so that successive permutations differ in exactly three positions.

Key words. permutations, Gray codes, combinatorial algorithms, Cayley graphs, Hamilton cycles

AMS(MOS) subject classifications. 05A15, 05C45

1. Introduction. The problem of generating permutations of n distinct objects is of fundamental importance in both computer science and combinatorics. Many practical problems require for their solution a sampling of random permutations or, worse, a search through all $n!$ permutations. For such a search to be possible, even for moderate size n , permutation generation methods must be extremely efficient. A survey of efficient techniques is presented in [24].

A problem that was initially motivated by practical concerns is that of generating permutations by *transpositions*, that is, listing all $n!$ permutations, each exactly once, in such a way that successive permutations on the list differ only by the exchange of two elements. This was shown to be possible in several papers, including [26], [1], [2], and [9], which are described in [24]. In fact, it is possible even if the two elements exchanged are required to be in adjacent positions [13], [25]. It is interesting to note that in the Wells–Boothroyd–Heap algorithms, the last permutation differs from the first permutation by a transposition only when n is odd, so the scheme is not “cyclic.” The Johnson–Trotter scheme for generating permutations by adjacent transpositions is cyclic and is shown in Fig. 1 for $n = 1, 2, 3, 4$.

A new twist on the problem is to generate all permutations by transpositions so that every other transposition is of the elements in the first two positions. See Fig. 2 for an example taken from [22]. Recently, it has been shown that this is possible even when the transpositions are required to be of adjacent positions [23].

A contrary approach to the problem is to require that permutations be listed so that each one differs from its predecessor in *every* position [20], that is, by a “derangement.” The existence of such a list when $n \neq 3$ was established in [17] using Jackson’s theorem [12] and a constructive solution was presented in [7]. A simpler construction, ascribed to Yarbrough, is discussed in [21] and is shown in Fig. 3 for $n = 4$.

To generalize the problems of listing permutations to differ, on the one hand, in every position and, on the other hand, in only two positions, we consider the following. Given n and k satisfying $n \geq k \geq 2$, is it possible to list all permutations so that successive permutations differ in exactly k positions? In this paper, we show by construction that the answer is yes, unless $k = 3$. (We have learned that Putnam, working independently, has recently solved the same problem [19].) When $k = 3$, two permutations that differ in exactly three positions, in fact, differ by a product of two 2-cycles, so that successive permutations must have the same parity. Our motivation for studying this problem of

* Received by the editors September 5, 1989; accepted for publication (in revised form) March 6, 1990. This work was supported in part by National Science Foundation grant CCR8906500.

† Department of Computer Science, North Carolina State University, Raleigh, North Carolina 27695.

$n = 2$	$n = 4$
1 2	1 2 3 4
2 1	1 2 4 3
	1 4 2 3
	4 1 2 3
	4 1 3 2
	1 4 3 2
	1 3 4 2
	1 3 2 4
$n = 3$	3 1 2 4
	3 1 4 2
1 2 3	3 4 1 2
1 3 2	4 3 1 2
3 1 2	4 3 2 1
3 2 1	3 4 2 1
2 3 1	3 2 4 1
2 1 3	3 2 1 4
	2 3 1 4
	2 3 4 1
	2 4 3 1
	4 2 3 1
	4 2 1 3
	2 4 1 3
	2 1 4 3
	2 1 3 4

FIG. 1. Johnson–Trotter scheme for generating permutations by adjacent transpositions.

permutation generation with k -differences is clearly not for efficient permutation generation, which would be better handled by techniques from [24]. Rather, we found the problem mathematically interesting, particularly in view of its combinatorial connections which are discussed below.

Recently, permutation generation has been studied from a combinatorial perspective, due in part to its ties with some interesting graph theoretic problems. In this context, it is of interest to specify some set S of “changes” and to then ask if it is possible to list all

$n = 2$	$n = 4$
1 2	1 2 3 4
2 1	2 1 3 4
	2 1 4 3
	1 2 4 3
	1 4 2 3
	4 1 2 3
	4 2 1 3
	2 4 1 3
$n = 3$	2 4 3 1
	4 2 3 1
1 2 3	4 3 2 1
2 1 3	3 4 2 1
2 3 1	3 2 4 1
3 2 1	2 3 4 1
3 1 2	2 3 1 4
1 3 2	3 2 1 4
	3 1 2 4
	1 3 2 4
	1 3 4 2
	3 1 4 2
	3 4 1 2
	4 3 1 2
	4 1 3 2
	1 4 3 2

FIG. 2. Generating permutations by transpositions so that alternate transpositions swap the first two positions [21].

1 2 3 4	3 2 1 4
4 1 2 3	4 3 2 1
2 3 4 1	2 1 4 3
3 4 1 2	1 4 3 2
1 3 2 4	2 3 1 4
4 1 3 2	4 2 3 1
3 2 4 1	3 1 4 2
2 4 1 3	1 4 2 3
3 1 2 4	2 1 3 4
4 3 1 2	4 2 1 3
1 2 4 3	1 3 4 2
2 4 3 1	3 4 2 1

FIG. 3. Generating permutations by derangements, due to Yarbrough [20].

permutations so that each permutation differs from its predecessor on the list only by a change from the set S . More precisely, for fixed n , let $G[S]$ be the directed graph whose vertices are the $n!$ permutations of $1 \cdots n$. For permutations p and q , there will be an edge in $G[S]$ from p to q if and only if q can be obtained from p by a change specified in S . A question of interest, then, is whether or not $G[S]$ has a Hamiltonian path or cycle. For example, by results cited above, $G[S]$ is Hamiltonian when S allows only adjacent transpositions [13], [25] or when S allows only derangements and $n \neq 3$ [7], [17]. As another example, it is shown in [15] that for a set S of transpositions (of specific positions), $G[S]$ is Hamiltonian if the elements of S form a basis for the symmetric group.

A well-known open problem in graph theory, due to Lovász [16], is to determine whether every undirected, connected, vertex transitive graph has a Hamilton path. In the case of permutation graphs, there are many interesting sets S for which $G[S]$ is connected, vertex transitive, and symmetric (and can therefore be regarded as undirected). In particular, for $n \geq k \geq 2$, if S requires only that permutations differ in exactly k positions, the graph $G[S]$ is symmetric, vertex transitive, and, unless $k = 3$, connected. Thus, with our result in this paper that $G[S]$ is Hamiltonian, we exhibit a new class of graphs which will not yield a counterexample to the conjecture of Lovász.

In fact, $G[S]$ is a Cayley graph and it is an open question whether even every (undirected) Cayley graph is Hamiltonian. No general results on Cayley graphs apply to show that the k -differences graph is Hamiltonian. A related problem due to Wilf [28] is whether $G[S]$ is Hamiltonian when S allows only (1) a swap of the first two positions (i.e., $123 \cdots n$ to $213 \cdots n$) or (2) a cyclic shift one position left or right of all positions (i.e., $123 \cdots n$ to $23 \cdots n1$ or to $n123 \cdots n - 1$). In work to appear in his Ph.D. thesis, Compton has shown that this graph is Hamiltonian [5]. In the case of the alternating group, with generating set the 3-cycles $S = \{(1\ 2\ n), (1\ 3\ n), \dots, (1\ n-1\ n)\}$, Gould and Roth have shown that the undirected Cayley graph is Hamiltonian [11].

Finally, we note that although the existence of a listing of permutations with k -differences for $k = n$ is guaranteed by Jackson's theorem [12] (as shown in [17] and [27]), the theorem does not apply to establish existence when $k < n - 1$. Jackson's result is that every 2-connected, d -regular graph of at most $3d$ vertices is Hamiltonian. In the case $k = n$, the degree of each vertex in $G[S]$ is the number of derangements of $1 \cdots n$, which is (see, e.g., [8, pp. 9–10])

$$n!(1 - 1/1! + 1/2! - 1/3! + \cdots + (-1)^n/n!).$$

For $n \geq 3$ this is at least

$$n!(1 - 1/1! + 1/2! - 1/3!) = n!/3.$$

So, at least the vertex degree is large enough for Jackson's theorem to apply. However, this is not the case for arbitrary k since the degree will be

$$(n!/(n-k!))(1 - 1/1! + 1/2! - 1/3! + \cdots + (-1)^k/k!)$$

and for $n - 2 \geq k \geq 2$ this is at most

$$(n!/2)(1 - 1/1! + 1/2!) = n!/4 < n!/3.$$

2. Listing 2-element subsets. In [7], permutations of $1 \cdots n$ were listed by derangements using a recursive listing of the permutations of $1 \cdots n - 2$ by derangements, together with a listing of the 2-element subsets of $[n]$. Our strategy is to generalize this construction and several new ideas are required.

Our technique for listing permutations of n objects with k -differences combines a recursive listing of the permutations of $n - 2$ objects with $k - 2$ -differences together with a particular way of listing the 2-sets of $[n]$. As is common, we will use "2-sets" to denote "2-element subsets."

For a subset S of $[n]$, let $(1 \cdots n) \setminus S$ denote the unique permutation of the elements of $[n] \setminus S$ in which the elements appear in sorted order. For example,

$$(1 \cdots 5) \setminus \{2\} = 1345$$

$$(1 \cdots 6) \setminus \{3, 5\} = 1246.$$

For $x, y, z \in [n]$, note that although the sets $[n] \setminus \{x, y\}$ and $[n] \setminus \{x, z\}$ differ in only one element, the permutations $(1 \cdots n) \setminus \{x, y\}$ and $(1 \cdots n) \setminus \{x, z\}$ could differ in many positions. For example,

$$(1 \cdots 7) \setminus \{1, 7\} = 23456$$

whereas

$$(1 \cdots 7) \setminus \{6, 7\} = 12345.$$

For the construction of this paper, we would like to list the 2-sets of $[n]$ in such a way that the following two conditions hold.

Condition 1. If $\{u, v\}$ follows $\{x, y\}$ on the list of 2-sets, then the permutations $(1 \cdots n) \setminus \{u, v\}$ and $(1 \cdots n) \setminus \{x, y\}$ differ in exactly one position. (In particular, this means that successive 2-sets differ in exactly one element.)

Condition 2. The first 2-set is $\{1, 2\}$ and the last is $\{3, 4\}$.

To satisfy these properties, we will construct a certain graph whose vertices are the 2-sets of $[n]$ and in which adjacent vertices satisfy Condition 1. We then observe that this graph has a Hamilton path from $\{1, 2\}$ to $\{3, 4\}$, thereby giving a listing of 2-sets of $[n]$ satisfying conditions 1 and 2.

For $n \geq 3$ let G_n be the undirected graph whose vertices are the 2-element subsets of $[n]$ and whose edges are defined as follows. (See Fig. 4.)

For $x < y$, $\{x, y\}$ is adjacent to

1. $\{x, y + 1\}$, if $1 \leq x < y \leq n - 1$ (vertical edges);
2. $\{x + 1, y\}$, if $1 \leq x \leq y - 2 \leq n - 2$ (horizontal edges);
3. $\{x + 1, y + 1\}$, if $1 \leq x = y - 1 \leq n - 2$ (diagonal edges).

LEMMA 1. *If $\{x, y\}$ and $\{x, z\}$ are adjacent vertices of G_n , then the permutations $(1 \cdots n) \setminus \{x, y\}$ and $(1 \cdots n) \setminus \{x, z\}$ differ only in one position.*

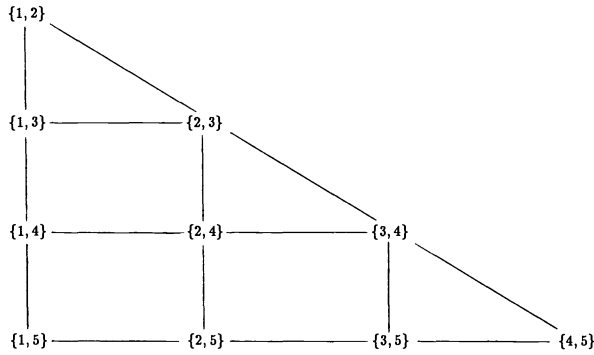


FIG. 4. The graph G_5 .

Proof. Without loss of generality, assume $y < z$. Then by definition of G_n , either

1. $z = y + 1$ (horizontal or vertical edge) or
2. $z = x + 1 = y + 2$ (diagonal edge).

In case 1, if $x < y$,

$$(1 \cdots n) \setminus \{x, y\} = 1 \cdots (x-1)(x+1) \cdots (y-1)z(y+2) \cdots n,$$

$$(1 \cdots n) \setminus \{x, z\} = 1 \cdots (x-1)(x+1) \cdots (y-1)y(y+2) \cdots n.$$

Similarly, if $x > y$, the sequences differ in only one position. In case 2,

$$(1 \cdots n) \setminus \{x, y\} = 1 \cdots (y-1)z(y+3) \cdots n,$$

$$(1 \cdots n) \setminus \{x, z\} = 1 \cdots (y-1)y(y+3) \cdots n. \quad \square$$

LEMMA 2. For $n \geq 4$ there is always a Hamilton path in G_n that starts at $\{1, 2\}$ and ends at $\{3, 4\}$.

Proof. Construction of such a path depends on the parity of n . See Fig. 5 for details. $\square$

There are many algorithms in the literature for generating the k -sets of an n -set. The revolving door algorithm in [18] generates all k -sets so that successive sets differ in only one element. If we used this algorithm to generate 2-sets, neither property 1 nor 2 would be satisfied.

Other algorithms [4], [3], [10] can in some cases generate k -sets with the *minimal change property*, that is, so that successive sets differ in exactly one element, and this element has either increased or decreased by 1. For $k = 2$, this would amount to using only horizontal and vertical edges in G_n . However, it is known that there is a listing of k -sets with the minimal change property if and only if either n is even and k is odd or if k is one of 0, 1, n , $n - 1$.

More closely related to our listing is the *strong minimal change property* (SMC) [6]. That is, k -sets are generated so that if a k -set is represented as a sorted sequence, successive k -sets differ in only one position. This is what property 1 requires: a listing of the $n - 2$ -sets of $[n]$, with the SMC property, starting with $34 \cdots n$ and ending at $1256 \cdots n$. We cannot directly apply the algorithm of [6] only because the algorithm of that paper starts with $12 \cdots n - 2$ and ends with $34 \cdots n$. Note that although the sets $\{1, 2\}$ and $\{1, 4\}$ are adjacent in SMC order, their complements in $\{1, \dots, 4\}$ are not. Thus, a listing of 2-sets in SMC order does not necessarily give a corresponding listing of the complementary $k - 2$ -sets with the same property.

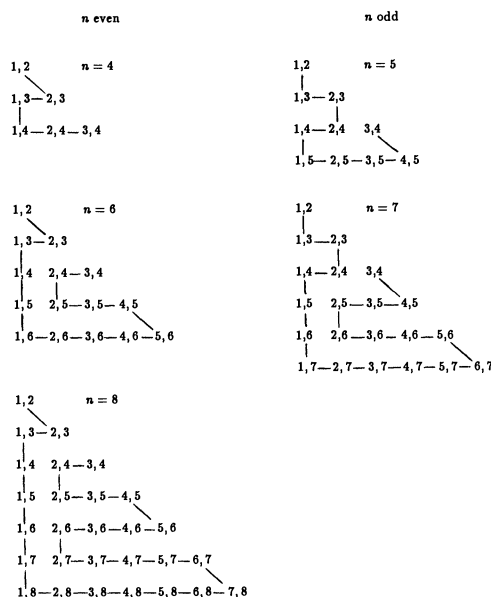


FIG. 5. Construction of Hamilton path in G_n from $\{1, 2\}$ to $\{3, 4\}$.

3. The basis cases. As stated in § 2, our technique for listing permutations of n objects with k -differences requires that it be possible to recursively list permutations of $n - 2$ objects with $(k - 2)$ -differences. For even values of k , the case $k = 2$ will serve as the basis of the recursion. For the case $k = 2$, any method for generating permutations by transpositions will suffice as long as the first permutation differs from the last in only two positions. This property will be satisfied by the Johnson–Trotter scheme for listing all permutations by adjacent transpositions, which works as follows. For $n = 2$, the listing is 12, 21. For $n > 2$, make n copies of each permutation on the list for $n - 1$. In each group of n identical permutations, insert “ n ” into each possible position going from position n down to position 1 on odd numbered groups and from position 1 up to position n on even numbered groups. The lists for $n = 2, 3, 4$ are shown in Fig. 1.

For odd k , the case $k = 3$ cannot serve as the basis since it is not possible to generate all permutations with 3-differences. Two permutations which differ in three locations differ by a 3-cycle, which is an even permutation. Thus, from any given permutation, π , any permutation obtained from π by applying a sequence of 3-cycles must have the same parity as π .

However, we show that it is possible to list all *even* (*odd*) permutations of n items with 3-differences. For our recursive construction in § 4 this is sufficient. We show there how to construct a list of all permutations of n objects with 5-differences from listings of all even permutations of $n - 2$ elements with 3-differences.

We describe now a way to list all even permutations of n items (for $n \geq 3$) with 3-differences. In fact, we will require that the list start with the permutation $1234 \cdots n$ and end with $3124 \cdots n$, so the first and last elements will also differ in three positions.

If $n = 3$, the list is 123, 231, 312. For $n > 3$ and for $i = 1 \cdots n$, let group i consist of the set of even permutations of $1 \cdots n$ which have i in their first position. The list of even permutations of $1 \cdots n$ will consist of listings of the groups 1, $\cdots$, n in the following order:

group 1, group n , group $n - 1$, $\cdots$, group 4, group 2, group 3.

Within each group i , we list recursively the even permutations of $(1 \cdots n) \setminus \{i\}$ with 3-differences, and prefix each with “ i .” It remains to specify the first element of each group i . The first element of group 1 is $12 \cdots n$. The first element of group 3 is $32415 \cdots n$, so the list will always end with $31245 \cdots n$. For groups $n, n-1, \cdots, 4$, the first element is obtained from the last element of the preceding group by a cyclic permutation of the following three elements: 2, i , and the element in position 1. Precisely, the element in position 1 moves to the location containing element 2, element 2 moves to the location containing element i , and element i moves to position 1. Element 2 is acting as a “place holder” for the element in position 1. (See Fig. 6.) Finally, the first element of group 2 is $23145 \cdots n$ if n is divisible by 3 and $24315 \cdots n$ otherwise. (See Fig. 6.)

This clearly gives all even permutations of $1 \cdots n$ so that it remains to verify that successive permutations differ in exactly 3 positions. This is evident within each group. Between groups 1, $n, \cdots, 4$ this follows since the first permutation of each group was defined by permuting 3 positions in the last permutation of the preceding group. To check between groups 2 and 3, note that group 2 ends in either $24315 \cdots n$ or $21435 \cdots n$ and that each of these differs in 3 positions from $32415 \cdots n$. Finally, to check between groups 4 and 2, note that group 4 always ends in $4(213)^n \cdots n$ where $(213)^n$ denotes a right cyclic shift n times of 213.

In summary, we have described a procedure which, given distinct symbols $x_1 \cdots x_n$, can be used to generate all even permutations of $x_1 \cdots x_n$ with 3-differences. The list begins with the permutation $x_1 \cdots x_n$ and ends with the permutation $x_3 x_1 x_2 x_4 \cdots x_n$. Note that if, in every permutation on the list, the elements x_1 and x_2 were exchanged, the result would be a listing of all odd permutations of $1 \cdots n$ with 3-differences, beginning with the permutation $x_2 x_1 x_3 x_4 \cdots x_n$ and ending with $x_3 x_2 x_1 x_4 \cdots x_n$.

4. The recursive construction. In this section we describe the recursive step of the construction. Given (n, k) , where $n \geq k \geq 2$ and $k \neq 3$, the goal is to list all permutations of n items so that successive permutations differ in exactly k positions. As a preliminary, note that if it is possible to generate all permutations with k -differences so that the first

$n = 4$	$n = 5$	$n = 6$
1 2 3 4	1 2 3 4 5	1 2 3 4 5 6
1 3 4 2	.	.
1 4 2 3	.	.
4 2 1 3	.	.
4 1 3 2	1 4 2 3 5	1 4 2 3 5 6
4 3 2 1	5 4 1 3 2	6 4 1 3 5 2
2 4 3 1	.	.
2 3 1 4	.	.
2 1 4 3	.	.
3 2 4 1	5 3 4 1 2	6 3 4 1 5 2
3 4 1 2	4 3 2 1 5	5 3 4 1 2 6
3 1 2 4	.	.
	.	.
	.	.
	4 1 3 2 5	5 1 3 4 2 6
	2 4 3 1 5	4 1 3 2 5 6
	.	.
	.	.
	.	.
	2 1 4 3 5	4 2 1 3 5 6
	3 2 4 1 5	2 3 1 4 5 6
	.	.
	.	.
	3 1 2 4 5	2 4 3 1 5 6
		3 2 4 1 5 6
		.
		.
		3 1 2 4 5 6

FIG. 6. Listing even permutations of $1 \cdots n$ with 3-differences.

and last permutations differ in k positions, then we can arrange for the first and last to differ in *any* k positions. In particular, suppose there is an algorithm, $A(x_1 \cdots x_n, k)$ to list all permutations of $x_1 \cdots x_n$ with k -differences, starting with $x_1 \cdots x_n$, so that the first and last permutations differ exactly in the first k positions. Then for any i , $1 \leq i \leq n$, there is an algorithm $B(x_1 \cdots x_n, k, i)$ that does the same, except that the first and last permutations differ exactly in the k positions *beginning with position i* , allowing wrap-around. Algorithm B works as follows: apply algorithm A to $x_i x_{i+1} \cdots x_n x_1 \cdots x_{i-1}$, but then rotate each permutation generated by A to the right $i - 1$ positions. See Fig. 7.

Now, for the recursive construction, the list PLIST of permutations of $1 \cdots n$ with k -differences is composed of two parts:

$$\text{PLIST} = \text{LIST } A, \text{LIST } B.$$

(Following [18], a comma is used to denote the concatenation of lists.) LIST B will be obtained from LIST A by swapping the first two positions in every permutation on LIST A .

For $m = \binom{n}{2}$, let $T_1, \dots, T_m$ be a listing of all 2-sets of $[n]$ which satisfies properties (1) and (2) of § 2. That is, $T_1 = \{1, 2\}$, $T_m = \{3, 4\}$ and for $1 \leq j < m$, the permutations $(1 \cdots n) \setminus T_j$ and $(1 \cdots n) \setminus T_{j+1}$ differ in exactly one position.

To describe LIST A , first consider the case $k \geq 4$, $k \neq 5$. Corresponding to each 2-set $T_i = \{x_i, y_i\}$ will be a sublist L_i . L_i consists of a recursive listing of the permutations of $(1 \cdots n) \setminus \{x_i, y_i\}$ with $k - 2$ -differences, each permutation prefixed alternately with $x_i y_i$, $y_i x_i$. In order to complete the specification of L_i , two things must be determined, one of which depends on L_{i-1} and the other on T_{i+1} .

First, which of $x_i y_i$, $y_i x_i$ should be the prefix of the first permutation on L_i ? This is decided so as to force the last element of L_{i-1} and the first element of L_i to differ in the first two positions. See L_1, L_2, L_3 in Fig. 8 when $(n, k) = (6, 4)$. In this case, $T_1 = \{1, 2\}$, $T_2 = \{2, 3\}$, and $T_3 = \{1, 3\}$.

Second, when generating recursively the permutations of $(1 \cdots n) \setminus T_i$ with $k - 2$ -differences, how should the last permutation differ from the first? For $i < m$, we will do

$A(1\ 2\ 3\ 4, 2)$	$A(4\ 1\ 2\ 3, 2)$	$B(1\ 2\ 3\ 4, 2, 4)$
1 2 3 4	4 1 2 3	1 2 3 4
1 2 4 3	4 1 3 2	1 3 2 4
1 4 2 3	4 3 1 2	3 1 2 4
4 1 2 3	3 4 1 2	4 1 2 3
4 1 3 2	3 4 2 1	4 2 1 3
1 4 3 2	4 3 2 1	3 2 1 4
1 3 4 2	4 2 3 1	2 3 1 4
1 3 2 4	4 2 1 3	2 1 3 4
3 1 2 4	2 4 1 3	4 1 3 2
3 1 4 2	2 4 3 1	4 3 1 2
3 4 1 2	2 3 4 1	3 4 1 2
4 3 1 2	3 2 4 1	2 4 1 3
4 3 2 1	3 2 1 4	2 1 4 3
3 4 2 1	2 3 1 4	3 1 4 2
3 2 4 1	2 1 3 4	1 3 4 2
3 2 1 4	2 1 4 3	1 4 3 2
2 3 1 4	1 2 4 3	2 4 3 1
2 3 4 1	1 2 3 4	2 3 4 1
2 4 3 1	1 3 2 4	3 2 4 1
4 2 3 1	3 1 2 4	1 2 4 3
4 2 1 3	3 1 4 2	1 4 2 3
2 4 1 3	1 3 4 2	3 4 2 1
2 1 4 3	1 4 3 2	4 3 2 1
2 1 3 4	1 4 2 3	4 2 3 1

FIG. 7. Making the first and last permutations differ in k positions, beginning with position i .

L_1	12 3456
	21 3465
	12 3645
	$\vdots$
	21 4356
L_2	32 1456
	23 1465
	32 1645
	$\vdots$
	23 4165
L_3	31 2456
	$\vdots$

FIG. 8. Prefix of first element of L_i for $i = 1, 2, 3$ when $(n, k) = (6, 4)$.

this in such a way to ensure that last (L_i) and first (L_{i+1}) differ in exactly k positions, where first (L) and last (L) are used to denote the first and last elements, respectively, of a list L . By definition of the T_i , the two permutations

$$p_i = (1 \cdots n) \setminus T_i$$

and

$$p_{i+1} = (1 \cdots n) \setminus T_{i+1}$$

differ in exactly one position j (and position j of p_{i+1} contains an element of T_i). So, assume by induction that we can generate recursively the permutations of $(1 \cdots n) \setminus T_i$ with $k - 2$ differences so that the first and last permutations differ in $k - 2$ positions. We would like these $k - 2$ positions to include position j . To be specific, we will arrange for the first and last permutations to differ in the $k - 2$ positions beginning with position j . For if q is a permutation of $(1 \cdots n) \setminus T_i$ such that

1. q and p_i differ in $k - 2$ positions and
2. j is one of the positions in which q and p_i differ,

then also q and p_{i+1} differ in exactly $k - 2$ positions. It follows then that last (L_i) and first (L_{i+1}) will differ in exactly k positions.

As an example, when $(n, k) = (6, 4)$, note from Fig. 5 that $T_3 = \{1, 3\}$ and $T_4 = \{1, 4\}$. From Fig. 8, first (L_3) = 312456. Thus, last (L_3) has prefix 13, forcing first (L_4) to have prefix 41. Then, first (L_4) = 412356. To construct all of L_3 , we need to generate the permutations of $p_3 = 2456$ with 2-differences. Since $p_4 = 2356$, p_3 and p_4 differ in position 2, so we must arrange for the last permutation of 2456 to differ from 2456 in the two positions beginning with position 2. The result is shown in the first column of Fig. 9, which describes all of LIST A , as well as LIST B for the case $(n, k) = (6, 4)$.

In the case $k = 5$, the construction differs only as follows. LIST A is composed of two sublists,

$$\text{LIST } A = \text{LIST } A1, \text{LIST } A2.$$

Similar to LIST A , the lists LIST $A1$ and LIST $A2$ consist of sublists $L1_i$ and $L2_i$, respectively, corresponding to each 2-set $T_1, \cdots, T_m$. $L1_i$ is constructed exactly as L_i ,

LISTA		LISTB	
12 3456	64 1235	21 3456	46 1235
⋮	⋮	⋮	⋮
21 4356	46 5231	12 4356	64 5231
⋮	⋮	⋮	⋮
32 1456	65 1234	23 1456	56 1234
⋮	⋮	⋮	⋮
23 4156	56 4231	32 4156	65 4231
⋮	⋮	⋮	⋮
31 2456	45 1236	13 2456	54 1236
⋮	⋮	⋮	⋮
13 2546	54 1263	31 2546	45 1263
⋮	⋮	⋮	⋮
41 2356	35 1246	14 2356	53 1246
⋮	⋮	⋮	⋮
14 2365	53 1426	41 2365	35 1426
⋮	⋮	⋮	⋮
51 2346	25 1346	15 2346	52 1346
⋮	⋮	⋮	⋮
15 6342	52 1364	51 6342	25 1364
⋮	⋮	⋮	⋮
61 2345	24 1356	16 2345	42 1356
⋮	⋮	⋮	⋮
16 3245	42 1536	61 3245	24 1536
⋮	⋮	⋮	⋮
62 1345	34 1256	26 1345	43 1256
⋮	⋮	⋮	⋮
26 1435	43 2156	62 1435	34 2156
⋮	⋮	⋮	⋮
63 1245		36 1245	
⋮		⋮	
36 1254		63 1254	

FIG. 9. $\text{PLIST} = \text{LIST } A, \text{LIST } B \text{ for } (n, k) = (6, 4)$.

except that since it is impossible to generate *all* permutations of $(1 \cdots n) \setminus T_i$ with $k - 2 = 3$ -differences, we generate only the *even* permutations. $L2_i$ is obtained from $L1_i$ as follows. If

$$(1 \cdots n) \setminus T_i = a_1 a_2 \cdots a_{n-2}$$

then $L2_i$ is obtained by swapping a_1 and a_2 in every permutation on the list $L1_i$. See Fig. 10 for an example in the case where $(n, k) = (6, 5)$.

LISTA1	LISTA2	LISTB1	LISTB2
123456	641235	124356	642135
⋮	⋮	⋮	⋮
215346	465132	215436	465231
⋮	⋮	⋮	⋮
321456	651234	324156	652134
⋮	⋮	⋮	⋮
235146	564132	235416	563214
⋮	⋮	⋮	⋮
312456	451236	314256	452136
⋮	⋮	⋮	⋮
132645	546213	134625	543126
⋮	⋮	⋮	⋮
412356	351246	413256	352146
⋮	⋮	⋮	⋮
146325	531624	146235	532614
⋮	⋮	⋮	⋮
512346	251346	513246	253146
⋮	⋮	⋮	⋮
156243	526314	156342	524136
⋮	⋮	⋮	⋮
612345	241356	613245	243156
⋮	⋮	⋮	⋮
164235	421635	164325	423615
⋮	⋮	⋮	⋮
621345	341256	623145	342156
⋮	⋮	⋮	⋮
261534	435126	263514	436126
⋮	⋮	⋮	⋮
631245		632145	
⋮		⋮	
365214		364215	

FIG. 10. The case $k = 5$: listing permutations of $1 \cdots 6$ with 5-differences.

The entire construction is summarized in the procedures PERMUTE and PARTIAL of Fig. 11 and 12. These procedures should be regarded as a concise description of the list rather than an efficient procedure for generating the list. (In fact, the algorithm can be implemented to run in time proportional to the size of the output using storage $O(n)$ instead of $n!$ storage.)

As for correctness, it follows from § 3 and the discussion at the beginning of this section that the construction works correctly for the basis cases. For the inductive case, we assume inductively that sublists L_i and L_{i+1} are as claimed. In particular then, every permutation occurs exactly once on PLIST. Furthermore, these sublists were defined so that last (L_i) and first (L_{i+1}) would differ in exactly k positions for $1 \leq i < (\frac{n}{2})$, and similarly for L_{i+1} and L_{i+2} . It remains to show that the lists LIST A and LIST B (when $k \neq 5$) and the lists (when $k = 5$) LIST $A1$, LIST $A2$, LIST $B1$, LIST $B2$ glue together properly and that the final list is cyclic.

To show for $k \neq 5$ that last (LIST A) and first (LIST B) differ exactly in the first k positions, note that last (LIST A) = last (L_m) and $T_m = \{3, 4\}$. So,

$$\text{first} (L_m) = 341256 \cdots k \cdots n \text{ or } 431256 \cdots k \cdots n.$$

Then if $q = \text{last} (L_m)$, recall that

1. q differs from first (L_m) exactly in the first k positions and
2. the first two positions of q contain 3 and 4 in some order.

Since first (LIST B) = $213456 \cdots k \cdots n$, it follows that q also differs from first (LIST B) exactly in the first k positions. Similarly, last (LIST B) and first (LIST A) differ exactly in the first k positions, so that the listing is cyclic.

In the case $k = 5$, we must show that in each of the following pairs, the two permutations differ in exactly five positions.

1. last (LIST $A1$), first (LIST $A2$);
2. last (LIST $B1$), first (LIST $B2$);
3. last (LIST $A2$), first (LIST $B1$).

To show 1, first (LIST $A2$) = $124356 \cdots n$ and last (LIST $A1$) = last (L_{1m}). Since $T_m = \{3, 4\}$,

$$\text{first} (L_{1m}) = 341256 \cdots n \text{ or } 431256 \cdots n,$$

```

procedure PERMUTE( $a_1 \dots a_n, k, t$ )
{Returns a list of all permutations (only the even ones if  $k = 3$ ) of  $a_1 \dots a_n$  with
 $k$ -differences, starting with  $a_1 \dots a_n$  and ending at a permutation which differs
from  $a_1 \dots a_n$  exactly in the  $k$  positions beginning with position  $t$ . Assumes
 $n \geq k \geq 2$ .}

begin
 $b_1 \dots b_n := a_1 \dots a_n a_1 \dots a_{t-1}$ 
if  $k = 2$  then LIST := Johnson-Trotter list for  $b_1 \dots b_n$ 
    from  $b_1 \dots b_n$  to  $b_2 b_1 b_3 \dots b_n$ .
if  $k = 3$  then LIST := even permutations of  $b_1 \dots b_n$  with 3-differences
    from  $b_1 \dots b_n$  to  $b_3 b_1 b_2 b_4 \dots b_n$ .
else begin
    LISTA := PARTIAL( $b_1 \dots b_n, k$ ) {See Figure 12.}
    LISTB := LISTA, but swap first two positions of every permutation
    on LISTA.
    LIST := LISTA, LISTB
end
Rotate each permutation on LIST to the right  $t - 1$  positions
return(LIST)
end { procedure }

```

FIG. 11. The procedure PERMUTE.

```

procedure PARTIAL( $b_1 \dots b_n, k$ )
{Returns LISTA of half the permutations of  $b_1 \dots b_n$  with  $k$ -differences. The
first permutation is  $b_1 \dots b_n$  and the last permutation differs from  $b_1 \dots b_n$  ex-
actly in positions  $1 \dots k$ . Assumes  $n \geq k \geq 4$ . }

begin
Initialize lists LISTA, LISTA1, and LISTA2 to empty.
( $x, y$ ) := (1, 2)
for  $i := 1$  to  $m$  do
  begin
 $p := b_1 \dots b_n \setminus \{x, y\}$ 
if  $i < m$  then
  begin
 $\{x', y'\} := T_{i+1}$ 
 $p' := b_1 \dots b_n \setminus \{x', y'\}$ 
 $j :=$  position in which  $p$  and  $p'$  differ
  else  $j := 1$ 
   $LTEMP := PERMUTE(p, k - 2, j)$  {See Figure 11. }
  Prefix each element of LTEMP alternately with  $b_x b_y, b_y b_x,$ 
  beginning with  $b_x b_y$ .
  if  $i < m$  then begin
    Let  $uv$  be the prefix of the last element of LTEMP.
    Assign to  $(x, y)$  either  $(x', y')$  or
     $(y', x')$  so that  $u \neq x$  and  $v \neq y$ 
  end {if }
  if  $k \neq 5$  then LISTA := LISTA, LTEMP
  else begin
    LISTA1 := LISTA1, LTEMP
    Let  $p = r_1 \dots r_{n-2}$  and swap  $r_2$  and  $r_3$ 
    in every permutation on LTEMP
    LISTA2 := LISTA2, LTEMP
  end { else }
end { for }
if  $k = 5$  then LISTA := LISTA1, LISTA2
return(LISTA)
end { procedure }

```

FIG. 12. The procedure PARTIAL.

so

$$\text{last}(L1_m) = 435126 \cdots n \text{ or } 345126 \cdots n,$$

both of which differ from first (LIST A2) exactly in the first five positions. Case 2 is similar.

To show 3,

$$\text{first}(\text{LIST } B1) = 213456 \cdots n.$$

The last sublist of LIST A2 is $L2_m$ which is obtained from $L1_m$ by swapping elements 1 and 2 in every permutation. So, last (LIST A2) = last ($L1_m$) with elements 1 and 2 swapped. Then,

$$\text{last}(\text{LIST } A2) = 345216 \cdots n \text{ or } 435216 \cdots n,$$

both of which differ exactly in the first five positions from first (LIST B1). Finally, to show that the list is cyclic, note that

$$\text{last}(\text{LIST } B) = 345216 \cdots n \text{ or } 435216 \cdots n,$$

both of which differ in the first five positions from

$$\text{first}(\text{LIST } A) = 1 \cdots n.$$

5. Conclusions. Although we have been able to list permutations to differ in k positions, these positions are not always contiguous unless $k = 2$ or $k = n$. In Fig. 9, for example, successive permutations 132546 and 412356 from LIST A differ in the four positions 1, 2, 4, and 5, which are not contiguous even if the last position is considered contiguous to the first. However, “differing in k contiguous positions” would also give rise to a connected, undirected, vertex transitive graph on permutations. So, it seems likely that permutations could be listed by k -contiguous differences; otherwise, this would be a counterexample to the conjecture that all graphs in this class do have Hamilton

paths. In any case, requiring contiguous differences would appear to require a different approach.

Acknowledgments. Thanks to Herb Wilf and the referee for helpful suggestions and to Terry Wang for comments on an earlier version of this paper.

REFERENCES

- [1] J. BOOTHROYD, PERM (Algorithm 6), *Comput. Bull.*, 9 (1965), p. 104.
- [2] ———, *Permutation of the elements of a vector* (Algorithm 29) and *Fast permutation of the elements of a vector* (Algorithm 30), *Comput. J.*, 10 (1967), pp. 310–312.
- [3] M. BUCK AND D. WIEDEMANN, *Gray codes with restricted density*, *Discrete Math.*, 48 (1984), pp. 163–171.
- [4] P. EADES, M. HICKEY, AND R. C. READ, *Some Hamilton paths and a minimal change algorithm*, *J. Assoc. Comput. Mach.*, 31 (1984), pp. 19–29.
- [5] C. COMPTON, *Hamilton cycles in the Cayley graph of S_n and a doubly adjacent Gray code*, Ph.D. Thesis, University of California, San Diego, 1990.
- [6] P. EADES AND B. MCKAY, *An algorithm for generating subsets of fixed size with a strong minimal change property*, *Inform. Process. Lett.*, 19 (1984), pp. 131–133.
- [7] R. B. EGLETON AND W. D. WALLIS, *Problem 86: Solution I*, *Math. Mag.*, 58 (1985), pp. 112–113.
- [8] M. HALL, JR., *Combinatorial Theory*, second edition, John Wiley, New York, 1986.
- [9] B. R. HEAP, *Permutations by interchanges*, *Comput. J.*, 6 (1963), pp. 293–294.
- [10] T. HOUGH AND F. RUSKEY, *An efficient implementation of the Eades, Hickey, Read adjacent interchange combination generation algorithm*, *J. Combin. Math. Comb. Comput.*, 4 (1988), pp. 79–86.
- [11] R. J. GOULD AND R. ROTH, *Cayley digraphs and $(1, j, n)$ sequencings of the alternating group A_n* , *Discrete Math.*, 66 (1987), pp. 91–102.
- [12] B. JACKSON, *Hamilton cycles in regular 2-connected graphs*, *J. Combin. Theory Ser. B*, 29 (1980), pp. 27–46.
- [13] S. M. JOHNSON, *Generation of permutations by adjacent transpositions*, *Math. Comput.*, 17 (1963), pp. 282–285.
- [14] J. T. JOICHI AND D. E. WHITE, *Gray codes in graphs of subsets*, *Discrete Math.*, 31 (1980), pp. 29–41.
- [15] V. L. KOMP'EL'MAKHER AND V. A. LISKOVETS, *Sequential generation of arrangements by means of a basis of transpositions*, *Kibernetika*, 3 (1975), pp. 17–21.
- [16] L. LOVÁSZ, *Problem 11*, in *Combinatorial Structures and Their Applications*, Gordon and Breach, London, 1970.
- [17] J. METZGER, *Problem 1186*, *Math. Mag.*, 58 (1985), pp. 113–114.
- [18] A. NIJENHUIS AND H. S. WILF, *Combinatorial Algorithms for Computers and Calculators*, Academic Press, Inc., New York, 1978.
- [19] T. C. PUTNAM, *A Gray code variant: sequencing permutations via fixed points*, manuscript, 1989.
- [20] S. RABINOWITZ, *Problem 1186*, *Math. Mag.*, 57 (1984), p. 109.
- [21] D. F. RALL AND P. J. SLATER, *Generating all permutations by graphical derangements*, unpublished manuscript.
- [22] F. RUSKEY AND G. PRUESSE, *Transposition generation of the linear extensions of certain posets*, *SIAM J. Discrete Math.*, to appear.
- [23] F. RUSKEY AND C. SAVAGE, *Hamilton cycles which extend transposition matchings in Cayley graphs of S_n* , submitted.
- [24] R. SEDGEWICK, *Permutation generation methods*, *Computing Surveys*, 9 (1977), pp. 137–164.
- [25] H. F. TROTTER, PERM (Algorithm 115), *Comm. ACM*, 5 (1962), pp. 434–435.
- [26] M. B. WELLS, *Generation of permutations by transposition*, *Math. Comput.*, 15 (1961), pp. 192–195.
- [27] H. S. WILF, *Combinatorial Algorithms: An Update*, CBMS-NSF Regional Conference Series 55, Society for Industrial and Applied Mathematics, Philadelphia, 1989.
- [28] ———, *personal communication*, 1988.

ON THE NUMBER OF DISTINCT FORESTS*

LAJOS TAKÁCS†

Abstract. This paper contains a simple explicit formula, a recurrence formula and an asymptotic expansion for the number of distinct forests with n labeled vertices.

Key words. enumeration of forests, exact and asymptotic formulas

AMS(MOS) subject classifications. 05C05, 05C30

1. Introduction. A forest is a simple graph that has no cycles. In other words, a forest is a simple graph, all of whose components are trees. Denote by $F(n)$ the number of distinct forests having vertex set $\{1, 2, \dots, n\}$. In 1959, Dénes [3] and Rényi [7] gave two explicit expressions for $F(n)$. Rényi also proved that

$$(1) \quad \lim_{n \rightarrow \infty} F(n)/n^{n-2} = \sqrt{e} = 1.6487212707 \dots$$

See also Riordan [8]. In 1980 Stanley [10] observed that $F(n)$ can also be interpreted as the number of different score vectors in a tournament in which each pair of n players $1, 2, \dots, n$ plays a game. For each player a game may result in either a win, a tie, or a loss. Denote by v_i the total number of wins of player i . Then $(v_1, v_2, \dots, v_n)$ is the score vector of the tournament. In 1981, Kleitman and Winston [6] proved that there is a one-to-one correspondence between the elements of the set of forests with n labeled vertices and the elements of the set of score vectors in a tournament of n players. The sequence $F(1), F(2), \dots, F(10)$ is listed as Sequence 714 in the book of Sloane [9].

2. New results. In what follows we derive a simple explicit formula for $F(n)$. Namely we prove that

$$(2) \quad F(n) = \frac{n!}{n+1} \sum_{j=0}^{\lfloor n/2 \rfloor} (-1)^j \frac{(2j+1)(n+1)^{n-2j}}{2^j j! (n-2j)!}$$

if $n \geq 1$, or equivalently,

$$(3) \quad F(n) = H_n(n+1) - nH_{n-1}(n+1)$$

for $n \geq 1$ where $H_n(x)$ is the n th Hermite polynomial defined by

$$(4) \quad H_n(x) = n! \sum_{j=0}^{\lfloor n/2 \rfloor} \frac{(-1)^j x^{n-2j}}{2^j j! (n-2j)!}$$

for $n \geq 0$. We have $H_0(x) = 1$, $H_1(x) = x$ and

$$(5) \quad H_n(x) = xH_{n-1}(x) - (n-1)H_{n-2}(x)$$

for $n \geq 2$. It is convenient to use the recurrence formula (5) for calculating numerically $H_{n-1}(n+1)$ and $H_n(n+1)$ in (3). Table 1 contains $F(n)$ for $n \leq 24$.

We can also express $F(n)$ in the following way:

$$(6) \quad F(n) = \frac{n^n e^{1/2}}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-x^2/2 - ix} \left(1 + \frac{ix}{n}\right)^n ix \, dx$$

* Received by the editors August 3, 1989; accepted for publication October 4, 1989.

† Department of Mathematics and Statistics, Case Western Reserve University, Cleveland, Ohio 44106.

TABLE 1

n	$F(n)$	n	$F(n)$	n	$F(n)$	n	$F(n)$
1	1	7	36,961	13	$3.52563011 \cdots 10^{12}$	19	$1.02392064 \cdots 10^{22}$
2	2	8	561,948	14	$1.10284283 \cdots 10^{14}$	20	$4.86909744 \cdots 10^{23}$
3	7	9	10,026,505	15	$3.74835769 \cdots 10^{15}$	21	$2.44766976 \cdots 10^{25}$
4	38	10	205,608,536	16	$1.37557910 \cdots 10^{17}$	22	$1.29692217 \cdots 10^{27}$
5	291	11	4,767,440,679	17	$5.42117905 \cdots 10^{18}$	23	$7.22423439 \cdots 10^{28}$
6	2932	12	123,373,203,208	18	$2.28359487 \cdots 10^{20}$	24	$4.22040860 \cdots 10^{30}$

for $n \geq 1$. This integral representation is convenient to obtain the asymptotic expansion of $F(n)/n^n$ as $n \rightarrow \infty$. We shall prove that

$$(7) \quad F(n) \sim n^n e^{1/2} \sum_{\nu=0}^{\infty} \frac{f_{\nu}}{n^{\nu}}$$

as $n \rightarrow \infty$ where the coefficients f_{ν} ($\nu = 0, 1, 2, \dots$) are given in Table 2.

Explicitly,

$$(8) \quad f_{\nu} = \sum_{i=1}^{[\nu/2]} (-1)^{i-1} \frac{C(\nu, 2i-1)}{2^{\nu-i}(\nu-i)!}$$

for $\nu \geq 2$ where $C(\nu, j)$, ($0 \leq j \leq \nu-1$), is the number of permutations of $(1, 2, \dots, 2\nu-j)$ with $\nu-j$ cycles each of length >1 . $C(\nu, j) = 0$ if $j \geq \nu$. We have also

$$(9) \quad f_{\nu+1} = f_{\nu} + \sum_{i=0}^{[(\nu-1)/2]} (-1)^i \frac{C(\nu, 2i)}{2^{\nu-i-1}(\nu-i-1)!}$$

for $\nu \geq 1$. Table 3 contains $C(\nu, j)$ for $0 \leq j \leq \nu-1$ and $\nu \leq 10$.

We can calculate $C(\nu, j)$ for $0 \leq j \leq \nu-1$ by the recurrence formula

$$(10) \quad C(\nu+1, j) = [C(\nu, j) + C(\nu, j-1)](2\nu+1-j)$$

where $0 \leq j \leq \nu$, $\nu \geq 1$, $C(1, 0) = 1$, $C(1, j) = 0$ for $j > 0$ and $C(\nu, -1) = 0$ for $\nu \geq 1$. In particular, $C(\nu, 0) = 1 \cdot 3 \cdots (2\nu-1) = (2\nu)!/2^{\nu}\nu!$ for $\nu \geq 1$ and $C(\nu, \nu-1) = \nu!$ for $\nu \geq 1$.

The fact that the coefficients $C(\nu, j)$, $0 \leq j \leq \nu-1$, satisfy (10) can be proved in the following way: By definition $C(\nu+1, j)$ is the number of permutations of $(1, 2, \dots, 2\nu-j+2)$ with $\nu-j+1$ cycles each of length >1 . There are two possibilities: (a) Element $2\nu-j+2$ belongs to a cycle which contains more than two elements. If we remove element $2\nu-j+2$, the remaining permutation of $(1, 2, \dots, 2\nu-j+1)$ contains $\nu-j+1$ cycles of length >1 . There are $C(\nu, j-1)$ such permutations and ele-

TABLE 2

ν	f_{ν}	ν	f_{ν}
0	0	6	-17,207/384
1	0	7	-3,607/768
2	1	8	1,408,301/3072
3	5/2	9	8,181,503/6144
4	11/8	10	-137,483,257/61440
5	-203/16	11	-24,971,924,401/983040

TABLE 3
 $C(u, j)$

$\nu \backslash j$	0	1	2	3	4	5	6	7	8	9
1	1									
2	3	2								
3	15	20	6							
4	105	210	130	24						
5	945	2520	2380	924	120					
6	10395	34650	44100	26432	7308	720				
7	135135	540540	866250	705320	303660	64224	5040			
8	2027025	9459450	18288270	18858840	11098780	3678840	623376	40320		
9	34459425	183783600	416215800	520059540	389449060	177331440	47324376	6636960	362880	
10	654729075	3928374450	10199989800	14980405440	13642629000	7934927000	2920525608	647536032	76998240	3628800

ment $2\nu - j + 2$ can be included in $2\nu - j + 1$ ways. (b) Element $2\nu - j + 2$ belongs to a cycle which contains two elements. The companion element of $2\nu - j + 2$ can be chosen in $2\nu - j + 1$ ways. The remaining $2\nu - j$ elements form a permutation with $\nu - j$ cycles each of length > 1 . The number of such permutations is $C(\nu, j)$. This proves (10).

From (10) it follows by mathematical induction that

$$(11) \quad \sum_{i=0}^{\nu-1} (-1)^i C(\nu, i) = 1$$

for all $\nu \geq 1$. By (10) it follows also that (8) and (9) are equivalent.

In proving the above results we shall make use of some properties of the Hermite polynomials and of the Stirling numbers of the first kind.

3. Hermite polynomials. The m th Hermite polynomial is defined by

$$(12) \quad H_m(x) = (-1)^m e^{x^2/2} D^m e^{-x^2/2}$$

for $m = 0, 1, 2, \dots$ where $D = d/dx$ is the differential operator. Equivalently, we can write that

$$(13) \quad H_m(x) = e^{-D^2/2} x^m$$

for $m = 0, 1, 2, \dots$. Definition (13) has many advantages over (12). For example, by (13), it follows immediately that

$$(14) \quad \sum_{m=0}^{\infty} \frac{H_m(x)}{m!} z^m = e^{-D^2/2} e^{xz} = e^{-z^2/2} e^{xz}$$

and $DH_m(x) = mH_{m-1}(x)$ for $m \geq 1$. We mention that

$$(15) \quad H_m(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-u^2/2} (x - iu)^m du$$

for $m \geq 0$. This follows from (13) or from (5). Furthermore, we have

$$(16) \quad \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-x^2/2} x^n dx = i^n H_n(0) = \begin{cases} 1 \cdot 3 \cdots (2m-1) & \text{if } n = 2m, \\ 0 & \text{if } n = 2m-1, \end{cases}$$

and $m = 1, 2, \dots$.

4. Stirling numbers of the first kind. We define $S(n, k)$ for $0 \leq k \leq n$ as the number of permutations of $(1, 2, \dots, n)$ with k cycles, and $S(0, 0) = 1$. We define also $S(n, k) = 0$ for $k > n \geq 0$. The numbers $S(n, k)$, $0 \leq k \leq n$, are called Stirling numbers of the first kind. They can easily be determined by the recurrence formula

$$(17) \quad S(n+1, k) = S(n, k-1) + nS(n, k)$$

for $n \geq 0$ and $k \geq 1$ where $S(0, 0) = 1$, $S(n, 0) = 0$ for $n \geq 1$, and $S(0, k) = 0$ for $k \geq 1$. Formula (17) reflects the fact that in a permutation of $(1, 2, \dots, n+1)$, the element $n+1$ may form a cycle by itself or it may belong to a cycle of length > 1 . See Table 4 for $S(n, k)$, $0 \leq k \leq n \leq 6$.

By (17) it follows that

$$(18) \quad \sum_{k=0}^n S(n, k) x^k = x(x+1) \cdots (x+n-1)$$

for $n \geq 1$.

TABLE 4
 $S(n, k)$

$n \backslash k$	0	1	2	3	4	5	6
0	1						
1	0	1					
2	0	1	1				
3	0	2	3	1			
4	0	6	11	6	1		
5	0	24	50	35	10	1	
6	0	120	274	225	85	15	1

Obviously,

$$(19) \quad S(n, n-\nu) = \sum_{i=0}^{\nu-1} C(\nu, i) \binom{n}{2\nu-i}$$

for $1 \leq \nu \leq n$ where $C(\nu, i)$, $0 \leq i \leq \nu-1$, is the number of permutations of $(1, 2, \dots, 2\nu-i)$ with $\nu-i$ cycles each of length >1 . For, a permutation of $(1, 2, \dots, n)$ with $n-\nu$ cycles may have $n+i-2\nu$ cycles of length 1 and $\nu-i$ cycles of length >1 where $i = 0, 1, \dots, \nu-1$. The representation (19) was found by Jordan [4], [5 p. 150] in 1933. See also Ward [11] and Carlitz [1].

From (19) we obtain by inversion that

$$(20) \quad C(\nu, i) = \sum_{j=0}^{\nu-i} (-1)^{\nu-i-j} \binom{2\nu-i}{\nu+j} S(\nu+j, j)$$

for $0 \leq i \leq \nu-1$.

5. The determination of $F(n)$. By definition $F(n)$ is the number of distinct forests having vertex set $\{1, 2, \dots, n\}$. A forest may consist of $r = 1, 2, \dots, n$ distinct trees of orders $t_1, t_2, \dots, t_r$ where $t_1 + t_2 + \dots + t_r = n$. By a formula of Cayley [2] the number of distinct trees with t labeled vertices is t^{t-2} . Thus we obtain that

$$(21) \quad F(n) = \sum_{r=1}^n \frac{1}{r!} \sum_{t_1 + \dots + t_r = n} \frac{n!}{t_1! t_2! \dots t_r!} t_1^{t_1-2} t_2^{t_2-2} \dots t_r^{t_r-2}$$

for $n \geq 1$. Write $F(0) = 1$. Then

$$(22) \quad \sum_{n=0}^{\infty} \frac{F(n)}{n!} x^n = \exp \left\{ \sum_{n=1}^{\infty} \frac{n^{n-2}}{n!} x^n \right\}$$

for $|x| \leq e^{-1}$. If $|x| \leq e^{-1}$, then the equation

$$(23) \quad ye^{-y} = x$$

has exactly one root $y = y(x)$ in the unit disk $|y| \leq 1$, and by Lagrange's expansion we obtain that

$$(24) \quad y - \frac{y^2}{2} = \sum_{n=1}^{\infty} \frac{n^{n-2}}{n!} x^n$$

for $|x| \leq e^{-1}$. Thus by (22) and (14)

$$(25) \quad \sum_{n=0}^{\infty} \frac{F(n)}{n!} x^n = e^{y-y^2/2} = \sum_{r=0}^{\infty} \frac{H_r(1)}{r!} y^r.$$

Since by Lagrange's expansion

$$(26) \quad y^r = [y(x)]^r = r \sum_{n=r}^{\infty} \frac{n^{n-r}}{n(n-r)!} x^n$$

for $|x| \leq e^{-1}$ and $r \geq 1$, by (25) and (26) we get

$$(27) \quad F(n) = \sum_{r=1}^n \binom{n-1}{r-1} n^{n-r} H_r(1)$$

for $n \geq 1$. Now by (13)

$$(28) \quad H_r(1) = [e^{-D^2/2} x^r]_{x=1},$$

and it follows from (27) that

$$(29) \quad \begin{aligned} F(n) &= \left[e^{-D^2/2} \sum_{r=1}^n \binom{n-1}{r-1} n^{n-r} x^r \right]_{x=1} = [e^{-D^2/2} x(n+x)^{n-1}]_{x=1} \\ &= [e^{-D^2/2} (n+x)^n]_{x=1} - n[e^{-D^2/2} (n+x)^{n-1}]_{x=1} = H_n(n+1) - nH_{n-1}(n+1) \end{aligned}$$

for $n \geq 1$. This proves (3), and by (4) we obtain (2).

By using (5) we can also express (3) in the following form:

$$(30) \quad F(n) = H_{n+1}(n+1) - nH_n(n+1)$$

for $n \geq 0$. By (15) and (30) we obtain (6) and the following formula

$$(31) \quad F(n) = \sum_{r=0}^n \binom{n}{r} n^{n-r} H_{r+1}(1)$$

for $n \geq 0$.

6. The asymptotic expansion of $F(n)$. Since by (18)

$$(32) \quad \binom{n}{r} = \frac{1}{r!} \sum_{j=0}^r (-1)^{r-j} S(r, j) n^j$$

for $0 \leq r \leq n$, by (31) we have

$$(33) \quad F(n) n^{-n} = \sum_{r=0}^n \frac{1}{r!} H_{r+1}(1) \sum_{j=0}^r (-1)^{r-j} S(r, j) \frac{1}{n^{r-j}} = e^{1/2} \sum_{\nu=0}^n \frac{f_{\nu}(n)}{n^{\nu}}$$

where

$$(34) \quad (-1)^{\nu} e^{1/2} f_{\nu}(n) = \sum_{j=0}^{n-\nu} \frac{S(j+\nu, j)}{(j+\nu)!} H_{j+\nu+1}(1)$$

for $0 \leq \nu \leq n$. The limit $\lim_{n \rightarrow \infty} f_\nu(n) = f_\nu$ exists for $\nu = 0, 1, 2, \dots$ and we have

$$(35) \quad (-1)^\nu e^{1/2} f_\nu = \sum_{j=0}^{\infty} \frac{S(j+\nu, j)}{(j+\nu)!} H_{j+\nu+1}(1).$$

If $\nu = 0$, then $S(j+\nu, j) = 1$ and

$$(36) \quad e^{1/2} f_0 = \sum_{j=0}^{\infty} \frac{H_{j+1}(1)}{j!} = [e^{-D^2/2} e^x x]_{x=0} = e^{1/2} H_1(0) = 0,$$

that is, $f_0 = 0$. If $\nu \geq 1$, then by (19)

$$(37) \quad \begin{aligned} (-1)^\nu e^{1/2} f_\nu &= \sum_{j=0}^{\infty} \frac{H_{j+\nu+1}(1)}{(j+\nu)!} \sum_{k=0}^{\nu-1} C(\nu, k) \binom{j+\nu}{2\nu-k} \\ &= \sum_{k=0}^{\nu-1} C(\nu, k) \frac{1}{(2\nu-k)!} \sum_{j=\nu-k}^{\infty} \frac{H_{j+\nu+1}(1)}{(j+k-\nu)!} = e^{1/2} \sum_{k=0}^{\nu-1} \frac{C(\nu, k)}{(2\nu-k)!} H_{2\nu+1-k}(0). \end{aligned}$$

Here we used that

$$(38) \quad \sum_{j=\nu-k}^{\infty} \frac{H_{j+\nu+1}(1)}{(j+k-\nu)!} = \left[e^{-D^2/2} \sum_{j=\nu-k}^{\infty} \frac{x^{j+\nu+1}}{(j+k-\nu)!} \right]_{x=1} = [e^{-D^2/2} e^x x^{2\nu+1-k}]_{x=1} = e^{1/2} H_{2\nu+1-k}(0).$$

If $k = 2i$ ($0 \leq i \leq \nu$), then $H_{2\nu+1-k}(0) = 0$ and if $k = 2i-1$ ($1 \leq i \leq \nu$), then by (4)

$$(39) \quad H_{2\nu-2i+2}(0) = \frac{(-1)^{\nu-i+1} (2\nu-2i+2)!}{2^{\nu-i+1} (\nu-i+1)!}.$$

Since $H_3(0) = 0$, by (37) $f_1 = 0$, and if $\nu \geq 2$, then by (37) and (39)

$$(40) \quad f_\nu = \sum_{i=1}^{\lfloor \nu/2 \rfloor} (-1)^{i-1} \frac{C(\nu, 2i-1)}{2^{\nu-i} (\nu-i)!}.$$

For $0 \leq \nu \leq n$ we have

$$(41) \quad |f_\nu - f_\nu(n)| \leq e^{-1/2} \sum_{k=n+1}^{\infty} \frac{S(k, k-\nu)}{k!} |H_{k+1}(1)|$$

where

$$(42) \quad \frac{|H_{k+1}(1)|}{\sqrt{(k+1)!}} \leq e^{1/4} 1.086435 \dots < 1.396$$

and

$$(43) \quad \lim_{k \rightarrow \infty} \frac{S(k, k-\nu)}{k^{2\nu}} = \frac{C(\nu, 0)}{(2\nu)!} = \frac{1}{2^\nu \nu!}.$$

Accordingly, for every $\nu \geq 0$, $\lim_{n \rightarrow \infty} f_\nu(n) = f_\nu$, and the rate of convergence is exponential. Consequently

$$(44) \quad F(n)n^{-n} \sim e^{1/2} \sum_{\nu=0}^{\infty} \frac{f_\nu}{n^\nu}$$

as $n \rightarrow \infty$ where $f_0 = f_1 = 0$ and f_ν for $\nu \geq 2$ is given by (40). This proves (7) and (8).

REFERENCES

- [1] L. CARLITZ, *Note on the numbers of Jordan and Ward*, Duke Math. J., 38 (1971), pp. 783–790.
- [2] A. CAYLEY, *A theorem on trees*, Quarterly Journal of Pure and Applied Mathematics, 23 (1889), pp. 376–378. (The Collected Mathematical Papers of Arthur Cayley. Vol. XIII. Cambridge University Press, 1897, pp. 26–28.)
- [3] J. DÉNES, *The representation of a permutation as the product of a minimal number of transpositions, and its connection with the theory of graphs*, Publications of the Mathematical Institute of the Hungarian Academy of Sciences, 4 (1959), pp. 63–71.
- [4] C. JORDAN, *On Stirling's numbers*, Tôhoku Mathematical Journal, 37 (1933), pp. 254–278.
- [5] ———, *Calculus of Finite Differences*, Budapest, 1939. Reprinted by Chelsea, New York, 1947.
- [6] D. J. KLEITMAN AND K. J. WINSTON, *Forests and score vectors*, Combinatorica, 1 (1981), pp. 49–54.
- [7] A. RÉNYI, *Some remarks on the theory of trees*, Publications of the Mathematical Institute of the Hungarian Academy of Sciences, 4 (1959), pp. 73–85.
- [8] J. RIORDAN, *Forests of labeled trees*, J. Combin. Theory, 5 (1968), pp. 90–103.
- [9] N. J. A. SLOANE, *A Handbook of Integer Sequences*, Academic Press, New York, 1973.
- [10] R. STANLEY, *Decompositions of rational convex polytopes*, Ann. Discrete Math., 6 (1980), pp. 333–342.
- [11] M. WARD, *The representation of Stirling's numbers and Stirling's polynomials as sums of factorials*, Amer. J. Math., 56 (1934), pp. 87–95.

A SIMPLIFIED ANOMALY AND REDUCTION FOR PRECEDENCE CONSTRAINED MULTIPROCESSOR SCHEDULING*

CRAIG A. TOVEY†

Abstract. Graham's multiprocessor precedence-constrained scheduling anomaly is shrunk where no list schedule is optimal for both $m = 2$ and 3 processors, and a simpler anomaly for $m > 2$ is constructed. As a corollary a very easy proof is obtained that shows that the general problem is NP-complete under tighter restrictions than previously known (schedule length 3, no precedence chains of length more than 2).

Key words. scheduling, complexity, sensitivity, NP-complete

A set of n unit length jobs are to be scheduled on m parallel identical machines to minimize the latest completion time. A directed acyclic precedence graph constrains the schedule. This problem, often referred to as $P|\text{prec}|p_j = 1|C_{\max}$, is shown NP-complete in [3] and [2]. (For all fixed $m \geq 3$ its complexity is open.) We confine our attention to list schedules; i.e., a schedule is a permutation of the jobs. A machine is assigned to the first available unprocessed job, where availability is with respect to the precedence graph.

If the number of machines is uncertain or subject to change, e.g., due to breakdowns and repairs, we would be interested in a *robust* schedule that gives good performance for varying m . However, optimality is no guarantee of robustness. For example, if the precedence graph is as in Fig. 1 then the list schedule $\{1, 2, 3, \dots\}$ is optimal for $m + 1$ processors but as bad as any possible list schedule (performance ratio $2 - 1/m$ [1]) for m processors. However, it is obvious that for this example, there exists a robust list schedule, which is optimal for both m and $m + 1$ processors.

Graham, in his classic work on scheduling anomalies [1], gives an example with $n = 30$ such that no one list schedule is optimal for both $m = 2$ and $m + 1 = 3$. The optimal schedules have no slack, but result in an extra time period when the machine number changes. Thus Graham's anomaly shows that when m changes from 2 to 3 (3 to 2), the most robust schedule may be nonoptimal by a factor of $11/10$ ($16/15$).

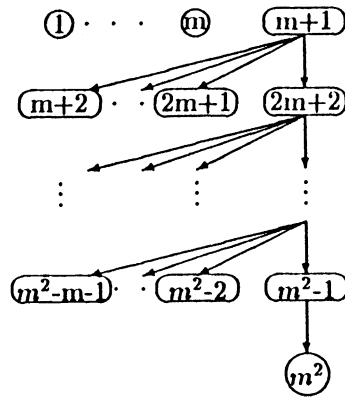
Graham's anomaly may be described as follows (see Fig. 2): jobs are a_i , b , w_i , x_i , y_i , and z_{ij} . Here for convenience indices run from 1 to the uppercase of the job class; e.g., the a_i run from a_1 to a_A . For the z_{ij} , $i = 1$ to X , and $j = 1$ to $m + 1$. The precedence constraints are: $a_i < x_j$, $a_i < y_k < z_{1j}$, $b < w_i < z_{1j}$, $b < y_k$, and $z_{jl} < z_{j+1,l}$, for all i , j , k , l .

In Graham's example, $A = 6$, $W = 1$, $Y = 6$, and $X = 4$. The key observation is that in any optimal schedule with $m = 2$ processors, b must precede some a , but with 3 processors all the a_i must precede b . Since none of these jobs has any predecessors, the same is true for any list schedule.

We can find constraints on A , W , X , Y , and m as follows. If all a 's precede b then the schedule is not optimal for m processors $\Leftrightarrow A + 1 \not\equiv 0 \pmod{m}$; if b precedes some a_i then nonoptimal for $m + 1$ processors $\Leftrightarrow 1 + w < m + 1$; $A \equiv 0 \pmod{m + 1}$; there exists a slackless schedule for $m + 1$ processors $\Leftrightarrow A \equiv 0 \pmod{m + 1}$; $X \geq m$; $X +$

* Received by the editors November 16, 1989; accepted for publication March 2, 1990. This research was supported by National Science Foundation grant ECS-8451032.

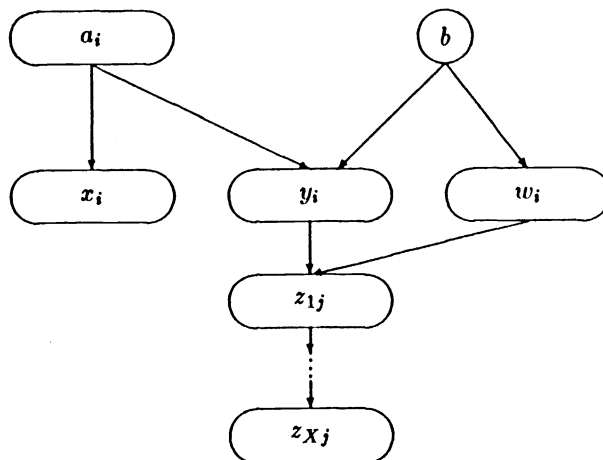
† Schools of Industrial and Systems Engineering and Information and Computer Science, Georgia Institute of Technology, Atlanta, Georgia 30332.

FIG. 1. Picture of $2 - 1/m$ anomaly.

$1 + Y + W \equiv 0 \pmod{m+1}$; $2 + Y + W \equiv 0 \pmod{m}$. This leads to an 18-node anomaly, $A = 6$, $X = Y = 2$, $W = 1$ for $m = 2$ and 3 processors, which also has a slackless schedule for 2 processors (the only Graham anomaly the author has been able to shrink).

Curiously, it is easier to find smaller anomalies for larger m . When $m > 3$, the smallest solution to the above is $A = m + 1$, $X = m$, $W = m - 2$, $Y = (m - 2)(m)$ for a total of $2m(m + 1)$ jobs. This solution seems fairly good, since a natural target size for an anomaly would be $m(m + 1)$ jobs. The solution fails for $m = 2$ because of the $A + 1 \not\equiv 0 \pmod{m}$ constraint (forcing $A = 2(m + 1)$ instead).

In fact, a much simpler and smaller construction will do for $m > 2$. Construct the bipartite graph $G = (V, U, E)$ with $|V| = 2m + 1$, $|U| = m + 2$, and precedences $v_i < u_j$: $1 \leq i \leq m$, for all j ; $v_i < u_j$: $m + 1 \leq i \leq 2m$, for all $j \neq 1$; $v_{2m+1} < u_j$: for all $j \neq 2, 3$. This graph is illustrated for $m = 3$ in Fig. 3. For an optimal schedule with $m + 1$ processors, the first $m + 1$ jobs run must be v_1 through v_m and v_{2m+1} , because no other choice of $m + 1$ jobs frees up a job for the second time period. On the other hand, in any optimal list schedule for m processors, the $2m$ jobs run in the first two time periods must be v_1 through v_{2m} because only this choice frees up at least two jobs for the third time period. Since the v_i have no predecessors, no list schedule is optimal for both m

FIG. 2. Generic $m, m + 1$ processor anomaly à la Graham.

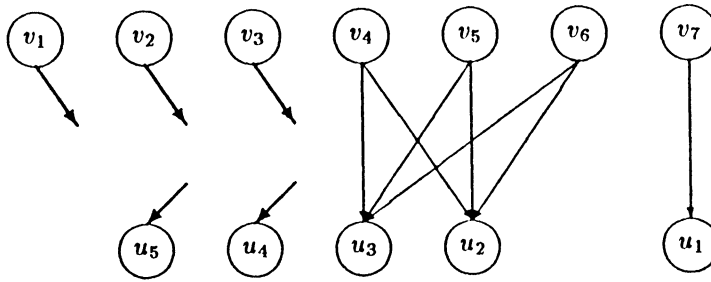


FIG. 3. Picture of a 12 node anomaly.

and $m + 1$ processors. In general this anomaly has only $3(m + 1)$ jobs. It implies for $m > 2$ that when the number of machines changes from m to $m + 1$ ($m + 1$ to m), the most robust schedule may be nonoptimal by a factor of $4/3$ ($5/4$). We suspect these ratios are as large as possible. When $m = 2$ we have only $7/6$ ($10/9$).

The idea that drives the preceding anomaly is that an optimal schedule must select at most a nodes to free up at least b jobs for the next period. This is strongly reminiscent of the NP-Complete Minimum Set Union problem (Instance: a collection C of sets over some ground set, integers a and b ; Question: are there b sets in C whose union has cardinality at most a ?). Thus, as a corollary, we get a very simple proof of the NP-completeness of the scheduling problem.

THEOREM. *Unit-Time Multiprocessor scheduling is NP-complete, even with a deadline of 3 and no precedence chains of length more than 2.*

Proof. For any instance (C, a, b) of Min Set Union, create the corresponding bipartite graph $G = (V, U, E)$ where each set in C corresponds to a node in U , V is the ground set, and an arc from V to U shows membership in a set. Pad the instance with sufficient elements contained in each or no set, C with sufficient sets containing all elements, and select m so that $|V| + |U| = 3m$; $a = m$; and $b = 2m - |V|$. Then the jobs can be scheduled in three periods if and only if the desired subcollection of sets exists. $\square$

Unfortunately, the proof gives no insight into the case of fixed m since Min Set Union is polynomial for fixed a .

REFERENCES

- [1] R. GRAHAM, *Bounds on the performance of scheduling algorithms*, in *Computer and Job-Shop Scheduling Theory*, E. G. Coffman, ed., John Wiley, New York, 1975.
- [2] J. K. LENSTRA AND A. H. G. RINNOOY-KAN, *Complexity of scheduling under precedence constraints*, *Oper. Res.*, 26 (1978), pp. 22–35.
- [3] J. ULLMAN, *NP-complete scheduling problems*, *J. Comput. System. Sci.*, 10 (1975), pp. 384–393.

RIVER ROUTING WITH A SMALL NUMBER OF JOGS*

TAI-CHING TUAN† AND S. LOUIS HAKIMI‡

Abstract. The one-layer wiring problem of providing a one-to-one connection between two sets of terminals that lie on two horizontal lines by means of wires that are in the forms of disjoint rectilinear curves on a unit grid (where one unit is the minimum spacing between two wires), is called the river routing problem. The problem has been widely studied. Here this problem is studied when the number of horizontal segments in each wire is at most 2, and provide an $O(n^3)$ time dynamic programming algorithm for finding the minimum separation between the two horizontal lines for n wires.

Key words. river routing, placement, PERT analysis, polynomial algorithm, optimization

AMS(MOS) subject classifications. 05, 68, 94

1. Introduction. Consider two sequences of increasing integers $A = (a_1, a_2, \dots, a_n)$ and $B = (b_1, b_2, \dots, b_n)$ that represent the coordinates of two sets of terminals (pins) on two parallel (horizontal) lines. The distance between these lines, denoted by s , is a positive integer and is a design variable called *separation*. A wire representing the net N_i , for $i = 1, 2, \dots, n$, must join the terminal at a_i to the terminal at b_i by means of a continuous rectilinear curve of total length $s + |a_i - b_i|$ on a unit grid (where one unit is the minimum spacing between two wires). The total length $s + |a_i - b_i|$ suffices to connect a_i and b_i because wires that are extended beyond the two end points do not help reduce separation. Furthermore, it is assumed that the coordinates of each horizontal or vertical segment of each wire are integers, and of course, no wire can touch another wire. The above wiring problem, denoted by the pair (A, B) , is sometimes called *river routing*.

Although the conditions for the problem are quite severe, it apparently occurs often in practice [11]. The river routing problem arises in practice when one wishes to provide connections (by means of wires) between two adjacent rectangular cells on a chip, where $(a_1, a_2, \dots, a_n)$ are the locations of the pins (of interest) on one cell and $(b_1, b_2, \dots, b_n)$ are the locations of the pins (of interest) on the other. When designing these cells with the ultimate intention of interconnecting them, the designer can often control the order of the pins on these cells such that the conditions of the above problem are satisfied. In Fig. 1(a), we depict such a situation where $n = 6$ and the connections are made by wires in the form of inclined straight line segments. This solution complicates the design rule checking because it leads to a violation of the minimum wire spacing requirement. In Fig. 1(b), we exhibit a solution to the same problem where the wire connecting a_i to b_i must be in the form of a rectilinear curve of total length $s + |a_i - b_i|$ for $i = 1, 2, \dots, n$, and s is the distance between the cells. In Fig. 1(c), we exhibit a solution to the same problem when no wire may have more than two horizontal segments. Finally, in Fig. 1(d), we exhibit a solution to the same problem when the upper cell is slid a distance of three units to the left. Note that in Figs. 1(b) and (c), the number of *tracks* is three. This implies that the cells must be four units of distance apart while in the solution of Fig. 1(d) the number of tracks is two, which means that the cells may be placed closer to each other. Furthermore, in Fig. 1(b) the wires $a_3 - b_3$ and $a_4 - b_4$ have three horizontal

* Received by the editors November 7, 1989; accepted for publication (in revised form) March 2, 1990. This work was supported by National Science Foundation grants DMC-8406854 and ECS-8511211.

† School of Electrical Engineering and Computer Science, University of Oklahoma, Norman, Oklahoma 73019.

‡ Department of Electrical Engineering and Computer Science, University of California, Davis, California 95616.

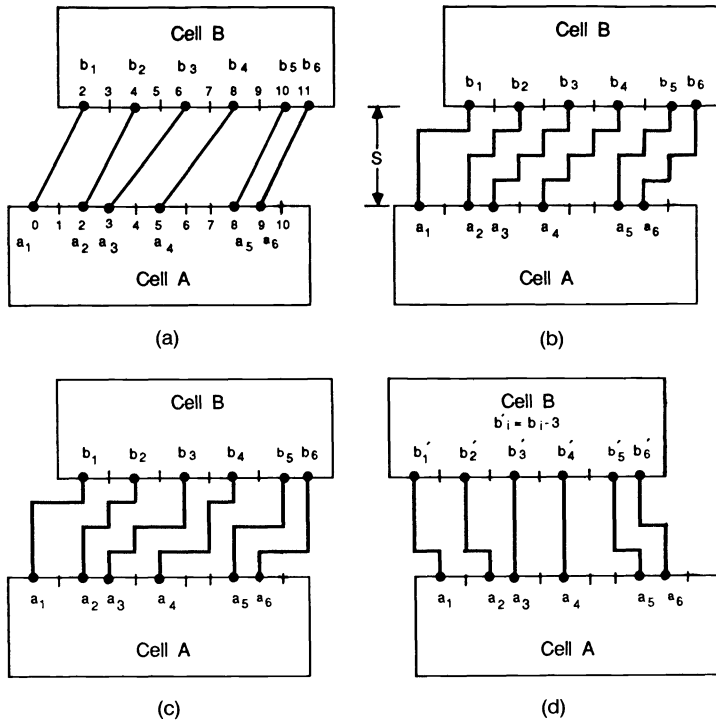


FIG. 1. (a) A layout with inclined lines; (b) a layout with three tracks; (c) a layout with three tracks in which every wire has at most two horizontal segments; and (d) a layout of $(A, B(-3))$.

segments (jogs) while in Fig. 1(c) each wire $a_i - b_i$ has at most two jogs. As corners often cause faults, a wiring in which each wire has fewer jogs is more desirable, particularly when it does not (substantially) increase the distance between cells. As this example indicates the above wiring problem is normally studied in conjunction with an optimization problem. We proceed with a formal definition of some of these problems and a brief review of the known results.

Given A and B and assuming momentarily that s is a fixed positive integer, the above wiring problem may not have a feasible solution. The reason for this is as follows. Suppose the ordinate of the line on which points in A lie is zero. If we use t horizontal tracks to solve our wiring problem, then we must have $s \geq (t + 1)$ for this solution to be feasible. For the remainder of this paper, let us assume that the terminals in A are fixed and that the terminals in B may be moved uniformly in a horizontal direction. More precisely, let w be an integer, called *offset*, and $B(w) = (b_1 + w, b_2 + w, \dots, b_n + w)$. The integer pair (s, w) is said to be feasible, if there is a feasible wiring of $(A, B(w))$ with separation s .

Given integer w , Dolev et al. [1], Siegel and Dolev [5], and Mirzaian [3] give $O(n)$ time algorithms for finding the minimum separation $s^*(w)$ for which $(s^*(w), w)$ is feasible.

For given separation s , let $W(s) = \{w | (s, w) \text{ is a feasible pair}\}$ and $\text{MINJOG}(s)$ be the minimum total number of jogs among all possible layouts with separation s . Algorithms of time complexity $O(n)$ for computing $W(s)$ are provided in [1], [3], [5]. Pinter [4] proposes an $O(\text{MINJOG}(s))$ time heuristic to achieve an s -separation layout

with small total jogs. In the worst case, the layout produced by Pinter's heuristic has at most $2 * \text{MINJOG}(s)$ total jogs [7]. Tuan and Teo [10] give an $O(n * s)$ time algorithm to find an s -separation layout with $\text{MINJOG}(s)$ total jogs. Let $(S, W) = \{(s, w) | (s, w) \text{ is feasible}\}$. Mirzaian [3] describes an $O(n^2)$ time algorithm for computing (S, W) . Mirzaian [3] also gives an $O(n^2)$ time algorithm for finding a feasible pair (s, w) , which achieves the shortest total wire length, $ns + \sum_{i=1}^n |a_i - b_i - w|$.

Let w^* be such that $s^*(w^*) \leq s^*(w)$ for all w . Mirzaian [3] gives an $O(n)$ time algorithm for finding $s^*(w^*)$ that is called *the optimum offset problem*. The optimum offset problem was previously solved in [1], [5] by $O(n \log n)$ time algorithms.

Let $d(w) = \max \{a_n, b_n + w\} - \min \{a_1, b_1 + w\}$. As before, let $s^*(w)$ be the smallest separation for which $(s^*(w), w)$ is feasible. Observe that $\alpha(w) = s^*(w)d(w)$ is the least area required to solve the wiring problem $(A, B(w))$. Mirzaian [3] gives an $O(n^2)$ time algorithm for finding an offset w^* that minimizes $\alpha(w)$. This problem was also attacked previously by Dolev et al. [1] and Leiserson and Pinter [2].

Siegel and Dolev [6] develop techniques that unify a variety of the above problems and give efficient solutions under extremely general conditions. In all of the above, the number of horizontal segments (called *jogs*) in each wire is permitted to be arbitrarily large. As was mentioned earlier in our example, there are practical reasons to place an upper bound J on the number of jogs in each wire. However, it appears that any of the above problems becomes very difficult under this additional constraint.

River routing for $J = 1$ is studied by Tuan and Hakimi [9] and Tuan [8]. Tuan and Hakimi [9] present a linear time algorithm for computing $s^*(w)$. They also describe an $O(n \log n)$ algorithm in the worst case and a linear time algorithm on the average to find an offset w that will achieve a layout with minimum number of jogs. This layout should, of course, satisfy $J = 1$. Tuan [8] gives a linear time algorithm for computing $W(s)$, an $O(n \log n)$ time algorithm for finding (S, W) , and an $O(n \log n)$ time algorithm with $O(n)$ space for finding $s^*(w^*)$, which is the minimum value of $s^*(w)$. In addition, $O(n \log n)$ time algorithms are described for finding an offset w^* that minimizes $\alpha(w)$ and for finding the shortest total wire length. Furthermore, it is shown that when J is fixed, $s \leq (\lfloor n/2 \rfloor + 1)$ and the upper bound is tight.

When $J = 2$ the river routing problem becomes much more difficult. In the next section, we assume $J = 2$ and provide an $O(n^3)$ time dynamic programming algorithm for finding $s^*(w)$.

2. River routing with two-jogs constraint. In this section, we consider the problem of computing s^* for the layout of (A, B) with at most two jogs in each net. We will assume that $b_i < a_i$ for $i = 1, 2, \dots, n$. This does not lead to a loss of generality because: (1) the case when $a_i > b_i$ for $i = 1, 2, \dots, n$ is just the reflexive mirror image of the previous case and can be dealt with in a similar manner; (2) if for some i and j , $b_i < a_i$ and $b_j > a_j$, we can partition the problem into smaller *parts* and solve each part separately where each part involves a set of indices, I , which is a maximal set of consecutive integers for which $b_i < a_i$ (or $b_j > a_j$); and (3) if for some i , $a_i = b_i$, then the wire corresponding to N_i has no horizontal segments and such nets may be ignored and wired as vertical lines after the remaining nets have been laid out.

We also assume that $b_{i+1} \leq a_i$ for $i = 1, 2, \dots, n - 1$. This also will not lead to a loss of generality, because if for some j , $b_{j+1} > a_j$, then we can lay out nets N_1 through N_j and then separately and independently lay out nets N_{j+1} through N_n .

For the purposes of our discussion, each net N_i is represented by the closed interval $[b_i, a_i]$ and in the Interval Representation Diagram (IRD) of our problem (A, B) , each

net N_i is represented by a horizontal line segment beginning with a point with x -coordinate b_i and ending with a point with x -coordinate a_i . Furthermore, in this diagram, we assume that the y -coordinate of N_i is i for $i = 1, 2, \dots, n$. For example, a layout problem may be described by $N_1 = [0, 2]$, $N_2 = [1, 5]$, $N_3 = [3, 7]$, $N_4 = [6, 9]$, $N_5 = [8, 10]$, and $N_6 = [9, 11]$. The IRD of this problem is shown in Fig. 2.1 (a). In any layout of (A, B) , each net N_i has either one or two horizontal segments. If, in a particular layout, $N_i = [b_i, a_i]$ has two horizontal segments (jogs) defined by nonempty subintervals $[b_i, c_i]$ and $[c_i, a_i]$, then we call c_i the "breakpoint" of N_i in this layout. (It should be noted that the former segment will fall in a higher track than the later segment.) If, on the other hand, in a layout, $N_i = [b_i, a_i]$ has exactly one horizontal segment, then in fact N_i (in this layout) has no breakpoint, but it is mathematically convenient to assume that in this case $c_i = a_i$. We now make a number of simple observations about an optimal layout, i.e., a layout for which s is the minimum possible. (1) $c_1 = a_1$; (2) $c_i < c_{i+1}$ for $i = 1, 2, \dots, n - 1$; (3) if c_i is known, then $c_{i+1} \in \{c_i + 1, a_i + 1\}$; (4) if $a_{i+1} = a_i + 1$ and $b_{i+1} \leq c_i$, then $c_{i+1} = c_i + 1$; and finally (5) if $b_{i+1} > c_i$, then $c_{i+1} = a_i + 1$. The proofs of the above observations are self-explanatory and may be constructed by merely drawing simple and appropriate diagrams.

Using the above observations, we see that $c_1 = a_1$, $c_2 = a_1 + 1$ and for $i = 2, 3, \dots, n - 1$, c_{i+1} may be computed in terms of c_i as follows:

$$(1) \quad c_{i+1} \in \begin{cases} \{c_i + 1\}, & \text{if } a_{i+1} = a_i + 1 \text{ and } b_{i+1} \leq c_i; \\ \{a_i + 1\}, & \text{if } b_{i+1} > c_i; \\ \{c_i + 1, a_i + 1\}, & \text{otherwise.} \end{cases}$$

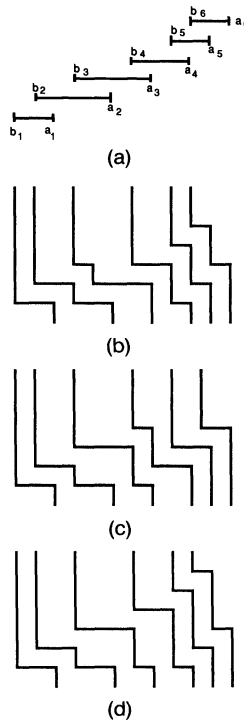


FIG. 2.1. (a) An IRD; (b) a layout with $s = 6$; (c) a layout with $s = 5$; and (d) a layout with $s = 7$.

A sequence $c_1, c_2, \dots, c_n$ of breakpoints is called feasible if it satisfies the above properties. To each sequence of breakpoints there is associated a unique (most compact) layout. (The term "most compact" refers to a layout that does not deliberately waste tracks.) Similarly, to each layout there is associated a sequence of breakpoints. Let C be the set of all sequences of breakpoints. Note that C has at most 2^{n-2} sequences. Although the cardinality of C is often far smaller than 2^{n-2} , the direct examination of all possible sequences in C for the determination of the optimal layout is often prohibitive. We will describe a dynamic programming approach that leads to an optimal solution in polynomial time.

Before that, we will examine the layout problem introduced by the example of Fig. 2.1(a). We note that $c_1 = a_1 = 2$, $c_2 = a_1 + 1 = 3$, $c_3 \in \{4, 6\}$. If we select $c_3 = 4$, then $c_4 = a_3 + 1 = 8$, but if we select $c_3 = 6$, then $c_4 \in \{7, 8\}$. As $a_5 = a_4 + 1$ and $a_6 = a_5 + 1$, the choice of c_4 determines the values of c_5 and c_6 uniquely. The above discussion implies that there are exactly three feasible sequences of breakpoints and these are $(2, 3, 4, 8, 9, 10)$, $(2, 3, 6, 7, 10, 11)$, and $(2, 3, 6, 8, 9, 10)$. The layouts corresponding to these breakpoint sequences are shown in Figs. 2.1(b)–2.1(d). The optimum layout is shown in Fig. 2.1(c) for which $s = 5$. Observe that if we eliminate the last two nets (i.e., N_5 and N_6), then the layout of Fig. 2.1(b) with $s = 4$ would be the best. This means that a simple greedy approach would not work, because even though the first four nets in the layout of Fig. 2.1(b) are laid out optimally, once the remaining two are laid out it will make this layout inferior to that of Fig. 2.1(c).

Let $FB = \{i \mid i \text{ is an integer } \geq 2 \text{ such that } a_i > a_{i-1} + 1\}$. We call FB the flexible breakpoint indices. Note that if, for integer j with $2 \leq j \leq n$, $j \notin FB$, then the breakpoint c_{j-1} determines c_j .

Let $FB^+ = FB \cup \{1, n\}$ and assume that members of FB^+ are listed in an increasing order, $1, i_1, i_2, \dots, i_k, n$. (Note that i_k may be equal to n in which case $|FB^+| = k + 1$.) In what follows we assume that $n > i_k$, but if $n = i_k$, slight and self-explanatory modifications of our procedure would produce the solution. For example, we may introduce a new net $N_{n+1} = [a_n, a_n + 1]$ and optimally lay out the augmented specification and then remove N_{n+1} from the solution. We will construct a directed graph (digraph) G with $(k + 2)$ vertices where vertices $v(1)$ and $v(n)$ correspond to net N_1 and N_n , and vertex $v(i_j)$ corresponds to net N_{i_j} , $1 \leq j \leq k$.

To define the edges of digraph G , we define a function $t: FB \cup \{1\} \rightarrow Z^+$ as follows. $t(1) = i_1$. (Note that i_1 is the smallest member of FB .) For $i \in FB$, let $t(i)$ be the largest positive integer k such that $a_{i-1} + k > b_{i-1+k}$. (Note that for any $k' < t(i)$, $a_{i-1} + k' > b_{i-1+k'}$.) For each $j \in FB^+ - \{1, i_1\}$, let j^* be the largest integer in FB that is less than j . Now we are ready to define the edge set of G . To begin with, the directed edge $(v(1), v(i_1))$ always belongs to G . For $i > 1$ the directed edge $(v(i), v(j))$ is in G when $i < j$ and $j^* < i + t(i)$.

Before we discuss the weights of the edges of the digraph G , we sketch what we plan to do. Through the above (acyclic) digraph we will seek an "optimal" path from $v(1)$ to $v(n)$ by a technique akin to PERT analysis. If this path has the vertex sequence $v(1), v(j_1), v(j_2), \dots, v(j_s), v(n)$, then it would mean that (in a minimum separation layout) the breakpoints on nets $j_1 (= i_1), j_2, \dots, j_s \in FB$ are equal to $(a_{j_1-1} + 1), (a_{j_2-1} + 1), \dots, (a_{j_s-1} + 1)$, while remaining breakpoints c_l , for $l \in \{1, 2, \dots, n\} - \{j_1, j_2, \dots, j_s\}$ are all determined as follows: (1) For $l = 1, 2, \dots, i_1 - 1$, $c_l = a_l$; (2) for $1 \leq \alpha \leq s - 1$ and integer l with $j_\alpha < l < j_{\alpha+1}$, if $b_l < c_{j_\alpha} + (l - j_\alpha) < a_l$, then $c_l = c_{j_\alpha} + (l - j_\alpha)$; otherwise $c_l = a_l$; and (3) for $j_s < l \leq n$, if $b_l < c_{j_s} + (l - j_s) < a_l$, then $c_l = c_{j_s} + (l - j_s)$; otherwise $c_l = a_l$.

Let $e = (v(i), v(j))$ be a (directed) edge in G . We first define the values of $M(e)$, $L(e)$, and $\text{Sgn}(e)$ and then explain the significances of these values.

$$(2) \quad (M(e), L(e), \text{Sgn}(e)) = \begin{cases} (i_1 - 1, i_1 - 1, +), & \text{if } i = 1, (\text{consequently } j = i_1); \\ (j - i, j - i, +), & \text{if } i > 1, j \leq n \text{ and } j < i + t(i); \\ (t(i) - 1, j - i + 1, -), & \text{if } i > 1, j \leq n, j \geq t(i) + i, \\ & \text{but } j^* < t(i) + i. \end{cases}$$

We will now explain what these triplets represent. We first note that N_1 has one jog and is laid out on track one. We consider $e = (v(1), v(i_1))$; the triplet $(i_1 - 1, i_1 - 1, +)$ represents the fact that, in an optimum layout, net N_{i_1} is the first net with two jogs (i.e., two horizontal segments). The left segment of this net will be $(i_1 - 1)$ tracks above N_1 (i.e., $L(e) = i_1 - 1$) and the number of tracks above N_1 used in the layout for nets $N_1, N_2, \dots, N_{i_1}$ is $(i_1 - 1)$ (i.e., $M(e) = i_1 - 1$). $\text{Sgn}(e) = +$ here implies that $M(e)$ and $L(e)$ are achieved at the same horizontal segment. This situation is depicted in Fig. 2.2 where i_1 is assumed to be 4. We next consider the weight of edge $e = (v(i), v(j))$ when $i > 1$ and $j \leq n$. First note that because e belongs to G it means that i and $j \in FB^+$. Consequently, in any layout that corresponds to a path in G that passes through this edge, the breakpoints on N_i and N_j are at $(a_{i-1} + 1)$ and $(a_{j-1} + 1)$, respectively. The breakpoint c_{i+k} for net N_{i+k} with $1 \leq k < j - i$ is at $a_{i-1} + k + 1$ provided $t(i) > k$. If $t(i) \leq k$, then $c_{i+k} = a_{i+k}$. In such a layout, the breakpoint of net N_i (as well as N_j) is known, the right horizontal segment of N_i is laid out on track 1, and if we know the track $T(i)$ on which the left horizontal segment of N_i is laid out, then layout of the nets $N_{i+1}, \dots, N_j$ is completely determined. The integer $M(e)$ always represents the number of additional tracks above $T(i)$ one needs to lay out nets $N_{i+1}, \dots, N_j$. When $\text{Sgn}(e) = +$, $L(e)$ represents the number of tracks above $T(i)$ where the horizontal segment of N_j to the left of the breakpoint of N_j is laid out. This situation happens when $j < t(i) + i$ and is depicted in Fig. 2.3. In particular, if $j = n = i + t(i) - 1$, then $\text{Sgn} = +$ and

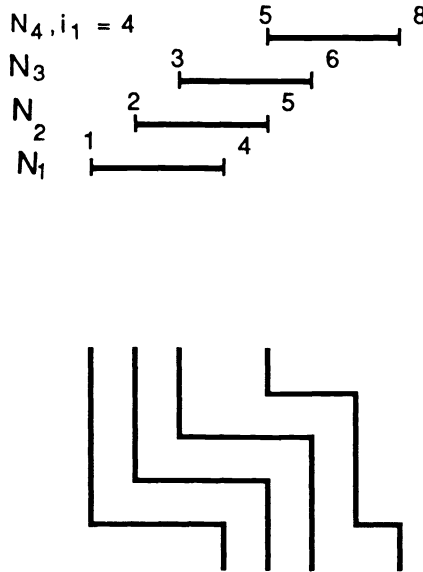


FIG. 2.2. An IRD and its layout.

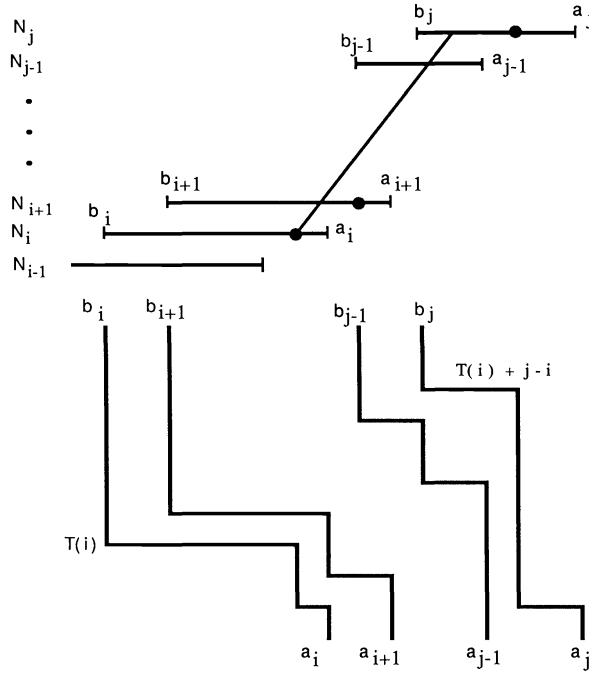


FIG. 2.3. Part of an IRD with $j < t(i) + i$ and its layout, inclined line is used to compute $t(i)$ and the marked points represent flexible breakpoints.

$M(e) = L(e) = n - i$. When $\text{Sgn}(e) = -$, $L(e)$ represents the track level (not above $T(i)$) on which the horizontal segment of N_j to the left of the breakpoint of N_j is laid out. This occurs when $j \geq t(i) + i$. Two cases when $\text{Sgn}(e) = -$ and $j < n$ are depicted in Figs. 2.4 and 2.5. In particular, we have in Fig. 2.4 the case when $j = t(i) + i$, and in Fig. 2.5 the case when $j > t(i) + i$ but $j^* \leq t(i) + i - 1$. One layout for $j = n$ is demonstrated in Fig. 2.6. Note that in this figure $i_k < i + t(i)$, $\text{Sgn} = -$, and therefore $L(e) = n - i + 1$ represents the level of the horizontal segment of N_n to the left of the breakpoint of N_n (which is at a_n).

We now must seek a path from $v(1)$ to $v(n)$ that leads to a layout with the minimum number of tracks. Recall that any path from $v(1)$ to $v(n)$ determines uniquely a layout and more importantly we will show later that any compact layout corresponds to such a path. The algorithm involves a labeling process. Each vertex $v(j)$ is labeled by an array $L_{v(j)}[1 \cdots j]$. $L_{v(1)}[1] = 1$, and initially $L_{v(j)}[i] = n + 1$ for $1 \leq i \leq j$ and $j \neq 1$. If i and i' , respectively, denote the number of tracks used in a partial layout for $N_1, N_2, \dots, N_j$ and the track on which the horizontal segment to the left of the breakpoint on net N_j lies, then $L_{v(j)}[i] = i'$ and $i \geq i' \neq n + 1$. For notational convenience, let us represent the array $L_{v(j)}$ by a list of 2-tuples arranged in the ascending order of the first component. So if $L_{v(j)}[i] = i'$, then the 2-tuple (i, i') is in the list representing $L_{v(j)}$. We will prepare a list representation of $L_{v(j)}$ only when $v(j)$ becomes a source (i.e., a vertex whose indegree is zero) during the application of the algorithm. This list representation will be denoted by $L'_{v(j)}$. The labeling proceeds from $v(1)$ to all neighbors (immediate descendants) of $v(1)$ in a breadth-first-search manner (in this case just $v(i_1)$). Once all neighbors of $v(1)$ are labeled, $v(1)$ is removed from G . Because the graph is acyclic this will generate a new vertex which is a source. If $v(j)$ is a source and (p_1, p_2) and $(q_1, q_2) \in L'_{v(j)}$ with $p_1 < q_1$ and $p_2 \leq q_2$, then (q_1, q_2) is called a redundant 2-tuple in the label. For the

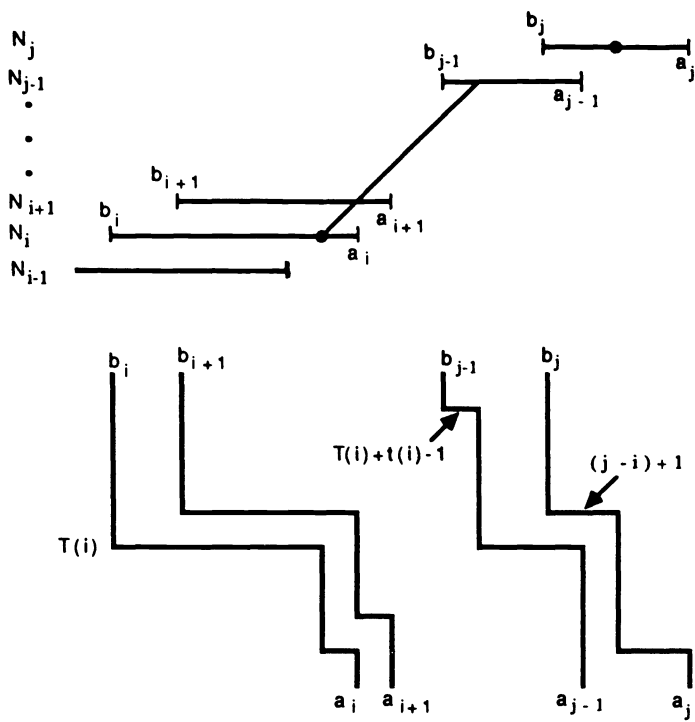


FIG. 2.4. Part of an IRD with $j = t(i) + i$ and its layout.

source $v(j)$, redundant 2-tuples can be singled out and excluded from the list representation and the resulting final list representation $L'_{v(j)} = \langle (N_1(v(j)), T_1(v(j))), (N_2(v(j)), T_2(v(j))), \dots, (N_s(v(j)), T_s(v(j))) \rangle$ has the property that $N_1(v(j)) <$

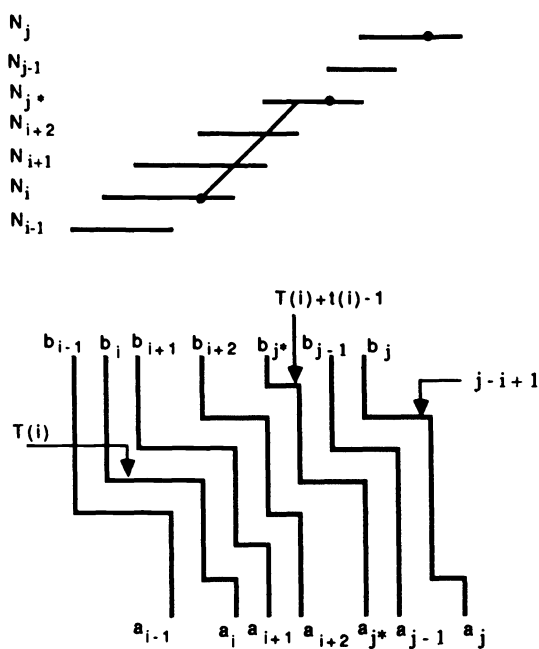
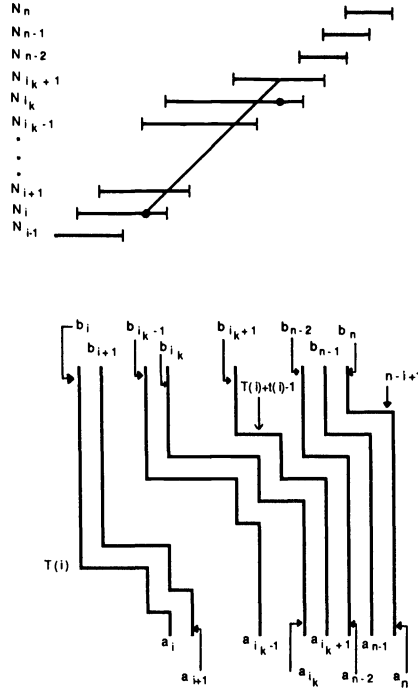


FIG. 2.5. Part of an IRD with $j > t(i) + i$ and its layout.

FIG. 2.6. Part of an IRD with $n > t(i) + i$ and its layout.

$N_2(v(j)) < \dots < N_s(v(j))$ and $T_1(v(j)) > T_2(v(j)) > \dots > T_s(v(j))$. This pruning of the list can be done in $O(n)$ time. For example, if the original list is $\langle (6, 4), (8, 4), (9, 2), (10, 3) \rangle$, then the resulting final list is $\langle (6, 4), (9, 2) \rangle$. This final list $L'_{v(j)}$ of the source is used to recompute the labels of its immediate descendants; this source is then removed and another source is selected. The process continues until the remaining graph has exactly one vertex $v(n)$. Let us suppose that at a certain stage of the algorithm, the list label of $v(j)$ is $L'_{v(j)} = \langle (N_1(v(j)), T_1(v(j))), (N_2(v(j)), T_2(v(j))), \dots, (N_s(v(j)), T_s(v(j))) \rangle$ having the property that $N_1(v(j)) < N_2(v(j)) < \dots < N_s(v(j))$ and $T_1(v(j)) > T_2(v(j)) > \dots > T_s(v(j))$. Each 2-tuple $(N_i(v(j)), T_i(v(j)))$ in $L'_{v(j)}$ represents a label obtained for $v(j)$ by having traversed a path in G , say $P_i(v(j))$, from $v(1)$ to $v(j)$. As before, path $P_i(v(j))$ uniquely defines a layout for nets $N_1, N_2, \dots, N_j$. The integers $N_i(v(j))$ and $T_i(v(j))$, respectively, denote the number of tracks used in the layout (implied by $P_i(v(j))$) and the track on which the horizontal segment to the left of the breakpoint on net N_j lies. This basic idea is used later to complete an inductive proof of the correctness of the algorithm.

We are now prepared to discuss the basic step in the labeling algorithm. Recall that we always use a source to modify the labels of its descendants immediately after a source has received its final label. We begin with $L'_{v(1)} = \langle (1, 1) \rangle$ and $L'_{v(i)} = \langle \rangle$ for $i \neq 1$.

Let us assume that at a certain stage $v(i)$ is a source and its list label is $L'_{v(i)} = \langle (N_1(v(i)), T_1(v(i))), (N_2(v(i)), T_2(v(i))), \dots, (N_l(v(i)), T_l(v(i))) \rangle$. Assume that the directed edge $(v(i), v(j))$ whose weight is (M, L, Sgn) is used to update $L'_{v(j)}$.

For $k = 1, 2, \dots, l$, compute $(N'_k(v(j)), T'_k(v(j)))$ as follows:

$$N'_k(v(j)) = \begin{cases} \max \{ N_k(v(i)), T_k(v(i)) + M \}, & \text{if Sgn} = + \\ \max \{ N_k(v(i)), T_k(v(i)) + M, L \}, & \text{if Sgn} = - \end{cases}$$

and

$$T_k^t(v(j)) = \begin{cases} T_k(v(i)) + M, & \text{if Sgn} = + \\ L, & \text{if Sgn} = -. \end{cases}$$

To update the array $L_{v(j)}$, we use the following criterion. If $L_{v(j)}[N_k^t(v(j))] > T_k^t(v(j))$, then $T_k^t(v(j))$ would be the new value for $L_{v(j)}[N_k^t(v(j))]$.

The above step represents the basic step of the algorithm that must be carried out for each edge $(v(i), v(j))$ where $v(i)$ has become source and its final list label is determined. Then $v(i)$ is eliminated, another source is selected, its final list label is determined, and the above step is repeated. This algorithm terminates until only $v(n)$ is left. Assume then that $L_{v(n)} = \langle (N_1(v(n)), T_1(v(n)), \dots) \rangle$. We state the results of this section by the following theorem.

THEOREM 1. (a) $N_1(v(n))$ is the least number of tracks needed to lay out (A, B) with at most two jogs per net; (b) the path from $v(1)$ to $v(n)$, which gives rise to the first member of $L_{v(n)}$, provides a layout with $N_1(v(n))$ tracks in which each net has at most two jogs; (c) the time complexity of this algorithm is $O(|E(G)|n)$.

Remarks. Once the basic idea is understood, the proof of this theorem is not difficult but it is quite long. To help the reader understand the basic idea in our algorithm, we present two examples. The first example is relatively simple while the second one is somewhat complicated.

The IRD of the first example is shown in Fig. 2.7(a) where the inclined lines are used to determine $t(i)$'s. As it is indicated in Fig. 2.7(a) there are three flexible breakpoints on nets N_2, N_3 , and N_4 and $t(2) = 2, t(3) = 2, t(4) = 3$. The digraph is shown in Fig. 2.7(b) where the weights of the edges are determined in accordance with (2). The labels of the vertices are obtained by the application of the algorithm. Note that the length of the list representation for $L_{v(i)}$ is 1 for all vertices. The optimum path from $v(1)$ to $v(6)$, which leads to the least number of tracks being 4, (or separation of 5) is the path $v(1)v(2)v(3)v(6)$. The layout corresponding to this path is shown in Fig. 2.7(c). This is in fact the same problem as the one in Fig. 2.1.

The IRD of the second example is shown in Fig. 2.8(a). The digraph is shown in Fig. 2.8(b) where the edges are labeled according to (2). The labels of the vertices are computed by the algorithm. Note that the lengths of the list representations for

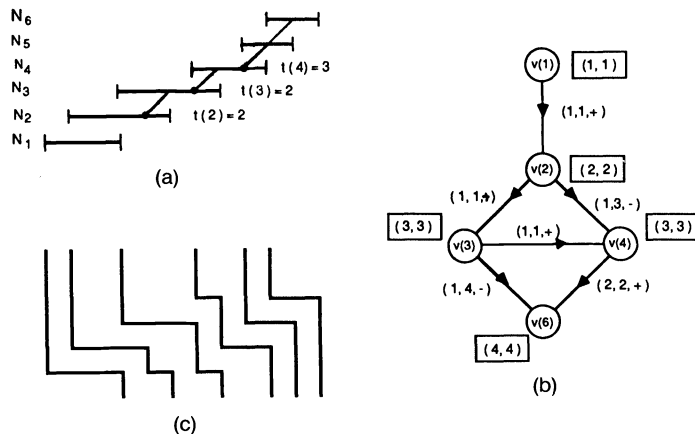


FIG. 2.7. (a) An IRD; (b) digraph constructed from the IRD; (c) an optimum layout corresponding to the path $v(1)v(2)v(3)v(6)$ in the digraph.

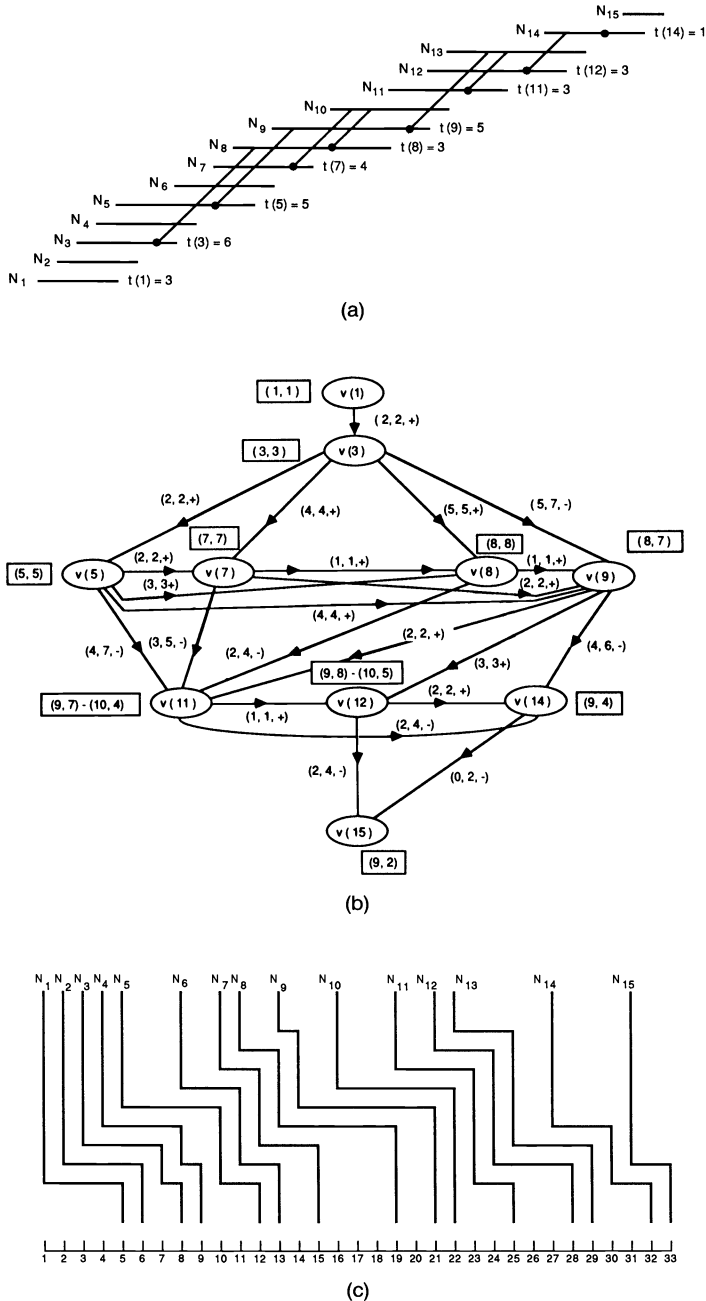


FIG. 2.8. (a) An IRD; (b) digraph constructed from the IRD; and (c) an optimum layout corresponding to the path $v(1) v(3) v(5) v(11) v(14) v(15)$.

L'_v s are all one except for two vertices that are two. The optimum path is $v(1) v(3) v(5) v(11) v(14) v(15)$ leading to the least number of tracks being 9. The corresponding layout is shown in Fig. 2.8(c).

We will now provide a sketch for the proof of Theorem 1. We trust that the reader can complete the details of the proof.

Sketch of the proof of Theorem 1. We first show that for any given *compact layout* there exists a layout that corresponds to a path in G from $v(1)$ to $v(n)$ that uses no more tracks than the given layout. To begin with, in any compact layout, the nets N_1 to N_{i_1} must be laid out exactly in the manner suggested by the edge $(v(1), v(i_1))$ in G ; that is, the breakpoint $c_{i_1} = a_{i_1-1} + 1$, and $c_i = a_i$ for $i = 1, 2, \dots, i_1 - 1$. Suppose, in the layout, the indices of the remaining breakpoints being in $FB \cup \{n\}$ are $j_1, j_2, \dots, j_s$, with $j_s = n$. Suppose now that the edges $(v(1), v(i_1)), (v(i_1), v(j_1)), \dots, (v(j_{\alpha-1}), v(j_{\alpha}))$ belong to G but $(v(j_{\alpha}), v(j_{\alpha+1}))$ does not. We observe that if $j_{\alpha+1} < t(j_{\alpha}) + j_{\alpha}$, then edge $(v(j_{\alpha}), v(j_{\alpha+1}))$ would belong to G . So we must have $j_{\alpha+1} \geq t(j_{\alpha}) + j_{\alpha}$ and $j_{\alpha+1}^* > t(j_{\alpha}) + j_{\alpha} - 1$. An example of an IRD that satisfies this condition is shown in Fig. 2.9(a). As the flexible breakpoint on $N_{j_{\alpha+1}}$ is not used, it means that $c_{j_{\alpha}} = a_{j_{\alpha}-1} + 1$; $c_{j_{\alpha}+k} = c_{j_{\alpha}} + k$ for $1 \leq k \leq t(j_{\alpha}) - 1$; the breakpoints on nets $N_{j_{\alpha}+t(j_{\alpha})}, \dots, N_{j_{\alpha+1}-1}$ are at $a_{j_{\alpha}+t(j_{\alpha})}, \dots, a_{j_{\alpha+1}-1}$ and the breakpoint on $N_{j_{\alpha+1}}$ is at $(a_{j_{\alpha+1}-1} + 1)$. The corresponding layout is shown in Fig. 2.9(b). We need to introduce a measure of “goodness” of this layout up to net $N_{j_{\alpha+1}}$ so that we can compare it to a layout that corresponds to a path in G from $v(1)$ to $v(j_{\alpha+1})$. This measure involves two quantities: (1) the total number of tracks used so far, i.e., $N(v(j_{\alpha+1}))$ and (2) the position of the last net $N_{j_{\alpha+1}}$ in the layout; in particular, the more to the left and/or to the lower track on which the horizontal segment to the left of the breakpoint on $N_{j_{\alpha+1}}$ lies, the more desirable. As the breakpoint on $N_{j_{\alpha+1}}$ is definitely at $a_{j_{\alpha+1}-1} + 1$, the only issue is the position of the horizontal segment of $N_{j_{\alpha+1}}$ to the left of its breakpoint, i.e., the quantity $T(v(j_{\alpha+1}))$.

Let us now examine an alternative to the layout of Fig. 2.9(b). As $j_{\alpha+1}^* > t(j_{\alpha}) + j_{\alpha} - 1$, assume (by induction) that $(j_{\alpha+1}^*)^* \leq t(j_{\alpha}) + j_{\alpha} - 1$ (in the example of Fig.

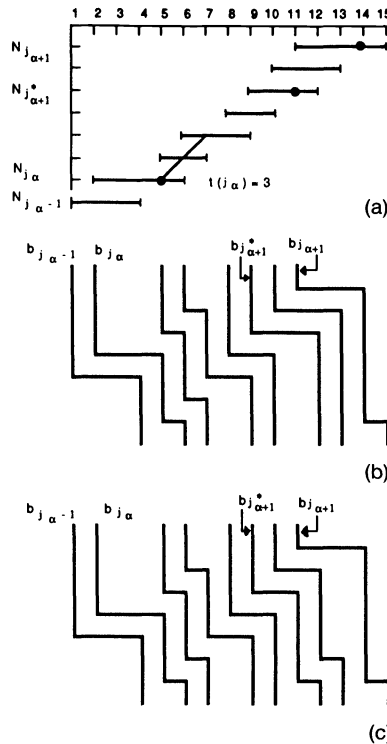


FIG. 2.9. (a) Part of an IRD with flexible breakpoints on $N_{j_{\alpha}}$, $N_{j_{\alpha+1}}$, and $N_{j_{\alpha+1}^*}$; (b) a layout in which the breakpoint on $N_{j_{\alpha+1}}$ is at $a_{j_{\alpha+1}}$; and (c) a layout in which the breakpoint on $N_{j_{\alpha+1}}$ is at $a_{j_{\alpha+1}-1} + 1$.

2.9(a), $(j_{\alpha+1}^*)^* = t(j_\alpha) + j_\alpha - 1$, then edge $(v(j_\alpha), (v(j_{\alpha+1}^*)))$ belongs to G . Similarly, edge $(v(j_{\alpha+1}^*), v(j_{\alpha+1}))$ belongs to G . The layout corresponding to the traversal of these two edges from $v(j_\alpha)$ to $v(j_{\alpha+1})$ is shown in Fig. 2.9(c). Observe that in this layout the position of $N_{j_{\alpha+1}}^*$ is more desirable (more to the left) than the position of the same net in the layout of Fig. 2.9(b), although $N(v(j_{\alpha+1}^*))$ and $T(v(j_{\alpha+1}^*))$ are the same for both layouts. In the example, the slight advantage (being more to the left at this stage) happens to become irrelevant by the time the next $N_{j_{\alpha+1}}$ is laid out, i.e., $N(v(j_{\alpha+1}))$ and $T(v(j_{\alpha+1}))$ are the same for both layouts. However, if, for example, $N_{j_{\alpha+1}}$ would have been [13, 15] instead of [11, 15], the slight advantage for the layout of Fig. 2.9(c) up to $N_{j_{\alpha+1}}^*$ would have lead to significant advantage in the $T(v(j_{\alpha+1}))$ by the time $N_{j_{\alpha+1}}$ was laid out. In summary, in this case, using the breakpoint on $N_{j_{\alpha+1}}^*$ cannot be a disadvantage. That is, there is a layout that corresponds to a path from $v(1)$ to $v(n)$ that is as good as or better than any other layout.

We next need to prove that our labeling algorithm produces a path in G that corresponds to a layout with the least number of tracks (among all layouts that correspond to paths in G). This part of the proof is long but also a direct consequence of our labeling algorithm and therefore will not be elaborated upon. $\square$

The time complexity of our algorithm can be determined as follows. The construction of the digraph takes $O(n^2)$ time. As the digraph is acyclic each edge $(v(i), v(j))$ is used exactly once. Thus the main step of the algorithm is repeated $|E(G)|$ times. The length of $L_{v(i)}$ for each vertex is at most n . (The length of the list representation for each $L_{v(i)}$ appears to be very short, but we have not been able to prove this in general.) The main step takes $O(n)$ time; this is because up to n values of $(N'_k(v(j)), T'_k(v(j)))$ must be computed and updating $L_{v(j)}$ for each such pair takes $O(1)$ time. Also, it takes $O(n)$ time to get the final list label for each node when selected as new source. So, it takes $O(|E(G)|n)$ time to compute $N_1(v(n))$. To find the path that leads to the number of tracks being $N_1(v(n))$, we must backtrack from $v(n)$. If we keep track of the immediate *father* $v(i)$ of each member $(N_r(v(j)), T_r(v(j)))$ in $L'_{v(j)}$ and the member $(N_r(v(i)), T_r(v(i)))$ in $L'_{v(i)}$ that leads to the pair $(N_r(v(j)), T_r(v(j)))$, then the backtracking can be done in linear time.

REFERENCES

- [1] D. DOLEV, K. KARPLUS, A. SIEGEL, A. STRONG, AND J. D. ULLMAN, *Optimal wiring between rectangles*, Proc. 13th ACM Symposium on Theory of Computing, 1981, pp. 312–317.
- [2] C. E. LEISERSON AND R. Y. PINTER, *Optimal placement for river routing*, SIAM J. Comput., 12 (1983), pp. 447–462.
- [3] A. MIRZAIAN, *River routing in VLSI*, J. Comput. System Sci., 34 (1987), pp. 43–54.
- [4] R. Y. PINTER, *The impact of layer assignment methods on layout algorithms for integrated circuits*, Ph.D. Thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, Cambridge, MA, 1982.
- [5] A. SIEGEL AND D. DOLEV, *The separation for general single-layer wiring barriers*, Carnegie-Mellon Conference on VLSI Systems and Computations, Computer Science Press, Rockville, MD, 1981, pp. 143–152.
- [6] ———, *Some geometry for general river routing*, SIAM J. Comput., 17 (1988), pp. 583–605.
- [7] K. H. TEO AND T. C. TUAN, *Performance analysis of greedy heuristic to find a minimum total-jogs layout for river routing*, Technical Report, School of EECS, University of Oklahoma, Norman, OK, 1990.
- [8] T. C. TUAN, *On optimal single jog river routing*, IEEE Trans. Comput., to appear.
- [9] T. C. TUAN AND S. L. HAKIMI, *Single jog river routing*, Proc. 25th Annual Allerton Conference, 1987, pp. 373–374.
- [10] T. C. TUAN AND K. H. TEO, *On river routing with minimum number of jogs*, IEEE Trans. CAD, to appear.
- [11] J. D. ULLMAN, *Computational Aspects of VLSI*, Computer Science Press, Rockville, MD, 1984.