

NEURAL NETWORKS, ACYCLIC ORIENTATIONS OF THE HYPERCUBE, AND SETS OF ORTHOGONAL VECTORS*

PIERRE BALDI†

Abstract. Recent models in the theory of neural networks suggest the possibility of constructing new combinatorial invariants by associating to families of subsets of an n -set orientations of the hypercube via “energy functions.” Here, we restrict ourselves to the Hopfield model and its higher-order versions with families of orthogonal binary vectors and homogeneous polynomial functions, corresponding to sums of outerproducts of degree d , and investigate several properties of the corresponding orientations. In particular, the stability of the vectors in the family and of those in the orthogonal space is analyzed. The existence of significant differences of behavior according to the congruences modulo 4 of d is shown.

Key words. neural networks, orientations, orthogonal vectors, hypercube, Hebbian learning

AMS(MOS) subject classifications. 05, 94, 68, 92

1. Introduction. Motivated principally by models [9], [13] in the theory of neural networks and also marginally by some questions regarding the structure of sets of orthogonal binary vectors, we investigate a special case of the following general combinatorial problem. Let $M^1, \dots, M^k$ be a family of k subsets of an n -elements set. Alternatively, these can be viewed as k n -dimensional vectors of $(0, 1)$ or $(+1, -1)$ coordinates. Assume we have an injective real valued function (a function into a totally ordered set would suffice):

$$E\langle M^1, \dots, M^k \rangle(X): \mathbb{R}^n \rightarrow \mathbb{R}$$

which is, in some sense, “characteristic” of the given family. Then, for any graph G embedded in $\mathbb{R}^n$, we can construct an acyclic orientation $E\langle M^1, \dots, M^k \rangle(G)$ by

$$X \rightarrow Y \quad \text{iff} \quad E\langle M^1, \dots, M^k \rangle(X) > E\langle M^1, \dots, M^k \rangle(Y).$$

The question is then to relate properties of the orientation to properties of the original family. It may be useful to think of E as an “energy function” and of the acyclic orientation as a “landscape” determined on G by E . Our main purpose here is to explore such a construction in the special case where $M^1, \dots, M^k$ are k orthogonal vectors of $(+1, -1)$ coordinates and $G = H^n = \{+1, -1\}^n$. The previous description is rather vague and will be made precise in § 2 with the introduction of a class of neural network type of automata based on polynomial energy functions. In § 3, we shall derive several preliminary results needed to investigate the linear and quadratic case in § 4, and the general case in § 5.

2. Neural networks. There exists an extensive recent literature ([12] and references therein) on neural networks. Here we shall only review the basic concepts needed for the rest of this article. As defined in [9], a neural network is a stochastic automaton made of n interconnected units, each one being capable of two states: $+1$ and -1 . The state of the automaton is therefore described by a vector $X = (x_1, \dots, x_n)$ on H^n . The interconnections among the units form a weighted simple graph which can be described by a matrix (α_{ij}) of real coefficients with $\alpha_{ij} = \alpha_{ji}$ and $\alpha_{ii} = 0$. In addition, every unit i has its own “threshold” t_i . The evolution rule of the network is as follows: randomly and asyn-

* Received by the editors March 9, 1987; accepted for publication (in revised form) October 9, 1987.

† Department of Mathematics, University of California, San Diego, La Jolla, California 92093.

chronously each unit updates its own state. If x_i^- and x_i^+ represent, respectively, the state of unit i before and after its updating step, then

$$x_i^+ = \text{sgn} \left(\sum_{j=1}^n \alpha_{ij} x_j - t_i \right).$$

If the linear input minus the threshold happens to be zero, we adopt the convention $x_i^+ = x_i^-$. This automaton possesses a fundamental emergent property: independently of the starting state and of the sequence of updating steps, it always converges to a stable state. Indeed, if we consider the function

$$E(X) = -\frac{1}{2} \sum_{i,j} \alpha_{ij} x_i x_j + t_i x_i$$

then, with the obvious notation

$$E^+ - E^- = -(x_i^+ - x_i^-) \left(\sum_{j=1}^n \alpha_{i,j} x_j - t_i \right).$$

Therefore $E^+ - E^- \leq 0$ and since there are only 2^n possible configurations, the network must reach a halting state.

The dynamical behavior of the system is based on its energy function E and a completely equivalent description can be given directly in the space of states H^n . Orient the edges of the hypercube as described in the Introduction (here E is not necessarily injective, but in the generally rare case where $E(X) = E(Y)$ we can decide to delete the edge (X, Y)). Starting from any vertex X of H^n , choose at random an edge (X, Y) . Then move to Y if and only if $E(X) > E(Y)$. In this form, it is clear that the energy function need not be quadratic and in fact any function from H^n to a partially ordered set such that the images of neighboring vertices are comparable would suffice. In the rest of this paper we shall restrict ourselves to functions that are special polynomials in n variables.

A *stable state* of a neural automaton can be assimilated to a point of the oriented hypercube with outdegree equal to 0. It can also be seen as a local minimum of the corresponding energy function restricted to H^n . A stable state is *weakly stable* if the local minima is not strict (the indegree is not equal to n) or equivalently if there exists a state at Hamming distance one with the same energy value. A state will be called *critical* if it is a critical point of the energy function when restricted to the sphere of radius $\sqrt{n}$. Finally, if X is a stable state, the *radius* $R(X)$ of its basin of attraction is the largest Hamming distance such that if the network is started in any state Y contained in the ball $B(X, R(X))$ then it will always converge to X regardless of the order in which the units are updated. Obviously, a weakly stable point has a radius equal to zero.

The original Hopfield model and its generalizations can be studied under two very different regimes. The first possibility is when the coefficients of the energy function are chosen “at random” (see, for instance, [6]), as in statistical mechanical models of disordered systems. The second regime, which is the one of interest here, is when the energy function is carefully constructed, so that special classes of computations can be mapped onto the trajectories of the network. This map has been partially investigated for optimization [10], [11] (or local improvement) and nearest neighbor search (or error correction-associative memory) type of computations [9]. In the latter, a subset of the stable states can be seen as “codewords.” If X is such a stable state, any Y in $B(X, R(X))$ will be decoded or associated to X by the dynamical evolution of the network.

A typical problem then is: given a set of k vectors $M^1, \dots, M^k$, how do we construct an energy function such that the given vectors are stable states of the corresponding

neural network, or equivalently local minima of the function over the hypercube? For this purpose, several strategies have been proposed (see [5] for a brief review). The most basic construction is defined by

$$\alpha_{ij} = \sum_{s=1}^k M_i^s M_j^s \quad \text{and} \quad t_i = 0$$

and is called Hebb's rule or the sum of outerproducts rule. Its justifications can be found in Hebb's original work [8], in simple statistical considerations as in [9], and in combinatorial considerations as in [5] or Proposition 3.1 herein. This construction is well suited for vectors that are uncorrelated, i.e., orthogonal or chosen at random [14].

We shall not be concerned here with the questions of effectiveness of such automata in computations nor with their relevance to neurobiological information processing. Our purpose is to investigate a generalization of the Hebb's construction and study it exclusively in the case where the vectors are orthogonal. We would like to derive exact analytical results to complement some of those that appear in the literature, based on simulations, asymptotic estimates, or statistical mechanical arguments [2]. As was already pointed out, in addition to the obvious relation to neural networks models, this approach may be of intrinsic combinatorial interest. The structure of sets of orthogonal vectors is not very well understood, as exemplified by the still open Hadamard conjecture that n orthogonal vectors of coordinates ± 1 exist if and only if n is a multiple of 4. It is not known how to characterize sets of orthogonal vectors that are saturated (i.e., that cannot be enlarged). For instance, an intriguing result of Verheiden [15], for which a completely theoretical proof has not been found yet, states that any set of k orthogonal vectors with $k > n - 8$ can always be completed into a Hadamard matrix. We do not address these questions here. However, they seem to suggest that it may be useful to find combinatorial invariants for sets of orthogonal vectors.

From now on, our setting is as follows. Our vectors are column vectors and, if n is not specified, “(++++-----)” typically denotes a vector with 50 percent ones at the beginning. If I is a subset of $\{1, \dots, n\}$ and $X = (x_1, \dots, x_n)^t$, we use the notation $X^I = \prod_{i \in I} x_i$. Let d be an integer satisfying $1 \leq d \leq n$. Let $M^1, \dots, M^k$ be k orthogonal vectors of dimension n of ± 1 coordinates. For any d -subset I of $\{1, \dots, n\}$ define

$$(1) \quad \alpha_I = \sum_{s=1}^k (M^s)^I.$$

Consider the homogeneous form of degree d :

$$(2) \quad E^d \langle M^1, \dots, M^k \rangle (X) = E^d(X) = - \sum_{|I|=d} \alpha_I X^I.$$

We shall refer to this construction as the *sum of outerproducts (or the generalized Hebb's rule) of degree d* . We are interested in properties of the acyclic orientation it induces on H^n , especially whether the vectors $M^1, \dots, M^k$ are indeed stable, and so forth. To alleviate notation and when the context is clear, we may drop the specification of the degree or of the list of vectors under consideration.

The energy function in (2) can be also written as

$$(3) \quad E^d(X) = - \frac{1}{d} \sum x_i I_i^d(X) = - \frac{1}{d} X \cdot I^d(X)$$

where

$$(4) \quad I_i^d(X) = \sum_{i \in I} \alpha_I X^{I - \{i\}}.$$

$I_i^d(X)$ can be seen as the “input” into unit i , and it is easy to check that a state is stable if and only if $x_i I_i^d(X) \geq 0$ for any component i . If, in addition for some i , $I_i^d(X) = 0$, then X is weakly stable, and X and its neighbor along the i coordinate have the same energy value. If we write the Lagrangian of E^d constrained to the sphere of radius $\sqrt{n}$ as

$$(5) \quad L(X) = E^d(X) + \lambda \left(\sum_{i=1}^n x_i^2 - n \right),$$

then a point is critical if and only if for any i

$$(6) \quad I_i^d(X) = 2\lambda x_i \quad \text{or} \quad I_i^d(X) = 2\lambda X.$$

Finally, notice that $E^d(-X) = (-1)^d E^d(X)$. In particular, if X is critical, so is $-X$. If d is even, X and $-X$ tend to have the same behavior, while if d is odd, they tend to have opposite behavior.

3. Preliminaries and combinatorial lemmas. For our purposes it is natural to introduce the action of the group G of the isometries of the hypercube for the Hamming distance, which is generated by the maps that permute two components and maps that invert the sign of one component. It can be written as the wreath product $G = \mathbb{Z}_2 \otimes_{wr} S_n$. Two k sets of vectors (or two orientations of H^n) are *isometric* if there exists an element in G sending the first set (or orientation) into the second one. Sets of vectors and orientations are therefore partitioned into equivalence classes. The next proposition taken from [5] describes a simple but fundamental property of the sum of outer products of degree d under the action of G .

PROPOSITION 3.1. *For every σ in G , every vector X and any set of vectors $M^1, \dots, M^k$, we have*

$$E^d \langle M^1, \dots, M^k \rangle (X) = E^d \langle \sigma M^1, \dots, \sigma M^k \rangle (\sigma X)$$

and therefore

$$E^d \langle \sigma M^1, \dots, \sigma M^k \rangle (H^n) = \sigma (E^d \langle M^1, \dots, M^k \rangle (H^n)).$$

Proof. It is enough to show it for the generators of G . However, see [5] for more details.

If we define the *stabilizer* $S^d \langle M^1, \dots, M^k \rangle$ of $M^1, \dots, M^k$ with respect to the sum of outer products of degree d by

$$(7) \quad S^d \langle M^1, \dots, M^k \rangle = \{ \sigma \in G : E^d \langle M^1, \dots, M^k \rangle (X) = E^d \langle \sigma M^1, \dots, \sigma M^k \rangle (X) \}$$

(the equality must of course hold for every vector X), then the action of the stabilizer partitions H^n into orbits of points of identical energy value and dynamical behavior (see [5]). In particular, $E^d(M^s)$ does not depend on s . In computing

$$E^d(X) = \sum_{|I|=d} \sum_{s=1}^k (M^s)^I X^I$$

we can obviously interchange the sums, and a term of the form $\sum_{|I|=d} (M^s)^I X^I$ depends only on the Hamming distance $d_H(M^s, X)$. As a scalar, it is invariant under the action of G and therefore we can always assume, for instance, that $X = (++ \dots +)^I$ and $M^s = (++ \dots + - - \dots -)^I$. Therefore the total energy of a point X is entirely determined by the sequence of its Hamming distances to the vectors $M^1, \dots, M^k$. As a consequence, not only $E^d(M^s)$ is constant but also $E^d(M^\perp)$, where $M^\perp$ is any vector in $H^n \cap \langle M^1, \dots, M^k \rangle^\perp$, provided this intersection is not empty (i.e., the sets of vectors can be

enlarged). In a similar fashion, the radius $R^d(M^s)$ does not depend on s . To explicitly compute the values of E^d and I^d for M^s , $M^\perp$, $\dots$ we shall need several lemmas.

Let a, b, n be three integers such that $a \leq n$ and $b \leq n$. Fix an a -element subset in an n -element set and let $f^+(n, a, b)$ (respectively, $f^-(n, a, b)$) be the number of b -element subsets of the n -elements set that have an even (respectively, odd) intersection with the a -elements set. We shall use n and a to denote both the sets and their cardinals. Obviously $f^+ + f^- = \binom{n}{b}$. We are interested in $f(n, a, b) = f^+ - f^-$.

PROPOSITION 3.2.

$$f^+(n, a, b) = \frac{1}{2} \binom{n}{b} + \frac{1}{2} \sum_{t=0}^a (-1)^t \binom{a}{t} \binom{n-a}{b-t},$$

$$f^-(n, a, b) = \frac{1}{2} \binom{n}{b} + \frac{1}{2} \sum_{t=0}^a (-1)^{t+1} \binom{a}{t} \binom{n-a}{b-t}.$$

Proof. It is enough to show it for f^+ . Obviously

$$f^+(n, a, b) = \sum_{2|t} \binom{a}{t} \binom{n-a}{b-t}.$$

Therefore if we look at the generating function

$$\sum_{2|t} \binom{a}{t} \binom{n-a}{b-t} x^n y^a z^t = \sum_{2|t} \binom{n-a}{b-t} x^{n-a} y^{b-t} \binom{a}{t} x^a y^t z^t$$

simple algebraic manipulations show that $f^+(n, a, b)$ is the coefficient of $x^n y^b$ in

$$\frac{1}{2} \frac{x^a(1+y)^d + x^a(1-y)^d}{1 - x(1+y)}$$

which, in turn, yields the formula of the proposition.

As an important special case that can be evaluated precisely we have the following result.

PROPOSITION 3.3.

$$f^+(2a, a, 2c+1) = f^-(2a, a, 2c+1) = \frac{1}{2} \binom{2a}{2c+1},$$

$$f^+(2a, a, 2c) = \frac{1}{2} \binom{2a}{2c} + \frac{(-1)^c}{2} \binom{a}{c},$$

$$f^-(2a, a, 2c) = \frac{1}{2} \binom{2a}{2c} + \frac{(-1)^{c+1}}{2} \binom{a}{c}.$$

As a trivial consequence

$$(8) \quad f(2a, a, 2c+1) = 0,$$

$$(9) \quad f(2a, a, 2c) = (-1)^c \binom{a}{c}.$$

In order to compute the values of the inputs $I^d(X)$, we shall also need the numbers $f_{i \in a}^+(n, a, b)$, $f_{i \notin a}^+(n, a, b)$, and, respectively, $f_{i \in a}^-(n, a, b)$, $f_{i \notin a}^-(n, a, b)$. Here $f_{i \in a}^+(n, a, b)$ is defined as the number of b -element subsets containing i which intersect the a -element set evenly when i is in the a -elements subset. The definition of the

other numbers is done consistently in the same way. Also let $f_{i \in a} = f_{i \in a}^+ - f_{i \in a}^-$ and similarly for $f_{i \notin a}$. Obviously

$$f_{i \in a}^+ + f_{i \in a}^- = \binom{n-1}{b-1} \quad \text{and} \quad f_{i \notin a}^+ + f_{i \notin a}^- = \binom{n-1}{b-1}.$$

It is easy to use Proposition 3.2 to find an expression for these numbers by noticing that

$$f_{i \in a}^+(n, a, b) = f^-(n-1, a-1, b-1)$$

and

$$f_{i \notin a}^+(n, a, b) = f^+(n-1, a-1, b-1).$$

Again we are especially interested in particular cases when $n = 2a$.

PROPOSITION 3.4.

$$f_{i \in a}(2a, a, 2c) = -f(2a-2, a-1, 2c-2) = (-1)^c \binom{a-1}{c-1},$$

$$f_{i \notin a}(2a, a, 2c) = -f(2a-2, a-1, 2c-2) = (-1)^c \binom{a-1}{c-1},$$

$$f_{i \in a}(2a, a, 2c+1) = -f(2a-2, a-1, 2c) = (-1)^{c+1} \binom{a-1}{c},$$

$$f_{i \notin a}(2a, a, 2c+1) = f(2a-2, a-1, 2c) = (-1)^c \binom{a-1}{c}.$$

Proof. If i is in a , we can fix a point j in the complement of a (we disregard the trivial cases $a = 0, 1$). Then by dividing the sets of size b according to whether or not they contain the element j we have in general

$$f_{i \in a}^+(n, a, b) = f^-(n-2, a-1, b-1) + f^-(n-2, a-1, b-2).$$

In a similar fashion, we have

$$f_{i \in a}^-(n, a, b) = f^+(n-2, a-1, b-1) + f^+(n-2, a-1, b-2),$$

$$f_{i \notin a}^+(n, a, b) = f^+(n-2, a-1, b-1) + f^-(n-2, a-1, b-2),$$

$$f_{i \notin a}^-(n, a, b) = f^-(n-2, a-1, b-1) + f^+(n-2, a-1, b-2).$$

Therefore

$$f_{i \in a}(2a, a, b) = f_{i \in a}^+ - f_{i \in a}^- = -f(2a-2, a-1, b-1) - f(2a-2, a-1, b-2),$$

$$f_{i \notin a}(2a, a, b) = f_{i \notin a}^+ - f_{i \notin a}^- = f(2a-2, a-1, b-1) - f(2a-2, a-1, b-2).$$

The proof is completed by setting $b = 2c$ or $b = 2c + 1$ and using relations (8) and (9).

We are now in a position to find simple expressions for $E^d(X)$ and $I^d(X)$. We have

$$E^d(X) = - \sum_{s=1}^k \sum_{|I|=d} (M^s)^I X^I = - \sum_{s=1}^k f^+(n, d_H(X, M^s), d) - f^-(n, d_H(X, M^s), d)$$

and so

$$(10) \quad E^d(X) = - \sum_{s=1}^k f(n, d_H(X, M^s), d).$$

In particular, if $n = 2a$

$$(11) \quad E^d(M^s) = -\binom{n}{d} - (k-1)f(2a, a, d)$$

and

$$(12) \quad E^d(M^\perp) = -kf(2a, a, d).$$

Similarly, for the inputs

$$(13) \quad I_i^d(X) = \sum_{i \in I} \alpha_i X^{I - \{i\}} = \frac{1}{x_{i_s=1}} \sum_{s=1}^k \sum_{i \in I} (M^s)^I X^I$$

and so

$$(14) \quad I_i^d(X) = x_i \sum_{s=1}^k f_{i(M^s, X)}(n, d_H(X, M^s), d)$$

where, if we set $a = d_H(X, M^s)$, we have

$$\begin{aligned} i(M^s, X) &= "i \in a" && \text{if } X_i \neq M_i^s, \\ i(M^s, X) &= "i \notin a" && \text{if } X_i = M_i^s. \end{aligned}$$

We conclude this section with several remarks. (1) As it will appear in § 5, the previous results lead to three cases: $d \equiv 1 \pmod{2}$, $d \equiv 0 \pmod{4}$, and $d \equiv 2 \pmod{4}$. In particular, we have $E^d(-X) = (-1)^d E^d(X)$. Therefore if d is even, X and $-X$ have the same properties (stable, weakly stable, global maxima, . . .) and if d is odd, they tend to have opposite properties (a local minimum is turned into a local maximum . . .).

(2) A central problem in the theory of neural networks is the question of storage capacity, essentially: how many vectors can we store as stable states? This aspect is investigated in [1], [2], [4], [6], [14]. Here, because of our choice of orthogonal vectors, the value of k is already constrained by trivial considerations. If n is odd, then $k \leq 1$. If $n \equiv 2 \pmod{4}$, then $k \leq 2$. In the remaining cases, $k \leq n$. (For any d , the cases $k = 1, 2$, or even 3 can be studied directly and we leave them as an exercise for the reader. The interesting situation is therefore $n \equiv 0 \pmod{4}$ and $3 \leq k \leq n$.)

(3) With the exception of the case $n \equiv 2 \pmod{4}$ and $k = 2$, k orthogonal vectors partition H^n into two classes of points, according to the parity of their distances to these vectors. We leave as an exercise the proof of the corresponding proposition.

PROPOSITION 3.5. *Let $M^1, \dots, M^k$ be k orthogonal vectors of H^n . Except for the case $k = 2$ and $n \equiv 2 \pmod{4}$, for every X in H^n , the numbers $d_H(X, M^1), \dots, d_H(X, M^k)$ are either all even or all odd.*

(4) In correspondence with the complementation operation on sets, there exists a simple map between forms of degree d and forms of degree $n - d$. Given $M^1, \dots, M^k$, it is natural to ask what is the relation between E^d and E^{n-d} . In general, it is not true that $E^d(X) = E^{n-d}(X)$. However, this is for instance true for the vectors M^s in our orthogonal case. More precisely, we have the following.

PROPOSITION 3.6. *Let n be a multiple of 4. Then for any set of orthogonal vectors $M^1, \dots, M^k$, $E^d\langle M^1, \dots, M^k \rangle(X) = E^{n-d}\langle M^1, \dots, M^k \rangle(X)$ (respectively, $-E^{n-d}(X)$) if $d_H(X, M^s)$ is even (respectively, odd). In particular,*

$$E^d(M^s) = E^{n-d}(M^s)$$

for any $1 \leq s \leq k$ and $E^d(M^\perp) = E^{n-d}(M^\perp)$ when $M^\perp$ exists. In addition, we can find a set of vectors $N^1, \dots, N^k$ isometric to the set $M^1, \dots, M^k$ such that, for any

set I , the coefficient α_I in $E^d\langle N^1, \dots, N^k \rangle$ is identical to the coefficient $\alpha_{\bar{I}}$ of $\bar{I}$ in $E^{n-d}\langle N^1, \dots, N^k \rangle$.

Proof. Let N^1 be the vectors with all 1's. We can always find an isometry σ such that $\sigma(M^1) = N^1$. Let then $\sigma M^s = N^s$. Obviously, since $n \equiv 0 \pmod{4}$, each of the vectors N^s contains an even number of -1 's. Therefore, for any subset I and any s $(N^s)^I = (N^s)^{\bar{I}}$. And so the coefficients α_I and $\alpha_{\bar{I}}$ of $E^d\langle N^1, \dots, N^k \rangle$ and $E^{n-d}\langle N^1, \dots, N^k \rangle$ must coincide. Now assume X and Y are two generic points such that $d_H(X, M^s)$ is constantly even (see Proposition 3.5) and $d_H(Y, M^s)$ is constantly odd. Then σX (respectively, σY) has the same property and therefore must have an even number (respectively, odd) of -1 's. As a consequence $\sigma X^I = \sigma X^{\bar{I}}$ and $\sigma Y^I = -\sigma Y^{\bar{I}}$. Using Proposition 3.1 twice we finally have $E^d\langle M^1, \dots, M^k \rangle(X) = E^d\langle N^1, \dots, N^k \rangle(\sigma X) = E^{n-d}\langle N^1, \dots, N^k \rangle(\sigma X) = E^{n-d}\langle M^1, \dots, M^k \rangle(X)$ and similarly for Y with a change of sign, which completes the proof.

The previous result has several simple consequences. As an example, if a vector X with $d_H(X, M^s)$ is to be stable both in E^d and E^{n-d} , then it is necessary to have $E^d(X) = E^{n-d}(X) \leq 0$; equality is possible if and only if $E^d(Z) = E^{n-d}(Z) = 0$ for any Z at Hamming distance one from X . Also if X and Y are two neighbors with $E^d(X) = E^d(Y)$, then in general $E^{n-d}(X) \neq E^{n-d}(Y)$ unless all these energy values are equal to 0. This fact will be useful in the analysis of weak stability.

(5) In the next sections we shall show that the vectors M^s are, in general, stable. It should be pointed out that these are by no means the only stable states. The set of all stable states has a complicated structure that has not been completely elucidated.

4. Linear and quadratic case. In this section and the next, we shall occasionally use majority vectors. Given a set of ± 1 vectors $M^1, \dots, M^k$, a majority vector $\text{Maj}(M^1, \dots, M^k)$ or just Maj is defined by the relations $\text{Maj}_i = \text{sgn}(\sum M_i^s)$. When k is odd, this vector is uniquely defined. When k is even, $\sum M_i^s$ may be zero for some i (but not all i in the orthogonal situation, because of independence), in which case we can complete with a $+1$ or a -1 . According to this definition, all such completions result in a majority vector.

THEOREM 4.1. *Let $d = 1$. Then*

$$E^1(M^s) = -n, \quad E^1(M^\perp) = 0, \quad I_i^1(X) = \alpha_i = \sum_{s=1}^k M_i^s.$$

The stable points are the majority vectors. They form a connected subgraph of H^n and they correspond to global minima of E^1 over H^n . When there is not a unique majority, the stability is weak. The only possible critical points may occur for $\pm \text{Maj}$ and only when Maj is uniquely determined. $M^\perp$ is never stable, never critical. For $k > 2$, M^s is generally not stable. Stability may occur in special cases with k even.

Proof. The expressions for E^1 , I^1 are direct consequences of the definitions. The hypercube is a convex polytope, and here E is linear. Therefore, by elementary results on linear programming, the global minima form a connected subgraph of H^n which corresponds also to the set of majority vectors. All these points have energy $-\sum |\alpha_i|$. When there is not unicity of the majority, optimal neighbors have the same energy value and therefore are weakly stable. A critical point X satisfies $\sum_s M^s = \lambda X$ and so because of independence M^s and $M^\perp$ cannot be critical. Here $E^1(M^s) < E^1(M^\perp)$ and again because of convexity $M^\perp$ cannot be a local minimum and therefore cannot be a stable point.

THEOREM 4.2. *Let $d = 2$. Then*

$$E^2(M^s) = -\frac{1}{2}n(n-k), \quad E^2(M^\perp) = \frac{1}{2}kn,$$

$$I_i^2(X) = \sum_i \alpha_{i,j} x_j.$$

Here $\pm M^s$ are stable and critical. They correspond to global minima of E on H^n . They are weakly stable only in the case of $k = n$ for which $E \equiv 0$. Also, $M^\perp$ is unstable, critical, and corresponds to a global maximum of E . Finally, $[(n-k)/2k] \leq R^2(M^s)$.

Proof. It is easy to see that in the quadratic case $E^2(X) = -\frac{1}{2} X^t A X$ where A is the real, symmetric, 0-diagonal matrix defined by

$$(15) \quad A = \sum_{s=1}^k M^s M^{s^t} - kI.$$

Therefore A has n real eigenvalues with trace 0 and a basis of eigenvectors. Obviously $A(M^s) = (n-k)M^s$ and $A(M^\perp) = -kM^\perp$ whenever $M^\perp$ exists. The expressions for E^2 and I^2 are then immediate. Because of the properties of quadratic forms, M^s and $M^\perp$ are critical points. In fact they correspond, respectively, to global maxima and global minima of E^2 over the sphere of radius $\sqrt{n}$ and therefore also over H^n . In particular, M^s is stable. In the special case $k = n$, $A(M^s) = 0$ and so $A \equiv 0$ and then $E^2(X) = 0$ for any X . Notice now that the set of global minima of E^2 over H^n , which is also the set of points with energy $-\frac{1}{2}n(n-k)$, corresponds exactly to the intersection of the linear span of $M^1, \dots, M^k$ with H^n .

To estimate the radius of attraction $R^2(M^s)$ we can assume without any loss of generality that $M^s = (++ \dots +)^t$ and $X = (-\dots - + \dots +)^t$. If X is to converge to M^s then we must have $I_i(X) > 0$ for $1 \leq i \leq d_H(X, M^s)$ and $I_i(X) \geq 0$ for the remaining components. By noticing that $M^{s^t}X = n - 2d_H(M^s, X)$ and using (15) these inequalities can be written as

$$n - 2d_H(X, M^s) + \sum_{t \neq s}^k (n - 2d_H(X, M^t)) > -k$$

and

$$n - 2d_H(X, M^s) + \sum_{t \neq s}^k (n - 2d_H(X, M^t)) \geq k.$$

By the triangle inequality $d_H(X, M^t) \leq (n/2 - d_H(X, M^s))$. Therefore in the worst case we must have $n - 2d_H(X, M^s) + (k-1)2d_H(X, M^s) \geq k$ which yields

$$d_H(X, M^s) \leq (n-k)/2k$$

and so $R^2(M^s) \geq [(n-k)/2k]$. This bound is in general the best possible for it is easy to construct examples where it is attained.

5. The general case.

5.1. Odd degrees. First, let us treat the case of $d = 2c + 1$, $d > 1$. Let

$$\beta_i(X) = \#\{M^s: M^s \neq X \text{ and } M_i^s = X_i\},$$

$$\gamma_i(X) = \#\{M^s: M^s \neq X \text{ and } M_i^s \neq X_i\}.$$

Clearly, for any X , $k-1 \leq \beta_i(X) + \gamma_i(X) \leq k$ and $-k \leq \beta_i(X) - \gamma_i(X) \leq k$.

THEOREM 5.1. *Assume $d = 2c + 1$, $c > 0$, $n = 2a$, $k \leq n$, and $M^1, \dots, M^k$ are k orthogonal vectors of H^n . Then*

$$\begin{aligned} E^d(M^s) &= -\binom{n}{d}, & E^d(M^\perp) &= 0, \\ I_i^d(M^s) &= M_i^s \left(\binom{n-1}{d-1} + (-1)^c \binom{a-1}{c} (\beta_i(M^s) - \gamma_i(M^s)) \right), \\ I_i^d(M^\perp) &= M_i^\perp \left((-1)^c \binom{a-1}{c} (\beta_i(M^\perp) - \gamma_i(M^\perp)) \right). \end{aligned}$$

Moreover, M^s is always stable, and weakly stable only if $d = 3$ and $k = n$ or $d = n - 1$ and $k = n$. M^s is never critical and in general it is not a global minimum of E over H^n . ($-M^s$ is always unstable, never critical, and in general not a global maximum.) $M^\perp$ is never stable, never critical, and never a global maxima.

Proof. $E^d(M^s)$ and $E^d(M^\perp)$ are readily obtained by applying Proposition 3.3 to (11) and (12). For the inputs, from (14) $I_i^d(M^s) = M_i^s \sum_{t=1}^k f_{i(M^t, M^s)}(n, d_H(M^s, M^t), d) = M_i^s f_{i \notin a}(2a, 0, 2c + 1) + M_i^s \sum_{t \neq s} f_{i(M^t, M^s)}(2a, a, 2c + 1)$. Now $f_{i \notin a}(2a, 0, 2c + 1) = \binom{n-1}{d-1}$ whence using Proposition 3.4 for the second part of the sum we get the formula of the theorem. The proof is similar for $I_i^d(M^\perp)$.

Obviously $E^d(-M^s) > E^d(M^\perp)$ and so $M^\perp$ cannot be a global maximum. Moreover, M^s (respectively, $M^\perp$) cannot be a critical point for otherwise $\beta_i(M^s) - \gamma_i(M^s)$ (respectively, $\beta_i(M^\perp) - \gamma_i(M^\perp)$) would need to be constant and M^s (respectively, $M^\perp$) would be in the span of $M^1, \dots, M^{s-1}, M^{s+1}, \dots, M^k$ (respectively, $M^1, \dots, M^k$) if the constant value is nonzero. But the constant cannot be zero because orthogonal vectors are independent.

We turn now to the questions of stability. M^s is stable if and only if for any i

$$\binom{n-1}{d-1} + (-1)^c \binom{a-1}{c} (\beta_i(M^s) - \gamma_i(M^s)) \geq 0.$$

In particular, it will be sufficient to have

$$\binom{n-1}{d-1} \geq (-1)^c \binom{a-1}{c} (k-1).$$

When the inequality is strict the stability cannot be weak. Notice that since $n = 2a$ and $d = 2c + 1$ it follows that $a > c$. The following simple inequality will be useful.

LEMMA 5.2. *For any integers $1 \leq c < a$ and $1 \leq k \leq 2a$*

$$\binom{2a-1}{2c} \geq (k-1) \binom{a-1}{c}.$$

Equality occurs if and only if $k = 2a$ and $c = 1$ or $k = 2a$ and $c = a - 1$.

The proof is straightforward by studying first the special cases $c = 1$, $c = a - 1$, and then by induction on a .

The above lemma shows that M^s is always stable. Weak stability can only occur in the saturated case of $k = n$ with $c = 1$ ($d = 3$) or with $c = a - 1$ ($d = n - 1$). For $n > 3$, n must be a multiple of 4 and therefore c is odd. Thus weak stability for M^s occurs if and only if the Hadamard matrix made by the column vectors M^s contains a row i of constant sign, and this can happen in at most one row. As a result, when weak stability appears, it is only along a unique common component i for all the vectors $M^1, \dots, M^k$ and here $E^d \neq 0$.

As an example, let $n = 4$, $d = 3$ and consider the two Hadamard matrices corresponding to $M^1 = (+++-)^t$, $M^2 = (++-+)^t$, $M^3 = (+-++)^t$, $M^4 = (+---)^t$, and $N^1 = (---+)^t$, $N^2 = M^2$, $N^3 = M^3$, $N^4 = M^4$. Then in the first situation

$$E^3\langle M^1, \dots, M^4 \rangle(X) = 4x_2x_3x_4,$$

$E^3(M^s) = -4$ and $M^1, \dots, M^4$ are weakly stable along the first coordinate. In the second situation, $E^2\langle N^1, \dots, N^4 \rangle(X) = 2(x_1x_2x_3 + x_2x_3x_4 - x_1x_3x_4 - x_1x_2x_4)$ and $E^3(N^s) = -4$. However, this time the vectors N^s are strictly stable. For instance, for the four neighbors of N^1 we have $E^3(+---)^t = 8$, $E^3(-++-)^t = 0$, $E^3(--+-)^t = 4$, and $E^3(----)^t = 0$. This example also shows that the equation $E^{2c+1}(X) = 0$ may have solutions, over H^n , other than the vectors $M^\perp$. Assume now, for contradiction, that $M^\perp$ is stable. Then because of the expression for its input, $\beta_i(M^\perp) - \gamma_i(M^\perp)$ must have a constant sign and therefore $M^\perp$ or its opposite must be a majority vector. But this is impossible because of the following proposition.

PROPOSITION 5.3. *A majority of k orthogonal $M^1, \dots, M^k$ vectors can never lie in the space orthogonal to the space spanned by $M^1, \dots, M^k$.*

Proof. The proposition is trivial for $k = 1, 2$. Notice now that orthogonality and majority relations are preserved under action of the group G . Therefore, if $k > 2$ we can always assume that $M^1 = (+++-)^t$, $M^2 = (++-+)^t$, $\text{Maj} = (+++++)^t$. Assume now that k is even, $k = 2l$. Then in the matrix formed by the vectors $M^3, \dots, M^k$ it is easy to check that there must be at least ln ones because of the majority constraints, and exactly $(l-1)n$ ones because of the orthogonality constraints. This is a contradiction. A similar argument works for the case k odd (it is not even necessary to introduce M^2).

We finally give an example showing that M^s need not be a global minimum of E^d over H^n . Take $d = 3$, $n = 4$, and $k = 3$ with $M^1 = (++++)^t$, $M^2 = (++-+)^t$, and $M^3 = (+-+-)^t$. Then $E(X) = x_1x_2x_3 - x_1x_2x_4 - x_1x_3x_4 - 3x_2x_3x_4$, $E(M^s) = -4$, $E(M^\perp) = 0$, and $E(\text{Maj}) = 6$ which implies $E(-\text{Maj}) = -6$ and it is the global minimum.

5.2. Even degrees. We now treat the case of $d = 2c$, $d > 2$.

THEOREM 5.4. *Assume $d = 2c$, $c > 1$, $n = 2a$ and that $M^1, \dots, M^k$ are k orthogonal vectors of H^n . Then*

$$E^d(M^s) = -\binom{n}{d} - (k-1)(-1)^c \binom{a}{c},$$

$$E^d(M^\perp) = -k(-1)^c \binom{a}{c},$$

$$I_i^d(M^s) = M_i^s \left(\binom{n-1}{d-1} + (-1)^c (k-1) \binom{a-1}{c-1} \right),$$

$$I_i^d(M^\perp) = M_i^\perp \left((-1)^c k \binom{a-1}{c-1} \right).$$

If $c = 2e$, then M^s , $M^\perp$, and their opposites are stable (never weakly) and critical. If $c = 2e + 1$, then M^s (and $-M^s$) is always stable, critical, and weakly stable if and only if $k = n$ with $d = 2$ or dually $d = n - 2$ which forces E^d to be 0 everywhere. $M^\perp$ is never stable and always critical. In general it is not a global maximum.

Proof. The proof follows the same steps as for the case of odd degree. Using the results of § 3, the expressions for E^d and I^d are easy to derive; the subcase $c = 2e$ is immediate. For $d = 2c = 4e + 2$ we need the following lemma.

LEMMA 5.5. *For any integers $1 \leq c < a$ and $1 \leq k \leq 2a$*

$$\binom{2a-1}{2c-1} \geq (k-1) \binom{a-1}{c}$$

with equality if and only if $k = 2a$ and $c = 1$ or $k = 2a$ and $c = a - 1$.

The proof is similar to that of Lemma 5.2. Consequently, weak stability occurs if and only if $d = 2$ or dually $d = n - 2$ with an Hadamard matrix corresponding to $k = n$. In the quadratic case, we have seen that this forces E^2 to be identically 0. By the duality of Proposition 3.6 this must also be true for $d = n - 2$.

6. Conclusion and open questions. From the point of view of the theory of general neural automata the orthogonal storage case is quite a special one. However, it is of importance since many of its properties can be computed exactly. In this sense it may play a role similar to the “exactly solvable models” of statistical mechanics [7] and may, especially for quadratic energy functions, have several properties in common with the case where the vectors $M^1, \dots, M^k$ are chosen at random with n large. Moreover, simulations [9] seem to indicate that these automata are “robust” and therefore many properties should be conserved when the condition of orthogonality is slightly relaxed.

So far we have succeeded in computing several properties for these networks. Our results show in particular that there are significant differences of behavior according to whether $d \equiv 1, 3 \pmod{4}$ or $d \equiv 0 \pmod{4}$ or $d \equiv 2 \pmod{4}$. It remains to be seen if these differences can be exploited in applications. Several theoretical questions require additional work: determination of the stabilizers, structure of the critical points, structure of the global extrema, values of radius like $R^d(M^s)$, structure of the set of all stable points, and under which conditions different sets of vectors generate the same orientation for a given d . Or, what is characterized by the sequence of orientations when d varies?

Under different possible sets of assumptions the maximal number of vectors that can be stored as stable points of a neural automaton with an energy function of degree d scales as $O(n^{d-1})$ [6]. Therefore, if this approach is to be tried on families of vectors that are not orthogonal, it is important to know in advance the maximal possible size for the family in question. This paper has been concerned with a particular family with linear size (see [2] for a more general type of linear family that may be relevant in this context).

Finally, from a purely combinatorial standpoint, we have seen that

$$E^2 \langle M^1, \dots, M^n \rangle \equiv 0$$

and it is easy to check that if $n \equiv 2 \pmod{4}$ then $E^n \langle M^1, M^2 \rangle \equiv 0$. In other words in these two trivial cases of saturation, there exists a relatively simple form which suddenly vanishes everywhere when saturation is reached. One may wonder if saturation in non-trivial cases may be detectable via a similar abrupt change in the landscape. The phenomenon should not be identical for we have checked that there exists a saturated system of four vectors with $n = 12$ such that $E^d \langle M^1, \dots, M^4 \rangle \not\equiv 0$ for any d .

Acknowledgment. I would like to thank Ed Bender for a useful discussion.

REFERENCES

- [1] Y. ABU MOSTAFA AND J. ST. JAKUES, *Information capacity of the Hopfield model*, IEEE Trans. Inform. Theory, 31 (1985), pp. 461–464.
- [2] D. J. AMIT, H. GUTFREUND, AND H. SOMPOLINSKY, *Spin glass models of neural networks*, Phys. Rev. A, 32 (1985), pp. 1007–1018.

- [3] M. ASCHBACHER, P. BALDI, E. BAUM, AND R. M. WILSON, *Embeddings of ultrametric spaces in finite dimensional structures*, SIAM J. Algebraic Discrete Methods, 8 (1987), pp. 564–577.
- [4] P. BALDI, *Neural networks, orientations of the hypercube and algebraic threshold functions*, IEEE Trans. Inform. Theory (1988) (to appear).
- [5] ———, *Group actions and learning for a family of automata*, J. Comput. System Sci., 36 (1988) to appear.
- [6] P. BALDI AND S. VENKATESH, *Number of stable points for spin glasses and neural networks of higher orders*, Phys. Rev. Lett., 58 (1987), pp. 913–915.
- [7] R. BAXTER, *Exactly Solved Models in Statistical Mechanics*, Academic Press, New York, 1982.
- [8] D. O. HEBB, *The Organization of Behavior*, John Wiley, New York, 1949.
- [9] J. J. HOPFIELD, *Neural networks and physical systems with emergent collective computational abilities*, Proc. Nat. Acad. Sci. USA, 79 (1982), pp. 2554–2558.
- [10] J. J. HOPFIELD AND D. W. TANK, *Neural computation of decisions in optimization problems*, Biol. Cybernet., 20 (1985), pp. 141–152.
- [11] S. KIRKPATRICK, C. D. GELATT, AND M. P. VECCHI, *Optimization by simulated annealing*, Science, 220 (1983), pp. 671–680.
- [12] R. P. LIPPMAN, *An introduction to computing with neural nets*, IEEE ASSP Magazine, 4–22 (April 1987).
- [13] W. A. LITTLE, *The existence of persistent states in the brain*, Math. Biosci., 19 (1974), pp. 101–120.
- [14] R. J. McELIECE, E. C. POSNER, E. R. RODEMICH, AND S. S. VENKATESH, *The capacity of the Hopfield associative memory*, IEEE Trans. Inform. Theory, 33 (1987), pp. 461–482.
- [15] E. VERHEIDEN, *Integral and rational completions of combinatorial matrices*, J. Combin. Theory Ser. A, 25 (1978), pp. 267–276.

ASYMPTOTIC ANALYSIS OF SOME NETWORK RELIABILITY MODELS*

DANIEL BIENSTOCK†

Abstract. We consider the problem of designing reliable networks at low cost, and show that for several standard and nonstandard models, 0-1 effects occur.

Key words. random graphs, network reliability

AMS(MOS) subject classifications. 05, 60, 68, 90, 94

1. Introduction. In this paper we will consider the design of large-scale, low hardware cost communication networks in the presence of probabilistic failure. Under such circumstances, the designer will ordinarily try to trade off reliability versus cost. We analyze several models of failures and performance requirements, and show that, asymptotically, 0-1 effects occur in the reliability vs. cost tradeoff (this statement will be made more precise below).

We will use the words “graph,” “network” and “design” interchangeably. “Design” will also be used to imply a **design rule** or **sequence of graphs**, one for each possible number of nodes.

In order to approach the reliability vs. cost issues from a mathematical viewpoint, we must precisely indicate what we mean by a “model.” In what follows, a reliability model will be a specification of:

(i) **A performance measure**, that is, what we mean when we say that a network “works.” The traditional network reliability literature has heavily favored connectivity measures. In a practical setting, however, links may have finite capacity, and certain flow levels (possibly stochastic) may have to be maintained. In this paper we will consider both connectivity and flow requirement measures.

(ii) **A distribution of failures.** In the standard literature network elements (e.g., edges) fail independently. We consider models both with independent and dependent failures.

(iii) **A measure of cost** of a design. The simplest measure for the quantity of hardware is the number of edges in a design. When considering flow requirements, the capacities of links should also be considered.

The main result concerning the asymptotic analysis of such models can be abbreviated as follows (each model requires a specific proof):

In a given model, let $\{G(n)\}$ be a sequence of graphs, where $G(n)$ has n nodes, hardware cost $H(n)$ and reliability $R(n)$. There exists a “threshold” function $T(n)$, such that if

(a) $H(n) \leq cT(n)$ where $c < 1$, then $R(n) \rightarrow 0$, independent of design. On the other hand, we can produce designs with

(b) $H(n) = O(T(n))$ and $R(n) \rightarrow 1$.

Thus, the designs that attain (b) are essentially optimal. The proof of (a) yields as a by-product, tight upper bounds on reliability that depend only on cost, not on any specific

* Received by the editors October 21, 1985; accepted for publication January 26, 1987. This research was carried out at the Massachusetts Institute of Technology, Cambridge, Massachusetts 02139 and was partially supported by National Science Foundation grant ECS-8316224.

† Bell Communications Research, Morristown, New Jersey 07960.

design. Similarly, we can easily compute tight lower bounds on the reliability of the efficient designs that attain (b).

Kel'mans [5], [6] analyzed the simplest case of a model. In his model we want to minimize the total number of edges in a design, in order to keep all nodes connected while edges are independently erased with probability $0 < p < 1$. Kel'mans showed that in this case $T(n) = \frac{1}{2}k(p)n \log n$, where $k(p) = -1/\log p$. The efficient networks are obtained by taking, roughly, n/m copies of k_m (where $m = k(p) \log n$), and making copy i share one node with copy $i + 1$, for $1 \leq i < n/m$.

We should mention that Leighton and Leiserson [7] have analyzed a set of problems which, although they do not fit in our framework, yield similar results.

Sections 3 and 4 analyze simple models. Section 5 analyzes some more complicated models.

2. Some notation and auxiliary results. Let $\{E_n\}$ denote a sequence of events (= sets of points in a probability space). We will say that E_n holds **with high probability** if $P(E_n) \rightarrow 1$ as $n \rightarrow \infty$.

Let $G(n)$ be the n th graph in a sequence of designs. We will denote by $\mathbf{G}(n)$ the random variable obtained through the failure of elements of $G(n)$. Let $C(n)$ be any property satisfied by saying: $\mathbf{G}(n)$ satisfies $C(n)$.

Let $\{G(n)\}$ be as above. Typically the reliability of $G(n)$ will depend on a family F of parameters of the model (e.g., edge reliabilities). We will denote the reliability of $G(n)$ by $R_n(F)$; and $R(F) = \lim_{n \rightarrow \infty} R_n(F)$. We will also say that $\{G(n)\}$ is **reliable** if $R(F) = 1$.

Given two functions $f(n)$ and $g(n)$, we will say that f and g are log-equivalent if $\log g(n) \sim \log f(n)$. Unless otherwise stated, all logarithms are to the base e .

An LEF (logarithmic-exponential function) is a function f of the form $f(n) = (1 - a(n))^{b(n)}$, where $a(n) \rightarrow 0^+$ and $b(n) \rightarrow \infty$ as $n \rightarrow \infty$. It is a simple exercise to verify that

$$(1) \quad \begin{array}{ll} \text{(i) if } \log b(n) + \log a(n) \rightarrow +\infty & \text{(i) then } f(n) \rightarrow 0, \\ \text{(ii) if } \log b(n) + \log a(n) \rightarrow -\infty & \text{(ii) then } f(n) \rightarrow 1. \end{array}$$

A useful tool in the asymptotic analysis of sums of random variables are the Chernoff bounds [2], here stated in abridged form. Let S_n be the sum of n independent variables distributed as

$$X = \begin{cases} 1 & \text{w.p. } Q, \\ 0 & \text{w.p. } 1 - Q, \end{cases}$$

and suppose $a < Q$. Then

$$(2) \quad P(S_n \leq an) \leq [r(Q, a)]^n$$

where

$$(3) \quad r(Q, a) = \left[\frac{Q}{a} \right]^a \left[\frac{1-Q}{1-a} \right]^{1-a} < 1.$$

3. A connectivity model with independent arc failures. The following model is a generalization of Kel'mans'. The edges of our network fail independently with probability $0 < p < 1$. We **randomly** select a subset S of $f(n)$ nodes, where n = total number of nodes. We want to minimize the total number of edges while guaranteeing that S remains connected. The only assumption concerning $f(n)$ is that $f(n) \rightarrow \infty$ as $n \rightarrow \infty$.

Consider a sequence $\{G(n)\}$ of graphs. Let $R_n(p)$ = reliability of $G(n)$ and $R(p) = \lim_{n \rightarrow \infty} R_n(p)$, and set $a(n)$ = number of edges of $G(n)$.

THEOREM 1. (a) Suppose $a(n) \sim (k(p')/2)n \log f(n)$, where $p' < p$. Then $R(p) = 0$.

Sketch of proof. (i) Choose $x > 0$ so that $k(p') + x < k(p)$. Then $T = 1 + (k(p') + x) \log p > 0$. Set $d(n) = (k(p') + x) \log f(n)$. We can find a set X , all of whose nodes have degree at most $d(n)$, with $|X| = \Omega(n)$. Thus, we can find an independent set X' all of whose nodes have degree at most $d(n)$, with $|X'| = \Omega(n/\log f(n))$ (this argument is similar to one used by Kel'mans).

(ii) In $\mathbf{G}(n)$, at least

$$\Omega\left(\frac{n}{[f(n)]^D \log f(n)}\right)$$

nodes are isolated, where $D = 1 - T > 0$. Let I be the set of isolated nodes in $\mathbf{G}(n)$.

(iii)

$$P(I \cap S = \emptyset) \leq \frac{\binom{n-|I|}{f(n)}}{\binom{n}{f(n)}} \rightarrow 0$$

(strictly speaking, the above estimation is correct if $f(n) \leq n - |I|$. If $f(n) > n - |I|$, then certainly $I \cap S \neq \emptyset$). $\square$

Before proceeding to give a tight upper bound, we point out that the above proof will still hold when there are parallel edges.

(b) There exist designs with $a(n) = O(k(p)n \log f(n))$ that attain $R(p) = 1$.

These designs are obtained by taking a copy of K_m , where $m = \lceil 2k(p) \log f(n) \rceil$, and connecting $n-m$ additional nodes to all the nodes of K_m . We will refer to such a graph as an (n, m) -starfish. The copy of K_m will be called the hub and the set of remaining nodes, the periphery.

In order to show that these designs are reliable, we prove the following:

(i) As a subgraph of $\mathbf{G}(n)$, the hub remains connected. This is an easy consequence of the Erdős and Rényi [3] results on random graphs, since with high probability $\theta(m^2)$ edges remain operative in the hub. Also, in [5] it is proven directly that the connectivity probability of K_N , when edges are erased with probability p , is at least

$$1 - (N-1)^2 p^{N-1}$$

if $Np^{N-3} \leq 1$.

(ii) At most $f(n)$ nodes of S are chosen in the periphery. With high probability, there is at least one operative edge connecting each such node to the hub. $\square$

This concludes the proof of Theorem 1. Notice that the (n, m) -starfish graphs are obtained from $K_{n-m, m}$ by adding only $O(m^2)$ edges. In fact, with m chosen as above, one can show that $K_{n-m, m}$ is also reliable (however, the starfish graphs have some additional desirable properties described below).

We note in passing that Kel'mans' design (intended to keep all nodes connected) can only be generalized to our model when $\log f(n) = \theta(\log n)$ (otherwise the design will not be simultaneously optimal and reliable). In such cases, the starfish graphs are more desirable. For example, let us assume $f(n) = n$ (all other cases are similar). Then the starfish design has $m = 2k(p) \log n$, and

(1) While Kel'mans' design is 1-connected, we can show that with high probability, the starfish graphs remain $\Omega(\log n)$ -connected. This statement is trivial: With high probability every pair of nodes remains adjacent to $\Omega(\log n)$ common hub nodes.

(2) The Kel'mans design has $\Omega(n/\log n)$ diameter while the starfish designs, with high probability, have diameter at most six.

One disadvantage of the starfish graphs in the case $f(n) = n$ is that they require high layout area (in the VLSI sense). However, we can produce a reliable design with $O(n \log^3 n)$ area, with diameter $O(\sqrt{n/\log n})$ and node connectivity $\Omega(\log n)$ (both with high probability). One such design is obtained by taking $\sim n/m$ copies of K_m (as above) and placing them in the grid points of an n/m -node square grid. Then, the nodes in each copy of K_m are connected to all of the nodes in four neighboring copies of K_m in the grid (thus, the resulting graph has an underlying mesh structure). We will call such a graph a **supermesh**.

Comment. It is straightforward to prove that a random graph with $cn \log n$ vertices almost always provides a reliable design for c large enough, but a construction is clearly preferable.

4. A flow model with independent arc failures. In this section we will consider a model where flows have to be routed to the nodes of the graph. More specifically, there is a special node (called the source node) from which we have to route one unit of flow to every other node. Each edge has a certain capacity parameter, which is an upper bound to the flow on that edge. Thus, in a feasible flow assignment, the difference between the flows leaving and arriving at a given node (different from the source) is exactly one.

Such a model arises in the analysis of power networks. Typically, the sum of all capacities will be a good measure of the cost of the network.

We will assume that all capacities are integer, and that edges fail independently with probability $0 < p < 1$. Now let $\{G(n)\}$ be a sequence of graphs, and $c(n)$ equal the sum of capacities in $G(n)$. Since all capacities are integer, it is clear that we must have

$$c(n) = \Omega(k(p)n \log n)$$

if $G(n)$ is not to fail. Surprisingly, this bound is tight. Before proving this fact, we state the following (simple) result.

Let $G = (U, V, E)$ be a bipartite graph, with $|U| = M$. Suppose $\deg u = d$ for all $u \in U$. Then to each $u \in U$ we can assign a node $v \in V$ such that $(u, v) \in E$ and no $v \in V$ is assigned more than $\lceil M/d \rceil$ times. (The proof of this fact is trivial. We can construct the desired assignment by first partitioning U into $\lceil M/d \rceil$ subsets of at most d nodes each, and assigning distinct nodes to the nodes in each subset.)

Now we will produce some provably good designs. These graphs are (n, m) -starfish graphs, with $m = 2k(p) \log n$, and the source located in the periphery. The capacities of the edges are as follows: All hub edges have capacity $\sim 2n^\alpha$, where $0 \leq \alpha < 1$ will be specified below. The source to hub edges have capacity $\sim (2/q^2)(n/m)$ where $q = 1 - p$. Finally, the remaining edges joining the periphery and the hub have capacity 1. We will refer to such a design as a capacitated (n, m) -starfish, or capacitated starfish for short. The total capacity is

$$O\left(\frac{n}{m} m\right) + O(n^\alpha m^2) + O(nm) = O(nm),$$

which is optimal provided we show that the designs are reliable. Let $0 < x < 1 - 1/\sqrt{2}$, and $r = r(q, (1 - x)q)$ be the Chernoff bound constant. Set

$$\alpha = \max \{0, 1 + 2(1 - x)qk(p) \log r\} < 1$$

since $\log r < 0$.

THEOREM 2. *With α defined as above, the capacitated starfish designs are reliable.*

Proof. (a) With high probability the source node remains adjacent to a set L of hub nodes, with $|L| \geq q(1 - x)m$.

(b) With high probability, every hub node not in L remains adjacent to $q^2(1-x)^2m$ nodes of L .

(c) The probability that a periphery node remains adjacent to at least $q^2(1-x)^2m$ nodes of L is at least

$$1 - r^{q(1-x)m}.$$

Let us denote by S the set of periphery nodes satisfying this condition. Clearly, in $(G)n$ there are at most $2nr^{q(1-x)m} \leq 2n^\alpha$ periphery nodes not in S .

(d) With high probability, every periphery node remains adjacent to at least one hub node.

Thus, we can route flow as follows: (1) the flow to nodes in L is sent directly. (2) In order to send flow to the remaining nodes in the hub, and the nodes in S , we send the flow from the source to L , and then directly to the respective node. Clearly this can be performed so that no more than

$$\left\lceil \frac{n}{q^2(1-x)^2m} \right\rceil$$

units of flow are switched by any node of L (and thus the source to hub capacities we assigned are sufficient, since $1/(1-x)^2 < 2$). (3) Finally, the flow to any remaining periphery node v is sent to the hub, then arbitrarily routed within the hub to any node that remains adjacent to v , and finally to v . This concludes proof of Theorem 2. $\square$

There are other designs that are nearly optimal but superior to the starfish graphs. For example, the supermesh graphs described in the previous section can be adapted to the flow model, with edges of bounded capacity and total capacity

$$O\left(\frac{k(p)}{q} n \log n\right),$$

i.e., off by a factor of $1/q$ from the optimum.

5. A model with dependent node failures. In the previous two models failures occurred independently. In this section we will consider a case of dependencies in node failures. This is intended to model (for instance) the spreading of damage in a communication network.

The failures will occur in two rounds. In the first round, nodes fail independently, with probability $0 < p < 1$. In the second round, for every node v that failed in the first round, a random neighbor w of v also fails (unless it failed already in the first round). We will then say: v kills w . All the kills are carried out independently (and as a result, damage overlap is allowed).

We want to keep all operative nodes connected. At the same time, we want to guarantee that there are “many” operative nodes (for instance, $\theta(n)$ operative nodes, where n = total number of nodes).

It turns out that the minimum number of edges that are then required is $\theta(k(p)n \log n)$. Before proving the lower bound, we will sketch the proof of a simple graph-theoretic fact.

LEMMA 1. *Let G be a connected graph with $n \gg 1$ nodes and at most $cn \log n$ edges, where c is a constant. Let $d > 2c$ be another constant. Then we find an independent set I in G with the following properties:*

- (i) $\log |I| \sim \log n$, and every node of I has degree at most $d \log n$.
- (ii) The neighbors of every node of I have degree at least two.
- (iii) If a node of I has degree larger than $\log \log n$, then the neighbors of that node also have degree larger than $\log \log n$.

(iv) For every node $z \in G$ that is adjacent to more than one node of I , there is a set $A(z)$ of nodes adjacent to z , such that:

- (1) $I \cap A(z) = \emptyset$, and no node of $A(z)$ is adjacent to a node of I .
- (2) $|A(z)| = \Omega(\log^3 n)$.
- (3) Every node of $A(z)$ has degree at most $d \log n$.
- (4) $A(z) \cap A(z') = \emptyset$ for any $z' \neq z$.

Proof. In § 3 we essentially proved that (i) can be attained. Let I be the obtained set, and replace every node of I not satisfying (ii) by a neighbor of degree 1. We will then have a new independent set of the same cardinality as previously which satisfies (ii). Without loss of generality, we will also call the new set I . Next, we show how to modify I in order to achieve (iii), while still satisfying (i) and (ii). The proof of (iv) is similar in spirit to that of (iii).

So let us assume that I satisfies (i) and (ii). Algorithm A given below will add nodes and drop nodes from I until we obtain a new set I all of whose nodes satisfy (iii).

ALGORITHM A

- (1) If every node of I satisfies (iii) STOP.
- (2) Otherwise, let v with $\deg v \leq \log \log n$ be adjacent to a node of I with degree larger than $\log \log n$. Then we
 - (2a) **Drop** from I all neighbors of v .
 - (2b) **Drop** from I every node w with $\deg w > \log \log n$, and such that w and v have a common neighbor v' with $\deg v' \leq \log \log n$.
 - (2c) **Add** v to I .
- (3) Go to (1).

Notice that we only add to I nodes that satisfy (iii) and that we add one such node at every iteration. Step (2b) guarantees that once we add a node, that node will not be dropped in a later iteration. Thus Algorithm A terminates after a finite number T of iterations. Step (2a) guarantees that the new set I is independent, and clearly it still satisfies (ii). Now let N denote the cardinality of the original set I (i.e., before running A), and M the cardinality of the obtained set I .

At every iteration of A we drop at most $(\log \log n)^2$ nodes. Thus, if

$$T \geq n/(2 \log^2 \log n)$$

since we added T nodes, we will have

$$M \geq T \quad \text{and thus} \quad \log M \sim \log N \sim \log n.$$

On the other hand, if $T < n/(2 \log^2 \log n)$, then

$$M \geq N - T(\log \log n)^2 > \frac{N}{2} \quad \text{and again}$$

$$\log M \sim \log n.$$

Hence the new set I still satisfies (i). $\square$

Now let $\{G(n)\}$ be a sequence of graphs, and suppose $G(n)$ has at most $\sim \frac{1}{2}k(p')n \log n$ edges. Let $R_2(\bullet)$ and $R(\bullet)$ have the usual meanings.

THEOREM 3. $R(p) = 0$ for $p > p'$.

Proof. Apply Lemma 1 to $G(n)$, with $c = \frac{1}{2}k(p')$ and $k(p') < d < k(p)$. Let C be the set of common neighbors of nodes of I . If $C \neq \emptyset$ then with high probability every node $z \in C$ will be killed by a node in $A(z)$. The probability of this event is at least

$$\left(1 - \left(1 - \frac{1}{d \log n}\right)^{|A(z)|}\right) \geq \left(1 - \left(1 - \frac{1}{d \log n}\right)^{\Omega(\log^3 n)}\right)^n \rightarrow 1.$$

We will next show that in $\mathbf{G}(n)$ at least one node v of I remains operative but all its neighbors are failed, after the two rounds. Given $v \in I$, let $Y(v)$ denote this event.

In order for $Y(v)$ to occur, v must survive the first round, which occurs with probability $q = 1 - p$. If $w \notin C$ is adjacent to v the probability that w fails in the first round but does not kill v is $p(1 - 1/\deg w)$. Finally, if $z \in C$ is adjacent to v , the probability that z does not kill v is at least

$$q + p \left(1 - \frac{1}{1 + |A(z)|} \right) \geq 1 - O \left(\frac{1}{\log^3 n} \right).$$

Consequently, if $\deg v \leq \log \log n$

$$P(Y(v)) \geq q \left[\left(1 - O \left(\frac{1}{\log^3 n} \right) \right) \left(\frac{p}{2} \right) \right]^{\log \log n} = \alpha(n).$$

If $\deg v > \log \log n$ then

$$P(Y(v)) \geq q \left[\left(1 - O \left(\frac{1}{\log^3 n} \right) \right) p \left(1 - \frac{1}{\log \log n} \right) \right]^{d \log n} = \beta(n)$$

since in that case every neighbor of v has degree at least $\log \log n$. Clearly $\beta(n) = o(\alpha(n))$, and thus for any $v \in I$, $P(Y(v)) \geq \beta(n)$.

Notice that if $Y(v)$ does not occur, for a certain v , this will imply that at most $d \log n$ nodes of I are killed by neighbors of v . Consequently,

$$P(\text{for at least one } v, Y(v)) \geq 1 - (1 - \beta(n))^{\Omega(|I|/\log n)} \rightarrow 1,$$

since $d < k(p)$, and $-\log \beta(n) \sim -d \log p \log n$. $\square$

Notice that Theorem 3 shows, in fact, that $\Omega(n^x)$ nodes remain operative but isolated in $\mathbf{G}(n)$, where $x > 0$.

We will next produce optimal designs. We partition the n nodes into $\sim n/m$ layers of (roughly) m nodes each, where $m = O(k(p) \log n)$, with the precise constant given below. Number the layers $1, 2, \dots, n/m$. Then, for $1 \leq i < n/m$ we connect all the nodes of layer i to all the nodes of layer $(i + 1)$ (i.e., we form a chain of complete bipartite graph). The total number of edges is $\sim (n/m)m^2 = O(k(p)n \log n)$.

Consider first the case $p \geq \frac{3}{4}$, i.e., $q \leq \frac{1}{4}$. We then choose

$$m = b / -\log r(q/2, q/2 + q/4) \log n,$$

where r is the Chernoff bound constant and b is a large enough constant independent of p . Without loss of generality, all logarithms are to the base e . One can show that

$$-\log r(Q, Q + Q/2) = \theta(-\log(1 - Q))$$

whenever Q is bounded away from 1. Also, notice that in this case $-\log(1 - Q) = \theta(Q)$. Consequently, we will have that

$$m = \theta \left(\frac{1}{q/2} \log n \right) = \theta(k(p) \log n).$$

Now we will sketch the proof of reliability of the layered designs. A complete proof is contained in [1]. First, with high probability at least $(q/2)m$ nodes survive the first round, in each layer. Consider layer i . Then, with high probability, the second round effects on layer i from layers $(i - 1)$ and $(i + 1)$ will overlap with each other (and with the first round failures in layer i) so that we can still count $\sim [(q/32) - \theta(q^2)]m$ in layer i that survive both rounds. Thus, with high probability, $\Omega(qn)$ nodes survive both rounds and remain connected.

The case $p < \frac{1}{4}$ is similar. $\square$

There is a variation of the model described in this section which is also interesting. Suppose in the second round, **all** neighbors of every node that failed in the first round are killed. Not surprisingly, the main result concerning this model is that there are **no** survivable designs, meaning:

With high probability, either

- (i) Most nodes ($\sim n$) do not survive both rounds, or
- (ii) Many nodes remain operative but isolated, or
- (iii) Both (i) and (ii) occur.

The proof of this fact (contained in [1]) is somewhat lengthy, but similar in spirit to the lower bound proof in this section.

6. Some extensions. The problems considered in the previous sections can be extended to spatial settings. These extensions are interesting because they allow geometry to play a role in costs and failure dependencies, and they are meaningful because such problems do arise in practice.

In [1] we have considered two closely related spatial settings: in one setting the nodes occur randomly within a square, and in the other the nodes occupy grid points within a square (this particular setting is similar to that used in [7]). There are two possible ways of approaching the hardware vs. reliability tradeoffs in such spatial settings:

- (1) The edges have negligible width. In this case, appropriate cost measures include total edge lengths, length of longest edge, and product of edge capacity times edge length.
- (2) The edges have a nonzero (say, constant) width. This assumption gives rise to area/reliability trade-off problems.

REFERENCES

- [1] D. BIENSTOCK, *Large-scale network reliability*, Ph.D. thesis, MIT Operations Research Center, Cambridge, MA, 1985.
- [2] H. CHERNOFF, *A measure of asymptotic efficiency for tests of a hypothesis based on a sum of observations*, Ann. Math. Stat., 23 (1952), pp. 311–327.
- [3] P. ERDŐS AND A. RÉNYI, *On random graphs I*, Publ. Math. Debrecen, 6 (1959), pp. 290–297.
- [4] P. ERDŐS AND J. SPENCER, *Probabilistic Methods in Combinatorics*, Academic Press, New York, 1960.
- [5] A. K. KEL'MANS, *Some problems of network reliability analysis*, Automat. Remote Control, 3 (1965), pp. 564–573.
- [6] ———, *Connectivity of a probabilistic network*, Automat. Remote Control, 5 (1967), pp. 444–460.
- [7] F. T. LEIGHTON AND C. E. LEISERSON, *Wafer-scale integration of systolic arrays*, Report MIT/LCS/TM-236, MIT Laboratory for Computer Science, Cambridge, MA, 1983.

UPPER AND LOWER BOUNDS ON TIME-SPACE TRADEOFFS FOR COMPUTATIONS WITH EMBEDDED FAST FOURIER TRANSFORMS*

DAVID A. CARLSON†

Abstract. Computations with embedded Fast Fourier Transforms (FFT) have many important applications which include high-speed polynomial multiplication. In this paper, we discuss the time-space tradeoff behavior of such computations. In particular, we examine the way compositions of FFT graphs with one another and with other graphs affect this behavior. We model a computation by the well-known pebble game on directed acyclic graphs. We are able to show that the way in which connections are made between subgraphs of a graph exerts a nontrivial effect on the time-space tradeoff behavior. The graphs studied correspond to the problem of finding the maximum of a vector produced by the FFT algorithm, and to problems containing back-to-back FFTs such as polynomial multiplication. For the first problem, we show that different ways of expressing the FFT algorithm lead to very different time-space tradeoff behaviors. Our results for a polynomial multiplication algorithm based on back-to-back FFTs allow us to conclude that it has the connection pattern with the best possible time-space tradeoff behavior.

Key words. Fast Fourier Transform, time-space tradeoff, pebble game, straight-line algorithm, computational complexity, polynomial multiplication

AMS(MOS) subject classification. 68Q25

1. Introduction. The development of lower bounds on the simultaneous resource requirements of algorithmic solutions to a computational problem has received a great deal of attention recently. This work is motivated by a desire to obtain a more complete understanding of a problem's computational complexity. Typical resources that affect the cost of a computation are time, measured by the number of machine instructions executed, and space, measured by the number of memory locations referenced. Bounds on the simultaneous time and space requirements of a problem yield valuable information concerning relationships between the resources. They supplement our knowledge of lower bounds on the individual resources of time and space. Usually, there is a tradeoff between time and space; a reduction in computation time requires an increase in space, and vice versa. An understanding of the time-space tradeoff characteristics of a given computational problem is important because it indicates the values of these quantities that can or cannot be achieved simultaneously by an algorithmic solution to the problem.

The simultaneous time and space requirements of a straight-line algorithm are analyzed by playing the well-known pebble game on a directed acyclic graph whose vertices represent operations performed in the algorithm and whose edges denote dependencies of one operation on another. The pebble game models the allocation of temporary storage locations to hold intermediate results during execution of the algorithm. In the game, pebbles are moved from inputs to outputs with the restriction that a pebble can be placed on a noninput vertex only if all its predecessors have pebbles on them. Time T is measured by the total number of moves required to pebble all output vertices of the graph, and space S corresponds to the maximum number of pebbles active on the graph at any one time.

When the space available to pebble a graph (perform a computation) is limited, it may be necessary to repeat the pebbling of many nonoutput vertices, from which a tradeoff between time and space can be observed. A large amount of research concerning

* Received by the editors January 27, 1986; accepted for publication (in revised form) September 1, 1987. This work was supported in part by National Science Foundation grant ECS 82-04894.

† Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, Massachusetts 01003. Present address, Supercomputing Research Center, 4380 Forbes Boulevard, Lanham, Maryland 20706.

the pebble game has dealt with the possible exchange between time and space. Paterson and Hewitt [9] introduced the pebble game and used it as a technique to study the effects of restricted space on the computation of a linear recursive program. A number of authors [6], [10], [14] have analyzed graph families corresponding to straight-line algorithms for which a decrease in space below a certain known value causes the time required to pebble a member of the family to grow faster than any polynomial in the size of the graph.

Savage and Swamy [11] used the pebble game to analyze the simultaneous time and space requirements of the Fast Fourier Transform (FFT) algorithm. For an n -input FFT, they showed that $TS = \Theta(n^2)$ for S in the range $\Theta(\log n)$ to $\Theta(n/\log n)$ (in this paper we denote the logarithm base two of n as $\log n$ and the natural logarithm of n as $\ln n$). The lower bound is a consequence of the fact that pebbling a set of $\Omega(S)$ outputs of the FFT graph requires the pebbling of $\Theta(n)$ inputs. Since there are $\Theta(n/S)$ such sets of outputs, it follows that $T = \Omega(n^2/S)$. Tompa [12] extended this work to obtain similar lower bounds on the time-space product for natural problems such as the computation of a Discrete Fourier Transform (DFT). His results rely on the fact that for any straight-line solution to a problem such as the DFT, the corresponding graph has certain connectivity properties between inputs and outputs. By an argument similar to that mentioned above, these connectivity properties imply that a large number of inputs must be repebbled in order to pebble each output set of size proportional to the available space.

The proof method outlined above can also be used to analyze the time-space tradeoff behavior of a graph composed of a sequence of subgraphs (outputs of one subgraph in the sequence are inputs to the next subgraph) as long as each subgraph possesses the proper connectivity property. Paul and Tarjan [10], Tompa [14], and Lengauer and Tarjan [8] used this method to construct graph families for which a decrease in space causes time to increase from polynomial to superpolynomial.

In this paper, we are concerned with graphs containing one or more FFT subgraphs. We show that the way FFT subgraphs are connected together and to other graphs can exert a nontrivial influence on the graph's time-space tradeoff behavior. One set of graphs we consider is an FFT subgraph whose outputs are connected (in different ways) to the inputs of a tree. This corresponds to the natural problem of computing the maximum or minimum component of the result of an FFT computation. Another set of graphs we analyze are pairs of FFT subgraphs connected "back-to-back." The computation of "back-to-back" FFTs has applications as a fast algorithm for all convolutional operations performed with FFTs, including polynomial multiplication. All of these computations arise frequently in digital signal processing [2].

Previous research on back-to-back FFT graphs has yielded lower bounds of the form $T = \Omega(n^3/S^2)$ [13] and $T = \Omega(n^2 \log n/S)$ [3], [4], both of which are independent of the pattern by which outputs of one FFT graph are connected to inputs of the other. For the FFT-tree graphs, the lower bound $T = \Omega(n^2/S + n \log n)$ follows from the work of Savage and Swamy [11] and also is independent of the connection pattern.

The results we obtain for FFT-tree and back-to-back FFT graphs can be summarized as follows. First, we exhibit a connection pattern for which the above lower bounds can be achieved over a large range of space and derive the corresponding upper bounds. We then consider a different connection pattern between subgraphs and derive lower bounds on pebbling time that are asymptotically stronger by a factor of $\log \log (n/S)$.

The results outlined above indicate that for a computation composed of two or more subcomputations, it may not be enough to know the time-space tradeoff characteristics of the subcomputations in order to determine the time and space requirements of the entire computation. The way in which inputs of one subcomputation depend on outputs of another may exert a nonnegligible influence. The results obtained also have

practical applications, e.g., to polynomial multiplication. The upper bounds presented here address this issue by indicating in the form of pebbling strategies a spectrum of multiresource efficient algorithms. Among other things, the lower bounds serve to indicate that polynomial multiplication performed using back-to-back FFTs has the connection pattern with the best possible time-space tradeoff behavior.

2. Preliminaries. In this section, we give definitions of the FFT, FFT-tree, and back-to-back FFT graphs. We also state basic properties of these graphs that will be useful in later analyses of their time-space tradeoff characteristics. The material presented here for the FFT graph is similar to that contained in Savage and Swamy [11].

The FFT algorithm can be defined either recursively or as an iterative, straight-line algorithm. Details of this definition, along with a definition of the Fourier transform of a vector, can be found in [1, Chap. 7]. The algorithm can be represented with a graph, which we define here as follows.

DEFINITION. Let $n = 2^d$, $d \geq 0$. The directed acyclic graph

$$\text{FFT}[d] = (V[d], E[d], \text{IN}[d], \text{OUT}[d])$$

on n inputs has vertex set $V[d]$, edge set $E[d]$, input vertex set $\text{IN}[d]$, and output vertex set $\text{OUT}[d]$. It is defined recursively in terms of two disjoint graphs

$$\text{FFT}_0[d-1] = (V_0[d-1], E_0[d-1], \text{IN}_0[d-1], \text{OUT}_0[d-1])$$

and

$$\text{FFT}_1[d-1] = (V_1[d-1], E_1[d-1], \text{IN}_1[d-1], \text{OUT}_1[d-1])$$

on $n/2$ inputs each as follows. Let $\text{IN}[d] = \{x_0^d, \dots, x_{2^d-1}^d\}$ be a set of vertices disjoint from $\text{FFT}_0[d-1]$ and $\text{FFT}_1[d-1]$. Then, $\text{OUT}[d] = \text{OUT}_0[d-1] \cup \text{OUT}_1[d-1]$, $V[d] = V_0[d-1] \cup V_1[d-1] \cup \text{IN}[d]$, and $E[d] = E_0[d-1] \cup E_1[d-1] \cup \Delta[d]$, where $\Delta[d] = \bigcup_{i=0}^{2^{d-1}-1} \delta_i[d]$ and $\delta_i[d] = ((x_i^d, x_{i,0}^{d-1}), (x_{i+2^{d-1}}^d, x_{i,0}^{d-1}), (x_i^d, x_{i,1}^{d-1}), (x_{i+2^{d-1}}^d, x_{i,1}^{d-1}))$. Here, (v, w) denotes an edge directed from v to w , and $\text{FFT}[0] = (V[0], \emptyset, V[0], V[0])$ where $V[0] = \{x_0^0\}$ and $\emptyset$ is the empty set.

Figure 1 illustrates an FFT graph on 16 inputs/outputs (inputs are at the top and outputs are at the bottom in all figures). Note that this graph is essentially equivalent to the definition in Savage and Swamy [11], with input vertices no longer being “scrambled” by the bit reversal permutation (the bit reversal of the d bit index of each input).

The following two lemmas illustrate important properties regarding the structure of an FFT graph.

LEMMA 1. *For $d \geq 2$ and each $1 \leq j \leq d-1$, the FFT graph $\text{FFT}[d]$ can be represented as the composition of 2^{d-j} disjoint FFT graphs on 2^j inputs*

$$\{A_l[j] \mid 0 \leq l \leq 2^{d-j} - 1\}$$

with 2^j disjoint FFT graphs on 2^{d-j} inputs $\{B_m[d-j] \mid 0 \leq m \leq 2^j - 1\}$ where the l th output of $B_m[d-j]$ is the m th input to $A_l[j]$ for $0 \leq l \leq 2^{d-j} - 1$ and $0 \leq m \leq 2^j - 1$. The i th input to $\text{FFT}[d]$ is the q th input to $B_r[d-j]$ for the unique integers $q, r \geq 0$ such that $i = q2^j + r$, $r < 2^j$; and the k th output of $\text{FFT}[d]$ is the s th output of $A_t[j]$ for the unique integers $s, t \geq 0$ such that $k = t2^j + s$, $t < 2^{d-j}$.

Figure 1 also shows the decomposition of an FFT graph implied by Lemma 1.

We note at this point that the subgraphs $A_l[j]$ and $B_m[d-j]$ extracted from $\text{FFT}[d]$ in Lemma 1 are themselves FFT graphs. Throughout the rest of this paper, we will refer to such subgraphs that can be isolated in an FFT graph as FFT subgraphs.

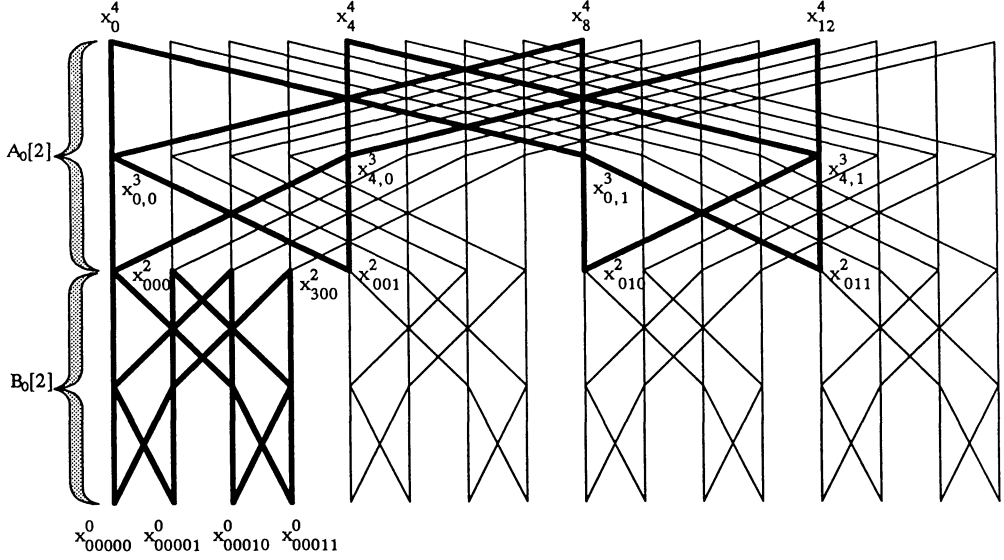


FIG. 1. FFT [4] decomposed according to Lemma 1.

LEMMA 2. In FFT $[d]$, the paths from any given input to outputs form a complete binary tree with 2^d leaves (outputs) as do the paths from all inputs to any given output.

At this point, we introduce a recursive definition of a complete binary tree that is motivated by the tree rooted at the leftmost output of FFT $[d]$. It is very similar in spirit to our previous definition of an FFT graph.

DEFINITION. Let $n = 2^d$, $d \geq 0$. The directed acyclic graph $\text{TREE}[d] = (V[d], E[d], \text{IN}[d], \text{OUT}[d])$ on n inputs has vertex set $V[d]$, edge set $E[d]$, input vertex set $\text{IN}[d]$, and output vertex set $\text{OUT}[d]$. It is defined recursively in terms of the disjoint graph $\text{TREE}[d-1] = (V[d-1], E[d-1], \text{IN}[d-1], \text{OUT}[d-1])$ on $n/2$ inputs as follows. Let $\text{IN}[d] = \{x_0^d, \dots, x_{2^{d-1}}^d\}$ be a set of vertices disjoint from $\text{TREE}[d-1]$. Then,

$$\text{OUT}[d] = \text{OUT}[d-1], V[d] = V[d-1] \cup \text{IN}[d],$$

and $E[d] = E[d-1] \cup \Delta[d]$, where

$$\Delta[d] = \bigcup_{i=0}^{2^{d-1}-1} \delta_i[d], \delta_i[d] = ((x_i^d, x_i^{d-1}), (x_{i+2^{d-1}}^d, x_i^{d-1})).$$

As a base case, $\text{TREE}[0] = (V[0], \emptyset, V[0], V[0])$ where $V[0] = \{x_0^0\}$ and $\emptyset$ is the empty set.

We are now in a position to define formally both an FFT-tree and a back-to-back FFT graph. These are the graphs that will be analyzed later in this paper.

DEFINITION. Let Π be a permutation on $1, \dots, n$. An FFT-tree graph on $n = 2^d$ inputs, $\text{FFT TREE}[d, \Pi]$ is composed of two disjoint graphs $\text{FFT}[d]$ and $\text{TREE}[d]$, each on $n = 2^d$ inputs where output i ($0 \leq i \leq 2^d - 1$) of $\text{FFT}[d]$ is connected to input $\Pi(i)$ of $\text{TREE}[d]$ for the permutation Π . A back-to-back FFT graph on $n = 2^d$ inputs, $\text{BBFFT}[d, \Pi]$ is composed of two disjoint FFT graphs $\text{FFT}_0[d]$ and $\text{FFT}_1[d]$, each on $n = 2^d$ inputs where output i ($0 \leq i \leq 2^d - 1$) of $\text{FFT}_0[d]$ is connected to input $\Pi(i)$ of $\text{FFT}_1[d]$ for the permutation Π .

To show that the connection pattern between subgraphs can have a nontrivial effect on a graph's time-space tradeoff behavior, we need to define two different connection pattern permutations and then explore the time-space tradeoff characteristics of

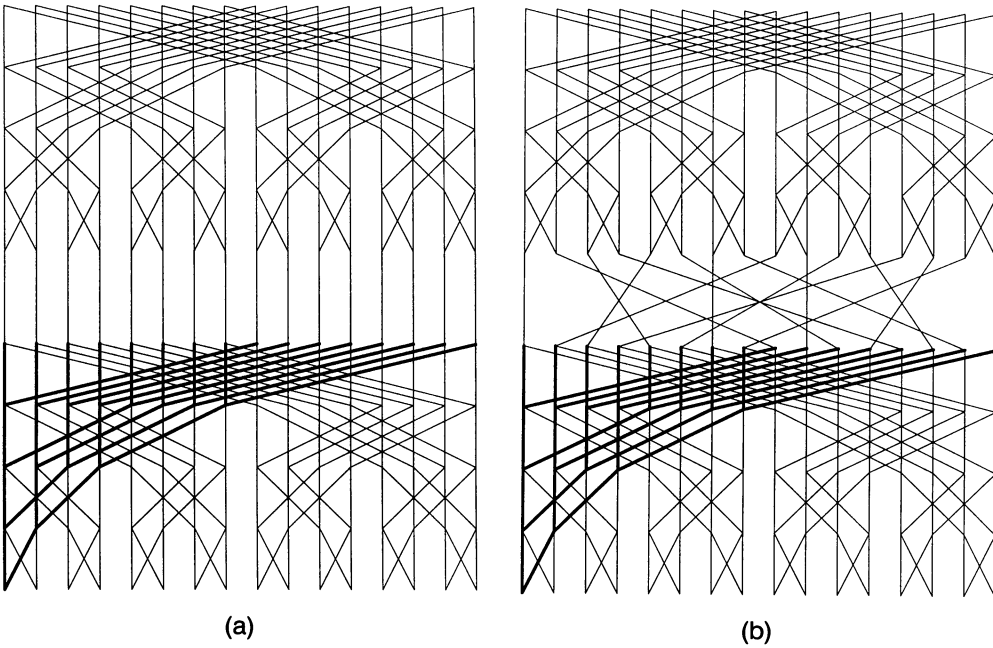


FIG. 2. (a) BBFFT [4, ID] (FFTTREE [4, ID] *highlighted*). (b) BBFFT [4, BR] (FFTTREE [4, BR] *highlighted*).

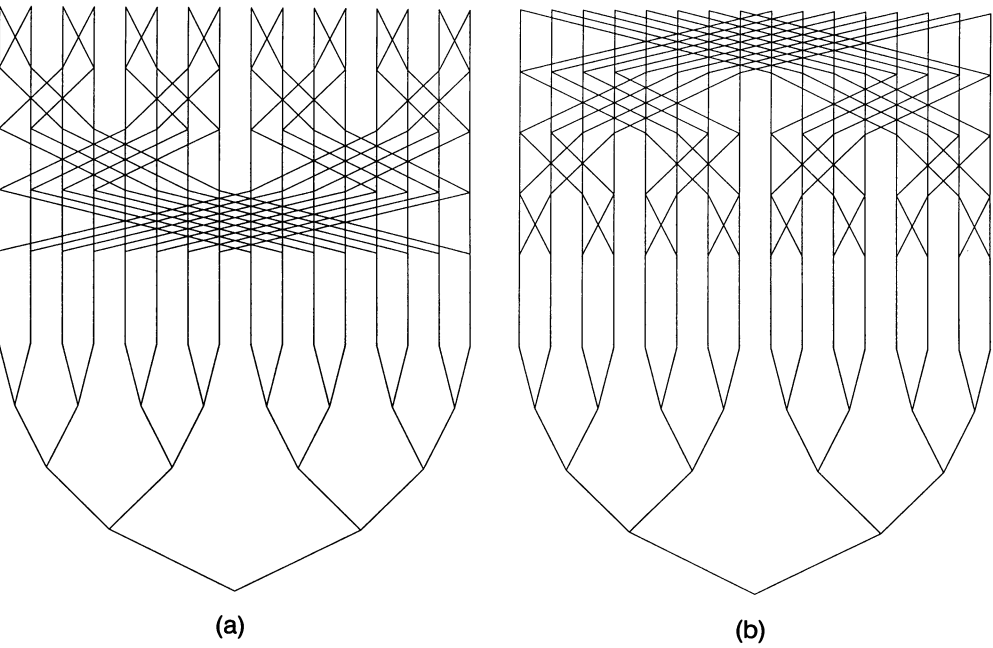


FIG. 3. *Alternative representation of* (a) FFTTREE [4, ID], (b) FFTTREE [4, BR].

the corresponding graphs. The permutations we will use here are the identity and the bit reversal permutations. Formally, when $n = 2^d$ outputs are being connected to the same number of inputs, the identity permutation ID maps output i to input i , and the bit reversal permutation BR maps output i to input $\text{BR}(i)$, where $\text{BR}(i)$ is the d -digit bit reversal of i , $0 \leq i \leq 2^d - 1$. Thus, we obtain the complementary graphs

$$\text{FFTTREE}[d, \text{ID}], \text{FFTTREE}[d, \text{BR}], \text{ and } \text{BBFFT}[d, \text{ID}], \text{BBFFT}[d, \text{BR}].$$

Figure 2 illustrates $\text{BBFFT}[4, \text{ID}]$ and $\text{BBFFT}[4, \text{BR}]$. The graphs $\text{FFTTREE}[4, \text{ID}]$ and $\text{FFTTREE}[4, \text{BR}]$ that they contain by definition are highlighted. For the FFT-tree graphs, it is interesting to note equivalent definitions with more naturally formed tree subgraphs. We omit formal definitions, but illustrate them in Fig. 3. $\text{FFTTREE}[d, \text{ID}]$ is equivalent to connecting the outputs of an FFT graph as defined by Savage and Swamy [11] to the inputs of a complete tree, while $\text{FFTTREE}[d, \text{BR}]$ is equivalent to connecting the outputs of an FFT graph as defined earlier in this paper to the inputs of a complete tree.

3. The effect of the connection pattern. In this section, we analyze the time-space tradeoff behavior of both the FFT-tree and the back-to-back FFT graphs for different connection patterns defined by the identity and bit reversal permutations. For the bit reversal permutation, we are able to derive upper bounds that match to within a constant factor lower bounds that are independent of the connection pattern. For the identity permutation, we are able to derive a lower bound that is asymptotically stronger than the upper bound for the bit reversal permutation. This allows us to conclude that the connection pattern does have an influence on the time-space tradeoff behavior of these graphs.

We start here by mentioning lower bounds that are independent of the connection pattern for later reference. Savage and Swamy's results [11] can be applied directly to the FFT-tree graph, which gives us the following proposition.

PROPOSITION 1. *Let $n = 2^d$ and Π be any permutation of $\{0, 1, \dots, 2^d - 1\}$. The number of moves required to pebble the FFT-tree graph $\text{FFTTREE}[d, \Pi]$ when S pebbles are used satisfies $T = \Omega(n^2/S + n \log n)$.*

Both Tompa [12], [13] and Carlson [3], [4] have given lower bounds on the number of moves required to pebble the outputs of a back-to-back FFT graph using different values of space, independent of the connection pattern. These results can be summarized as follows.

PROPOSITION 2. *Let $n = 2^d$ and Π be any permutation of $\{0, 1, \dots, 2^d - 1\}$. The number of moves required to pebble the back-to-back FFT graph $\text{BBFFT}[d, \Pi]$ when S pebbles are used satisfies $T = \Omega(n^3/S^2 + n^2 \log n/S)$.*

At this point, we briefly describe the basic argument that will be employed in analyzing the time-space tradeoff behavior of both the FFT-tree and back-to-back FFT graphs. Since a back-to-back FFT graph has embedded in it an FFT-tree graph, we concentrate for now on an FFT-tree graph. The tree portion of the graph can be decomposed into distinct subtrees whose inputs are inputs to the tree. All of these subtrees must be pebbled in order to reach the root of the tree. We will show that while groups of these subtrees can be pebbled in a unified fashion, there is not enough space available to pebble them all in this manner. From this it will follow that an ordering on time when groups of inputs to the tree must be pebbled is defined. Dependent on the connection pattern, this ordering defines another ordering for pebbling groups of outputs of the FFT portion of the graph. For the bit reversal connection pattern, we will see that these output groups are contained entirely within FFT subgraphs, while the identity permutation spreads

them out among different FFT subgraphs. This affects the time required to pebble the FFT graph in an adverse way.

We begin our detailed analysis of the FFT-tree and back-to-back FFT graphs by deriving upper bounds for the bit reversal connection pattern that match the lower bounds of Propositions 1 and 2.

THEOREM 1. *Let $n = 2^d$. The number of moves required to pebble the FFT-tree graph FFTTREE $[d, \text{BR}]$ when S pebbles are used satisfies $T = \Theta(n^2/S + n \log n)$.*

Proof. Noting that $S = \mathcal{O}(n)$ suffices to pebble the FFT-tree graph, the lower bound $T = \Omega(n^2/S + n \log n)$ follows from Proposition 1, and thus we only have to prove a matching upper bound of the form $T = \mathcal{O}(n^2/S + n \log n)$. The key point in a pebbling strategy for FFTTREE $[d, \text{BR}]$ arises when the inputs to a subtree in the subgraph TREE $[d]$ are considered. If these inputs are followed back to outputs of the subgraph FFT $[d]$, then it can be observed that the bit reversal pattern connecting the subgraphs TREE $[d]$ and FFT $[d]$ results in a block of consecutive outputs of FFT $[d]$ that must be pebbled. This is clear from Figure 3(b) and the discussion at the very end of § 2. Since these outputs appear consecutively, they either can be considered to be part of a subgraph $A_l[j]$ in FFT $[d]$ or composed of a multitude of subgraphs $A_l[j]$ in FFT $[d]$. In either case, it is obvious that the pebbling strategy due to Savage and Swamy [11] can be used on FFT $[d]$ to pebble its outputs. Since the number of moves made on TREE $[d]$ is of a lower order than the number made on FFT $[d]$, it follows that $T = \mathcal{O}(n^2/S + n \log n)$ for FFTTREE $[d, \text{BR}]$. $\square$

THEOREM 2. *Let $n = 2^d$. The number of moves required to pebble the back-to-back FFT graph BBFFT $[d, \text{BR}]$ when S pebbles are used satisfies*

$$T = \Theta(n^3/S^2 + n^2 \log n/S).$$

Proof. The proof is similar to that of Theorem 1. Details are left to the reader.

We note here that the upper bounds presented in Theorems 1 and 2 are slightly inaccurate when minimum space is used to pebble an FFT-tree or a back-to-back FFT graph. For example, the FFT-tree graph has a minimum space requirement of $2 \log n - 1$, and when this much space is used, time T is $\Omega(n^2)$, not $\mathcal{O}(n^2/\log n)$. However, the bounds do hold for values of space that are only a constant factor times the graph's minimum space requirement. Again for the FFT-tree graph, it can be pebbled in time $T = \mathcal{O}(n^2/\log n)$ using space $3 \log n$ by employing Savage and Swamy's strategy [11] on the FFT portion of the graph to pebble blocks of $\log n$ outputs.

Theorems 1 and 2 show that there is a connection pattern for which the lower bounds of Propositions 1 and 2 can be met. However, this may not be true for all connection patterns between subgraphs. We will verify this fact by showing that when the identity permutation is used as a connection pattern, an asymptotically stronger lower bound can be proved. For the graphs FFTTREE $[d, \text{ID}]$ and BBFFT $[d, \text{ID}]$, we begin our lower bound arguments by stating a preliminary result concerning how groups of inputs to the tree subgraph must be pebbled. The proof of this is long and involves detailed counting arguments, and thus is presented in the Appendix. Before its statement here as a lemma, we precisely define our notion of pebbling a group of vertices in a tree.

DEFINITION. Consider a set of disjoint complete binary trees. A "group" of vertices is a collection of vertices each in a different tree. The vertices are "pebbled as a group" if they are pebbled in sequence with no intervening moves on other vertices. The size of the group is the number of vertices it contains.

LEMMA 3. *Consider optimal strategies for pebbling 2^j disjoint complete binary trees each of depth k , $k = \omega(1)$, using $S \leq 2^j$ pebbles. At most 2^{j+k-1} of the 2^{j+k} leaves of these 2^j trees are "pebbled as groups" of size $\geq 2^{j+2}/(k-1)$ by any of these strategies.*

Proof. See the Appendix.

We now state and prove a lower bound for the graph FFTTREE $[d, \text{ID}]$ that is asymptotically stronger for a range of space than the upper bound of Theorem 1.

THEOREM 3. *Let $n = 2^d$. The number of moves required to pebble the FFT-tree graph FFTTREE $[d, \text{ID}]$ when S pebbles are used satisfies $T = \Omega((n^2/S) \log \log (n/S))$.*

Proof. Before proceeding with a detailed analysis of the time required to pebble FFTTREE $[d, \text{ID}]$, we present an example to illustrate the concepts involved. Figure 4 shows FFTTREE $[4, \text{ID}]$ and a decomposition of the subgraph TREE $[4]$. We have isolated 4 disjoint binary subtrees, each of depth 2, in TREE $[4]$. The leaves of these subtrees are inputs to the tree portion of the graph, which are connected by the identity permutation to the outputs of the FFT portion of the graph.

If we pebble the 4 binary subtrees one at a time, then it is easily seen that the leaves of an individual subtree are connected to outputs in each of the 4 separate subgraphs $A_l [2]$ in the FFT graph (the identity connection pattern causes this phenomenon).

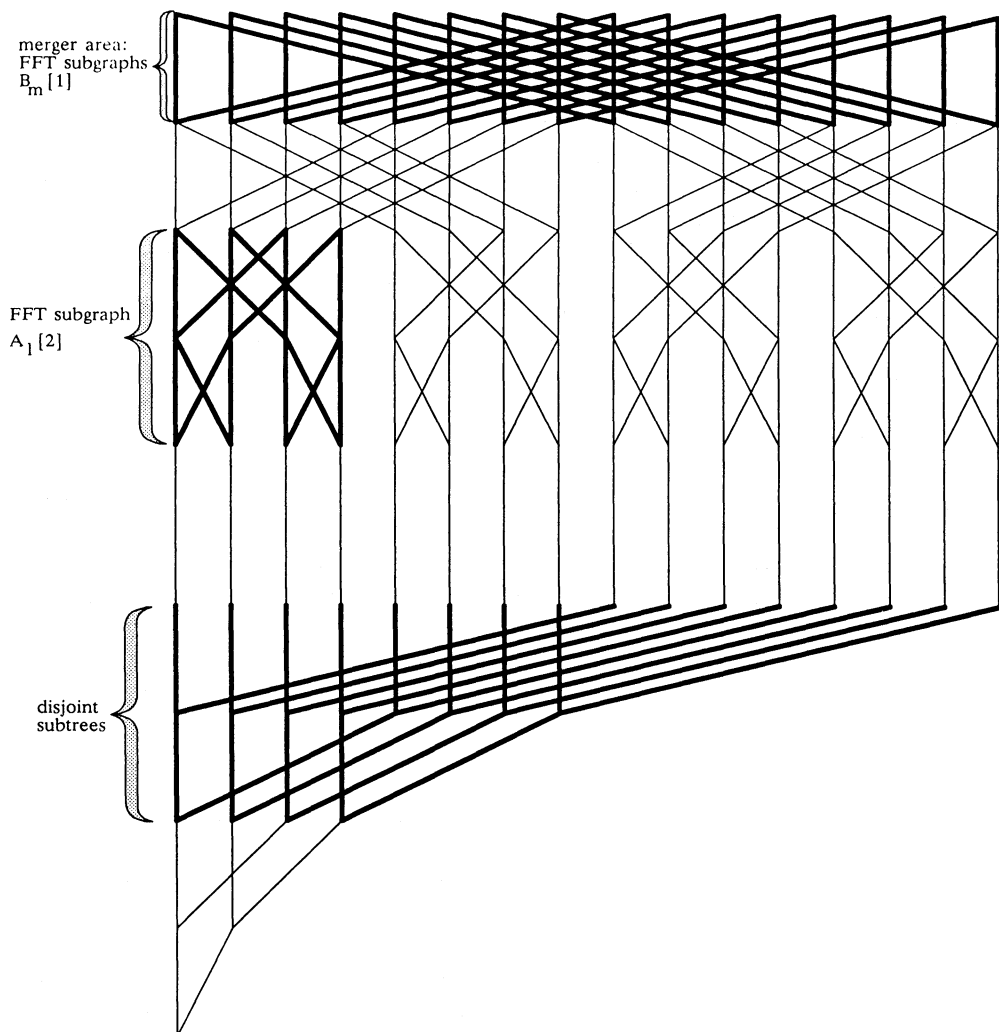


FIG. 4. Decomposition of FFTTREE $[4, \text{ID}]$.

This would force the pebbling of a large “merger area” (all vertices above the subgraphs A_i [2]) in the FFT graph for each of the trees. To reduce the work involved, we can attempt to pebble all 4 subtrees as a group. This would allow sets of 4 leaves (each in a separate tree) to be pebbled as a group, which is convenient because such a set of leaves connect to all outputs of a single subgraph A_i [2] in FFT [4]. Thus, the entire FFT graph can be pebbled much more efficiently when this is done.

However, there is a problem with attempting to pebble the 4 subtrees as a group that arises due to the space required to pebble each subtree. In our example, there are 4 subtrees and each has depth 2, so that a total of $4 \cdot 3 = 12$ pebbles are required to pebble them as a group. This is a factor of 3 increase over the input size of A_i [2] in the FFT portion of the graph, and in general will be a nonconstant factor increase over the space available to pebble the FFT-tree graph.

A realistic strategy that can be achieved is to pebble the set of subtrees in the tree portion of the graph in smaller groups. In our example, we could form 2 groups of 2 subtrees each, so that only 6 pebbles are required to pebble each group of 2 subtrees. This has a detrimental effect on the time required to pebble the FFT portion of the graph, because consecutive outputs are no longer contained in individual subgraphs A_i [2]. To pebble 2 subtrees as a group, it becomes necessary to pebble a merger area in FFT [4] consisting of all subgraphs B_m [1], as shown in Fig. 4. This merger area (along with some of the vertices in the subgraphs A_i [2]), is repebbled for each group of 2 subtrees pebbled in the course of this strategy.

The example presented above can be generalized to derive the lower bound of Theorem 3. We assume that $S = \Theta(2^j)$ pebbles are used to pebble FFTTREE $[d, \text{ID}]$ and consider the amount of work required to pebble the root of the tree portion of the graph. To do this, we must pebble all outputs of the FFT portion of the graph using at most S pebbles. We also have at most S pebbles available to pebble the tree, and we will apply Lemma 3 to identify key properties concerning how the tree must be pebbled.

To apply Lemma 3, we decompose the tree portion of the graph into $\Theta(S)$ subtrees each of depth $\Theta(d - \log S) = \Theta(d - j)$. We will discuss proper choices for the constants relating these quantities later in this proof. Lemma 3 with $k = \Theta(d - j)$ and $\Theta(S) = 2^j$ then allows us to conclude that at most half of the inputs to the tree may be “pebbled as groups” of size $\geq 2^{j+2}/(k - 1) = \Omega(S/(d - j))$; hence at least half must be pebbled as groups of size $\leq 2^{j+2}/(k - 1) = \mathcal{O}(S/(d - j))$. Again, a group is defined as a set of inputs, each being in a separate subtree, that are pebbled in a unified manner, i.e., with no intervening moves on other vertices of the tree. Looking at a group of leaves of size $\mathcal{O}(S/(d - j))$, they are connected directly to outputs of the FFT portion of the graph. The best situation occurs when these outputs are consecutively numbered, i.e., they form the outputs of an FFT subgraph $A_i[\Theta(\log(S/(d - j)))]$ with $\Theta(S/(d - j))$ outputs. Otherwise, more work is required to pebble such a set of outputs, as the merger area introduced earlier in this proof and formalized below expands in size. We further note that in an optimal strategy for pebbling the graph FFTTREE $[d, \text{ID}]$, outputs of the FFT portion of the graph and all vertices of the tree portion of the graph are pebbled only once, i.e., they are never repebbled (it is straightforward to show that a strategy that repebbles one of these vertices can be improved upon). Thus, each leaf of the tree belongs to at most one group that is either large (has size $\geq 2^{j+2}/(k - 1)$) or small (has size $\leq 2^{j+2}/(k - 1)$) in an optimal strategy.

We now consider the amount of work that must be done on the FFT portion of the FFT-tree graph when a set of $\Theta(S)$ of its outputs is pebbled. The discussion immediately above shows that such a set of outputs is composed of the outputs of $\Theta(d - j)$ FFT subgraphs $A_i[\Theta(\log(S/(d - j)))]$, all of which are disjoint; otherwise, it is not possible to

make progress pebbling the tree portion of the FFT-tree graph. Furthermore, each FFT subgraph $A_l[\Theta(\log(S/(d-j)))]$ is contained in a larger FFT subgraph

$$A_l[d - \Theta(\log(d-j))].$$

We note here that there are $\Theta(d-j)$ FFT subgraphs $A_l[d - \Theta(\log(d-j))]$ contained in $\text{FFT}[d]$. The above statements may not hold for every set of $\Theta(S)$ outputs, but Lemma 3 guarantees that they are true for a large enough proportion of them.

The structure of the graph $\text{FFT}[d]$ as outlined in Lemma 1 can now be used to show the existence of a merger area that must be pebbled in order to pebble the set of $\Theta(S)$ outputs of $\text{FFT}[d]$ in question. We define the merger area formally as the set of vertices in $\text{FFT}[d]$ for which paths can be found to two or more of the $\Theta(S)$ outputs of $\text{FFT}[d]$ being pebbled. From Lemma 1, $\text{FFT}[d]$ can be represented as the composition of $\Theta(d-j)$ disjoint FFT subgraphs $A_l[d - \Theta(\log(d-j))]$ with $\Theta(2^d/(d-j))$ disjoint FFT subgraphs $B_m[\Theta(\log(d-j))]$. Furthermore, with the exception of their outputs, each vertex of an FFT subgraph $B_m[\Theta(\log(d-j))]$ connects to at least two inputs of disjoint FFT subgraphs $A_l[d - \Theta(\log(d-j))]$, which obviously have a path to one of the $\Theta(S)$ outputs being pebbled. Thus, the merger area consists of all vertices except outputs of the disjoint FFT subgraphs $B_m[\Theta(\log(d-j))]$, and therefore has size (number of vertices) $\Theta(2^d \log(d-j))$. Figure 5 illustrates the decomposition of $\text{FFTTREE}[d, \text{ID}]$ that leads to these observations.

At this point, we note that up to S pebbles may be distributed over the vertices of $\text{FFTTREE}[d, \text{ID}]$ when we start making progress towards pebbling a new set of $\Theta(S)$ outputs of $\text{FFT}[d]$. By choosing an appropriate constant for the size of the output set ($4S$ is large enough) and using arguments similar to those presented in Savage and Swamy [11] and Carlson [3]–[5], the initial distribution of pebbles on the graph can be accounted for. Even when it is taken into account, a subset of the merger area of size $\Theta(2^d \log(d-j))$ must be pebbled for each set of $\Theta(S)$ outputs. Details of these arguments are tedious and straightforward, so we leave them to the interested reader.

We now summarize the above observations to obtain the proof of Theorem 3. Lemma 3 shows that $\Omega(n/S)$ sets of $\Theta(S)$ outputs in $\text{FFT}[d]$ must be pebbled as groups of size $\Theta(S/\log(n/S))$, each being in one of $\Theta(\log(n/S))$ disjoint FFT subgraphs of input size $\Theta(n/\log(n/S))$. To pebble such a set of outputs, a merger area of size $\Theta(n \log \log(n/S))$ must be pebbled. Thus, we conclude that

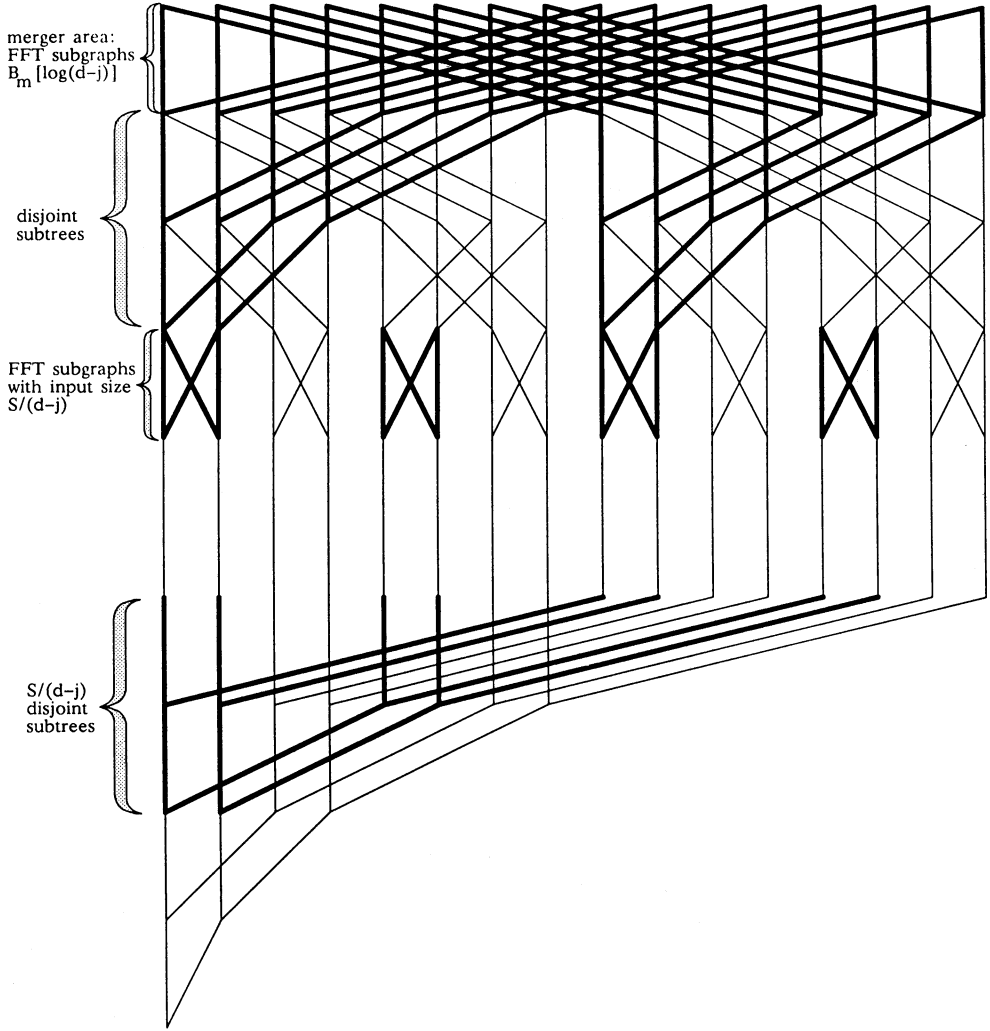
$$T = \Omega((n^2/S) \log \log(n/S)). \quad \square$$

A statement analogous to that of Theorem 3 can be made for the back-to-back FFT graph with the identity connection pattern. Its proof relies on the fact that an FFT-tree graph with the identity connection pattern is embedded in such a back-to-back FFT graph. For each set of $\Theta(S)$ outputs of the back-to-back FFT graph, an appropriate FFT-tree graph rooted at one of the outputs must be pebbled starting from a given initial configuration of pebbles on the graph. We come to the following conclusion.

THEOREM 4. *Let $n = 2^d$. The number of moves required to pebble the back-to-back FFT graph $\text{BBFFT}[d, \text{ID}]$ when S pebbles are used satisfies*

$$T = \Omega((n^3/S^2) \log \log(n/S)).$$

The bounds presented in Theorems 3 and 4 clearly illustrate that the connection pattern influences the time-space tradeoff characteristics of the FFT-tree and the back-to-back FFT graphs. For example, when $S = \Theta(n/\log n)$, $T = \Omega(n(\log n) \log \log \log n)$ for $\text{FFTTREE}[d, \text{ID}]$ whereas $T = \mathcal{O}(n \log n)$ for $\text{FFTTREE}[d, \text{BR}]$, and $T = \Omega(n(\log n)^2 \log \log \log n)$ for $\text{BBFFT}[d, \text{ID}]$ whereas $T = \mathcal{O}(n(\log n)^2)$ for $\text{BBFFT}[d, \text{BR}]$. When

FIG. 5. Decomposition of FFTTREE $[d, \text{ID}]$.

$S = \Theta(n^\epsilon)$, $0 < \epsilon < 1$, $T = \Omega(n^{2-\epsilon} \log \log n)$ for FFTTREE $[d, \text{ID}]$ whereas $T = \mathbf{O}(n^{2-\epsilon})$ for FFTTREE $[d, \text{BR}]$, and $T = \Omega(n^{3-2\epsilon} \log \log n)$ for BBFFT $[d, \text{ID}]$ whereas $T = \mathbf{O}(n^{3-2\epsilon})$ for BBFFT $[d, \text{BR}]$. In both cases, the time required for FFTTREE $[d, \text{ID}]$, BBFFT $[d, \text{ID}]$ is a nonconstant factor increase over that required for FFTTREE $[d, \text{BR}]$, BBFFT $[d, \text{BR}]$.

Upper bounds that match to within a constant factor the lower bounds of Theorems 3 and 4 are straightforward to demonstrate. They involve the use of pebbling strategies similar in spirit to the one presented in Theorem 1. Regarding this, we state the following proposition and leave the derivation of the associated pebbling strategies to the interested reader.

PROPOSITION 3. *Let $n = 2^d$. The number of moves required to pebble the FFT-tree graph FFTTREE $[d, \text{ID}]$ when S pebbles are used satisfies*

$$T = \mathbf{O}((n^2/S) \log \log (n/S) + n \log n).$$

The number of moves required to pebble the back-to-back FFT graph BBFFT $[d, \text{ID}]$ when S pebbles are used satisfies $T = \mathbf{O}((n^3/S^2) \log \log (n/S) + n^2 \log n/S)$.

4. Discussion and application of the results. We will show in this section that the graphs we have previously analyzed model important computational problems, and the results we have obtained with respect to their time-space tradeoff behavior become interesting when viewed in this context. The computational problems we consider here are combining together all elements of the resultant vector of an FFT to form a single result, and all convolutional operations via FFTs, including the multiplication of two n -degree polynomials.

A binary tree forms a natural representation for a computation that produces a single output from n distinct inputs. When we represent such a computation with a complete binary tree, and connect its inputs to the outputs of an FFT graph, we obtain a graph representation for an algorithmic solution to a problem where all elements of the Fourier transform of a vector are combined together to form a single result. Problems such as this arise frequently in digital signal processing applications [2]. For example, the power of a signal is the sum of squares of the elements of the transformed vector, and the resonant frequency of a signal corresponds to the element of the transformed vector of largest magnitude. Algorithms for both of these problems are naturally represented by an FFT-tree graph.

The FFT algorithm can be expressed either iteratively or recursively. A recursive expression corresponds to the graph representation defined by Savage and Swamy [11], where recursive subgraphs are composed together with a properly defined edge set. An iterative expression of the algorithm corresponds to the graph representation defined here in this paper; this is also discussed extensively in [1, Chap. 7].

Dependent on how the FFT algorithm is expressed (either iteratively or recursively), we can obtain different graph representations for an FFT-tree computation, as illustrated in Figs. 3(a) (recursive) and 3(b) (iterative). Both representations are valid means for computing the minimum or maximum of the resultant vector of an FFT. However, we have shown in § 3 that one method is clearly superior to the other in terms of its time-space tradeoff behavior. Basing reduced space algorithms on the iterative FFT algorithm followed by a complete binary tree as shown in Fig. 3 results in better efficiency than basing them on the recursive algorithm followed by a complete binary tree. This is because the iterative FFT algorithm followed by a complete binary tree is equivalent to FFTTREE [d , BR], whereas the recursive FFT algorithm followed by a complete binary tree is equivalent to FFTTREE [d , ID], which allows the results of § 3 to be applied. Designers of digital signal processing algorithms and software should be aware of the differences between these two possible implementations, and how this applies to the problems mentioned above.

The results of § 3 can also be applied to all convolutional operations via FFTs, including polynomial multiplication. Efficient $O(n \log n)$ algorithms for n -element convolution and n -degree polynomial multiplication have as their basis the FFT algorithm. They operate by transforming from a representation by coefficients to a representation as values at a set of points (forward transform), multiplying pairwise values at the same point, and then interpolating the resulting values (inverse transform) to obtain a representation as coefficients of the convolution/product polynomial. Using the FFT algorithm to compute the forward and inverse transforms gives an $O(n \log n)$ algorithm for convolution/polynomial multiplication (see [1, Chap. 7] for details).

From the points outlined above, it is easily seen that a graph representation for high-speed convolution/polynomial multiplication based on the FFT algorithm contains a back-to-back FFT graph as defined earlier. What is of interest is the connection pattern that exists between the FFT graphs. From the discussion in [1, Chap. 7], we have observed that it is described by the bit reversal permutation. Details leading to this observation are left to the reader.

Since the graph representation for high-speed convolution/polynomial multiplication contains BBFFT $[d, \text{BR}]$, the results of § 3 demonstrate that convolution/polynomial multiplication has the connection pattern between FFT graphs with the best possible time-space tradeoff behavior. This is true because BBFFT $[d, \text{BR}]$ can be pebbled in an amount of time that matches to within a constant factor the lower bound given in Proposition 2, whereas a back-to-back FFT graph with a different connection, BBFFT $[d, \text{ID}]$, was shown to require asymptotically more pebbling time.

5. Conclusions. The main results of this paper can be summarized as follows.

(1) An n -input FFT-tree graph whose connection pattern is specified by the bit reversal permutation has a time-space tradeoff behavior characterized by

$$T = \Theta(n^2/S + n \log n).$$

An n -input back-to-back FFT graph whose connection pattern is specified by the bit reversal permutation has a time-space tradeoff behavior characterized by

$$T = \Theta(n^3/S^2 + n^2 \log n/S).$$

(2) An n -input FFT-tree graph whose connection pattern is specified by the identity permutation has a time-space tradeoff behavior characterized by

$$T = \Theta((n^2/S) \log \log (n/S) + n \log n).$$

An n -input back-to-back FFT graph whose connection pattern is specified by the identity permutation has a time-space tradeoff behavior characterized by

$$T = \Theta((n^3/S^2) \log \log (n/S) + n^2 \log n/S).$$

The difference in these bounds for specific values of space serves to indicate the influence that the connection pattern can exert on an FFT-tree or a back-to-back FFT graph's simultaneous time-space requirements. Since graph representations for polynomial multiplication using the FFT algorithm have associated with them the bit reversal permutation, we can conclude that these polynomial multiplication algorithms have the best time-space tradeoff characteristics.

Appendix.

LEMMA 3. *Consider optimal strategies for pebbling 2^j disjoint complete binary trees each of depth k , $k = \omega(1)$, using $S \leq 2^j$ pebbles. At most 2^{j+k-1} of the 2^{j+k} leaves of these 2^j trees are "pebbled as groups" of size $\geq 2^{j+2}/(k-1)$ by any of these strategies.*

Proof. Let $\text{PP}(n, d, S)$ be the Problem of Pebbling n disjoint complete binary trees of depth d with $\leq S$ pebbles. The problem stated in Lemma 3 is then denoted by $\text{PP}(2^j, k, 2^j)$. Our technique in proving Lemma 3 is first to determine the maximum number of groups of leaves of size $2^j, 2^{j-1}, \dots$ that are pebbled in the course of solving $\text{PP}(2^j, k, 2^j)$ (groups of leaves of size r are counted in the smallest power of 2 greater than or equal to r). We then sum the total number of leaves in groups whose size is $\geq 2^{j+2}/(k-1)$ to obtain the desired conclusion.

It is obvious that "pebbling a group" of leaves of size 2^j exhausts all the available pebbles so that no further progress can be made in $\text{PP}(2^j, k, 2^j)$ when $k \geq 1$. Hence, no groups of 2^j leaves are pebbled in $\text{PP}(2^j, k, 2^j)$ when $k \geq 1$. Now, let us determine how many groups of 2^{j-1} leaves are pebbled in $\text{PP}(2^j, k, 2^j)$.

$\text{PP}(2^j, k, 2^j)$ can be decomposed into two subproblems $\text{PP}(2^{j-1}, k, 2^j)$ since the binary trees in $\text{PP}(2^j, k, 2^j)$ are all disjoint. It is obvious that at least one group of 2^{j-1} leaves can be pebbled in $\text{PP}(2^{j-1}, k, 2^j)$. Consider the first such group of 2^{j-1} leaves pebbled. Assume without loss of generality that each leaf is in the left subtree of the tree

in which it is contained. In an optimal strategy, at least one pebble will then remain on each left subtree until the root of the tree is pebbled. Thus, if we isolate the 2^{j-1} right subtrees, we have at most 2^{j-1} pebbles available with which to pebble them, i.e., we must solve PP $(2^{j-1}, k-1, 2^{j-1})$. From the previous argument, no groups of 2^{j-1} leaves can be pebbled in solving PP $(2^{j-1}, k-1, 2^{j-1})$ when $k-1 \geq 1$, i.e., $k \geq 2$. Isolating the left subtrees, we have actually formed the subproblem PP $(2^{j-1}, k-1, 2^j)$ since all 2^j pebbles are available when we start to pebble them. The argument now proceeds by induction until we reach PP $(2^{j-1}, 1, 2^j)$. There, at most two groups of 2^{j-1} leaves are pebbled. In all other subproblems encountered along the way to the base case, no groups of 2^{j-1} leaves are pebbled. In other words, once the first two groups of 2^{j-1} leaves are pebbled in the course of solving PP $(2^{j-1}, k, 2^j)$, there are at least 2^{j-1} pebbles that remain on subtrees in the structure, so it becomes impossible to pebble another group of 2^{j-1} leaves (no further progress could be made on the structure if this were done). Since a total of at most 2 groups of 2^{j-1} leaves are pebbled in each of the subproblems PP $(2^{j-1}, k, 2^j)$, a maximum of 4 groups of 2^{j-1} leaves are pebbled in PP $(2^j, k, 2^j)$, $k \geq 1$.

The above argument generalizes to show that at most

$$2^{i+1} \left[\binom{k-1}{2^i-2} + \cdots + \binom{k-1}{0} \right]$$

groups of 2^{j-i} leaves are pebbled in PP $(2^j, k, 2^j)$. The argument relies on splitting PP $(2^j, k, 2^j)$ into subproblems so as to maximize the number of groups of 2^{j-i} leaves, and then counts the maximum number of groups of 2^{j-i} leaves pebbled in each subproblem. Splitting PP $(2^j, k, 2^j)$ into 2^i subproblems PP $(2^{j-i}, k, 2^j)$ maximizes the number of groups of 2^{j-i} inputs that are pebbled since splitting PP $(2^j, k, 2^j)$ into subproblems of larger size may result in groups of leaves of larger size, which detracts from the total number of groups of size 2^{j-i} , and splitting it into subproblems of smaller size will result in no groups of size 2^{j-i} in these smaller sized subproblems. We note that in certain cases groups of size 2^{j-i} may possibly be combined together and thought of as a larger group.

Now, consider solving one of the subproblems PP $(2^{j-i}, k, 2^j)$ when $k \geq 2^i$. We will count the maximum number of subproblems PP $(2^{j-i}, 1, 2^{j-i+1})$ in PP $(2^{j-i}, k, 2^j)$ using an inductive argument and then multiply by 2 to obtain the maximum number of groups of 2^{j-i} leaves. Note that 2^{j-i+1} pebbles are needed to pebble 2^{j-i} trees of depth 1 if at least one group is pebbled, and that at most two groups of 2^{j-i} leaves are pebbled in the course of solving PP $(2^{j-i}, 1, 2^{j-i+1})$ (this accounts for the multiplicative factor of 2 mentioned above). We make the claim that there are at most

$$\binom{k-1}{2^i-2} + \cdots + \binom{k-1}{0}$$

subproblems PP $(2^{j-i}, 1, 2^{j-i+1})$ that occur in PP $(2^{j-i}, k, 2^j)$. To see this, we first discuss how the subproblem PP $(2^{j-i}, k, 2^j)$ is solved so that the number of groups of size 2^{j-i} is maximized.

Our discussion here regarding PP $(2^{j-i}, k, 2^j)$ parallels the previous one concerning PP $(2^{j-1}, k, 2^j)$. We identify the first “group” of 2^{j-i} leaves pebbled, and assume without loss of generality that each is in the left subtree of the containing tree. Since pebbles remain on these left subtrees until the roots of the trees are pebbled, the right subtrees are pebbled with at most $2^j - 2^{j-i} = (2^i - 1)2^{j-i}$ pebbles. The right subtrees thus form a subproblem PP $(2^{j-i}, k-1, (2^i - 1)2^{j-i})$, and similarly the left subtrees form a sub-

problem PP $(2^{j-i}, k-1, 2^j)$. It should be obvious that this decomposition into subproblems maximizes the number of groups of 2^{j-i} leaves pebbled.

Denoting the maximum number of subproblems PP $(2^{j-i}, 1, 2^{j-i+1})$ in PP $(2^{j-i}, k, 2^j)$ as $N(k, 2^i)$ (k represents the depth and 2^i the number of groups of 2^{j-i} pebbles available to solve the problem), the above discussion allows us to obtain the general recurrence

$$N(1, m) = \begin{cases} 0, & m = 1, \\ 1, & m > 1, \end{cases}$$

$$N(k, m) = N(k-1, m) + N(k-1, m-1).$$

It can be verified that this recurrence solves to

$$N(k, m) = \sum_{0 \leq h \leq m-2} \binom{k-1}{h}$$

by using double induction on each script of the recurrence. The argument is a straightforward one using standard combinatorial techniques [7], and so we leave the details to the interested reader. The proof of the claim stated above follows.

We now continue with the proof of Lemma 3. From the previous claim, at most $\sum_{0 \leq h \leq 2^i-2} \binom{k-1}{h}$ subproblems PP $(2^{j-i}, 1, 2^{j-i+1})$ are in each problem PP $(2^{j-i}, k, 2^j)$ and thus at most $2^{i+1} \sum_{0 \leq h \leq 2^i-2} \binom{k-1}{h}$ groups of 2^{j-i} leaves are “pebbled as a group” in the course of solving the problem PP $(2^j, k, 2^j)$. Thus, we can upper bound the number of groups of $\geq 2^{j-X}$ leaves that are pebbled as a group by

$$\sum_{0 \leq i \leq X} 2^{i+1} \sum_{0 \leq h \leq 2^i-2} \binom{k-1}{h}.$$

The total number of leaves that are pebbled in these groups is

$$\sum_{0 \leq i \leq X} 2^{j-i} 2^{i+1} \sum_{0 \leq h \leq 2^i-2} \binom{k-1}{h} = 2^{j+1} \sum_{0 \leq i \leq X} \sum_{0 \leq h \leq 2^i-2} \binom{k-1}{h}.$$

We now claim that choosing $X = \log(k-1) - 2$ yields the above sum to be a small fraction of the total number of leaves (which is 2^{j+k}). To see this, we further upper bound the above sum as follows, assuming $2^i \leq (k-1)/2$, which holds true when $X = \log(k-1) - 2$:

$$\begin{aligned} 2^{j+1} \sum_{0 \leq i \leq X} \sum_{0 \leq h \leq 2^i-2} \binom{k-1}{h} &\leq 2^{j+1} \sum_{0 \leq i \leq X} 2^i \binom{k-1}{2^i} \\ &\leq 2^{j+1} \binom{k-1}{2^X} \sum_{0 \leq i \leq X} 2^i \\ &\leq 2^{j+1} 2^{X+1} \binom{k-1}{2^X} \\ &= 2^{j+1} ((k-1)/2) \binom{k-1}{(k-1)/4}. \end{aligned}$$

An asymptotic analysis of the binomial coefficient $\binom{4m}{m}$ can be done to determine the behavior of the above formula. We will require the inequality $m! > (m/3)^m$, which is

easily verified by induction. Consequently,

$$\binom{4m}{m} = \frac{4m!}{m!3m!} < \frac{(4m)^m}{(m/3)^m} = 4^m \cdot 3^m = 2^{4m}(3/4)^m.$$

Thus, for $k = \omega(1)$,

$$2^{j+1}((k-1)/2) \binom{k-1}{(k-1)/4} < 2^{j+k} k (3/4)^{k/4} = o(2^{j+k}).$$

Therefore, at most $o(2^{j+k})$ of the 2^{j+k} total leaves to the trees are pebbled in groups of size larger than $2^{j+2}/(k-1)$. The statement of Lemma 3 immediately follows.

Acknowledgments. The author thanks both referees for a very careful reading of the paper and their numerous helpful comments.

REFERENCES

- [1] A. AHO, J. HOPCROFT, AND J. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.
- [2] R. E. BLAHUT, *Fast Algorithms for Digital Signal Processing*, Addison-Wesley, Reading, MA, 1985.
- [3] D. A. CARLSON, *Time-space and size-space tradeoffs for oblivious computations*, Tech. Rept. CS-63 (Ph.D. thesis), Dept. Computer Science, Brown University, Providence, RI, September 1980.
- [4] ———, *Time-space tradeoffs for back-to-back FFT algorithms*, IEEE Trans. Comput. C-32 (1983), pp. 585–589.
- [5] ———, *Time-space tradeoffs for back-to-back FFTs: the effect of the connection pattern*, Proc. 20th Allerton Conference, October 1982, pp. 374–383.
- [6] D. A. CARLSON AND J. E. SAVAGE, *Extreme time-space tradeoffs for graphs with small space requirements*, Inform. Process. Lett., 14 (1982), pp. 223–227.
- [7] D. E. KNUTH, *The Art of Computer Programming, Volume 1: Fundamental Algorithms, 2nd Edition*, Addison-Wesley, Reading, MA, 1973.
- [8] T. LENGAUER AND R. E. TARJAN, *Upper and lower bounds on time-space tradeoffs*, J. Assoc. Comput. Mach., 29 (1982), pp. 1087–1130.
- [9] M. S. PATERSON AND C. E. HEWITT, *Comparative schematology*, Proc. Proj. MAC Conference on Concurrent Systems and Parallel Computation, June 1970, pp. 119–127.
- [10] W. J. PAUL AND R. E. TARJAN, *Time-space tradeoffs in a pebble game*, Acta Inform., 10 (1978), pp. 111–115.
- [11] J. E. SAVAGE AND S. SWAMY, *Space-time tradeoffs on the FFT algorithm*, IEEE Trans. Inform. Theory, IT-24 (1978), pp. 563–568.
- [12] M. TOMPA, *Time-space tradeoffs for computing functions using connectivity properties of their circuits*, J. Comput. System Sci., 20 (1980), pp. 118–132.
- [13] ———, *Corrigendum: time-space tradeoffs for computing functions using connectivity properties of their circuits*, J. Comput. System Sci., 23 (1981), p. 106.
- [14] ———, *Two familiar transitive closure algorithms which admit no polynomial time, sublinear space implementations*, SIAM J. Comput., 11 (1982), pp. 130–137.

OPTIMAL DETECTION OF TWO DEFECTIVES WITH A PARITY CHECK DEVICE*

X. M. CHANG†, F. K. HWANG‡, AND J. F. WENG§

Abstract. Recently the problems of detecting two defectives under various testing models have been studied as a forum for exploring the boundaries between easy problems and hard problems in combinatorial searching. In this paper five levels of optimality to measure the difficulty are defined. These five levels are minimax, almost surely minimax, measure p minimax, first-order minimax, and order-of-magnitude minimax. A new procedure is given, which is almost surely minimax for the parity check model, improving the previously best procedure which is measure p minimax.

Key words. counterfeit coins, group testing, search procedures

AMS(MOS) subject classifications. 90B40, 05A05

1. Introduction. Recently Hwang [2] proposed the study of various testing models for detecting two defectives as a forum for exploring the boundaries between easy problems and hard problems in combinatorial search theory. The criterion used to evaluate a procedure A for model m is the worst-case number of tests required to detect the two defectives among n items, denoted by $A_m(n)$. Define $T_m(n) = \min_A A_m(n)$. We define five levels of optimality.

- (i) A is called *minimax* if $A_m(n) = T_m(n)$ for every n .
- (ii) A is called *almost surely minimax* if $A_m(n) = T_m(n)$ except for a set N of n -values such that $|N|/n \rightarrow 0$ as n gets large.
- (iii) A is called *minimax with measure p* if $A_m(n) = T_m(n)$ for p percent of the n -values.
- (iv) A is called *first-order minimax* if the first-order terms of $A_m(n)$ and $T_m(n)$ are the same.
- (v) A is called *order-of-magnitude minimax* if the first-order terms of $A_m(n)$ and $T_m(n)$ are of the same order.

In the above-mentioned paper all possible testing devices for the two-defective problems are classified into seven models labeled by B (basic), C (candy factory), D (degrees of defectiveness), G (group testing), P (parity), Q (quantity), and U (underweight). The following shows the states of the currently best procedures for the seven models:

minimax	almost surely	measure p	first-order term	order-of-magnitude.
B, C		G, P		D, Q, U

In this paper we give a procedure for model P which is almost surely minimax. This algorithm is similar to the minimax procedure for model C [1], which can be viewed as a generalization of the halving procedure that solves the one-defective problem so effectively. We show that the proposed procedure is minimax for every n except for a set U of density at most $\log_2 n/(2n)$ for which the proposed procedure can use one more test than necessary. Several interesting number-theoretic questions are posed for the unsettled set U .

2. The halving procedure. For model P the feedback to a test on a set S of items is the parity, odd or even, of the number of defectives contained in S . Let $[a \times b]$ denote

* Received by the editors June 6, 1986; accepted for publication (in revised form) September 23, 1987.

† Nankai University, Tienjien, China.

‡ AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

§ Baoshan General Steel Works, Shanghai, China.

the situation where two disjoint sets with a and b items each contains a defective. Let $[\sum_{i=1}^k a_i]$ denote the situation where one of the k disjoint sets with $a_1, \dots, a_k$ items, respectively, contains the two defectives. Finally, let $[\sum_{i=1}^k (a_i \times b_i)]$ denote the situation where there are $2k$ disjoint sets with $a_1, b_1, \dots, a_k, b_k$ items, respectively, and the two defectives are contained in one $[a_i \times b_i]$. Any pair of items which might be the pair of defectives in a given situation is called a *candidate pair*. The number of candidate pairs of a situation is called its *cardinality*. Clearly, the cardinality of $[\sum_{i=1}^k a_i]$ is $\sum_{i=1}^k \binom{a_i}{2}$ and the cardinality of $[\sum_{i=1}^k (a_i \times b_i)]$ is $\sum_{i=1}^k (a_i \times b_i)$. For convenience we will sometimes *not* distinguish a node from its situation.

We will represent a procedure by a rooted binary tree where each node is labeled by the current situation, i.e., the situation obtained by removing all the candidate pairs inconsistent with the test feedbacks from the starting situation. Each internal node is also associated with a test for the current situation with the left link leading to the new situation after an odd feedback and the right link after an even feedback. A terminal node is a node labeled by a situation with cardinality one, i.e., the pair of defectives is identified and no additional test is needed.

We now define the halving procedure. Let $\lfloor x \rfloor$ ($\lceil x \rceil$) denote the largest (smallest) integer not larger (smaller) than x . In the situation $[\sum_{i=1}^k a_i]$, let the next test consist of $\lfloor a_i/2 \rfloor$ items from each a_i . With the odd feedback the succeeding situation is of the type $[\sum_i (a_i \times b_i)]$; with the even feedback, it is of the type $[\sum_i a_i]$. In the situation $[\sum_{i=1}^k (a_i \times b_i)]$, let the next test consist of c_i of the a_i items and d_i of the b_i items from each a_i and b_i such that $\sum_{i=1}^k c_i d_i + \sum_{i=1}^k (a_i - c_i)(b_i - d_i) = \lfloor \sum_{i=1}^k a_i b_i / 2 \rfloor$. Then regardless of the feedback, the succeeding situation is always of the type $[\sum_i (a_i \times b_i)]$. We now prove that the integers c_i and d_i needed for the situation $[\sum_i (a_i \times b_i)]$ always exist.

LEMMA 1. *There exist integers $0 \leq c \leq a$ and $0 \leq d \leq b$ such that*

$$cd + (a - c)(b - d) = \lfloor ab/2 \rfloor.$$

Proof. If ab is odd, take $c = \lfloor a/2 \rfloor$ and $d = \lceil b/2 \rceil$. If ab is even, say a is even, take $c = a/2$ and $d = 0$.

LEMMA 2. *Suppose that $a_1 b_1$ and $a_2 b_2$ are both odd. Then there exist $0 \leq c_1 \leq a_1$, $0 \leq d_1 \leq b_1$, $0 \leq c_2 \leq a_2$, $0 \leq d_2 \leq b_2$ such that*

$$c_1 d_1 + c_2 d_2 + (a_1 - c_1)(b_1 - d_1) + (a_2 - c_2)(b_2 - d_2) = (a_1 b_1 + a_2 b_2)/2.$$

Proof. Take $c_1 = \lfloor a_1/2 \rfloor$, $d_1 = \lfloor b_1/2 \rfloor$, $c_2 = \lfloor a_2/2 \rfloor$ and $d_2 = \lceil b_2/2 \rceil$.

LEMMA 3. *There exist integers $0 \leq c_i \leq a_i$ and $0 \leq d_i \leq b_i$, $i = 1, \dots, k$ such that $\sum_{i=1}^k c_i d_i + \sum_{i=1}^k (a_i - c_i)(b_i - d_i) = \lfloor \sum_{i=1}^k a_i b_i / 2 \rfloor$.*

Proof. Let $O = \{a_i b_i: a_i b_i \text{ is odd}\}$ and $E = \{a_i b_i: a_i b_i \text{ is even}\}$. Use Lemma 2 on pairs of elements in O , and use Lemma 1 on each element in E as well as the remaining one element in O if $|O|$ is odd.

Figure 1 illustrates a typical halving tree where $d = \lceil \log_2 \binom{n}{2} \rceil$.

3. The optimality of the halving procedure. Let $H(n)$ denote the binary tree of the halving procedure with n items. A node is at *level* l if its distance to the root is l (the root is a level 0 node). Define $d = \lceil \log_2 \binom{n}{2} \rceil$. A level- l node is *regular* if the cardinality of the node does not exceed 2^{d-l} . Note that if all nodes of $H(n)$ are regular, then $H(n)$ has depth d , i.e., the worst-case number of tests for the halving procedure equals the information-theoretic lower bound (for any procedure with binary feedback). Hence $H(n)$ is minimax.

LEMMA 4. *Let v be a node on $H(n)$ labeled by the situation $[\sum_{i=1}^k a_i]$. Then:*

- (i) *The cardinality of its right son does not exceed the cardinality of its left son;*
- (ii) *The cardinality of the brother of v is at least twice the cardinality of the left son.*

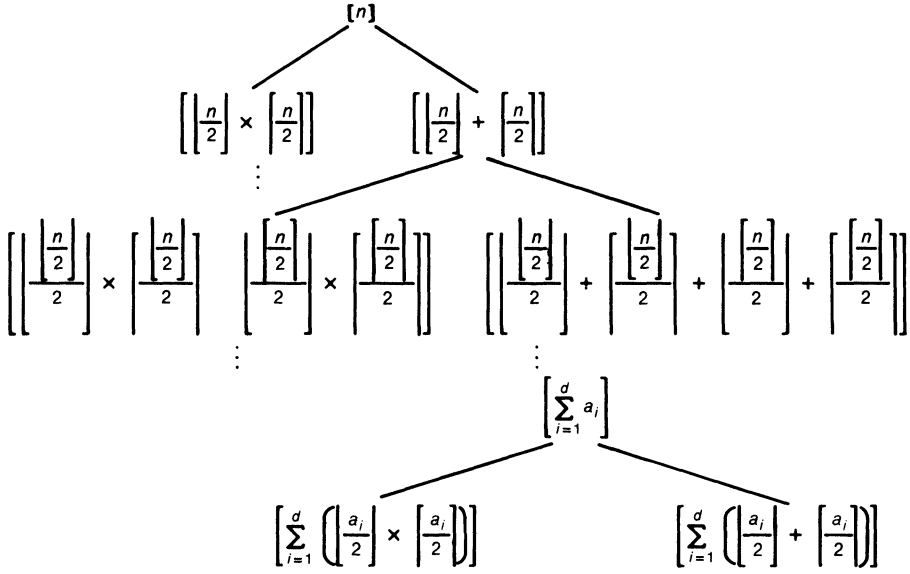


FIG. 1. A typical halving tree.

Proof. Note that for any positive integer a

$$\binom{\lfloor a/2 \rfloor}{2} + \binom{\lceil a/2 \rceil}{2} \leq 2 \binom{\lceil a/2 \rceil}{2} = \left\lfloor \frac{a}{2} \right\rfloor \left(\left\lceil \frac{a}{2} \right\rceil - 1 \right) \leq \left\lfloor \frac{a}{2} \right\rfloor \left\lfloor \frac{a}{2} \right\rfloor$$

and

$$\begin{aligned} 2 \left(\binom{\lfloor a/2 \rfloor}{2} + \binom{\lceil a/2 \rceil}{2} \right) &= \left\lfloor \frac{a}{2} \right\rfloor \left\lceil \frac{a}{2} \right\rceil - 1 \quad \text{for } a \equiv 2 \pmod{4} \\ &= \left\lfloor \frac{a}{2} \right\rfloor \left\lfloor \frac{a}{2} \right\rfloor \quad \text{otherwise.} \end{aligned}$$

Lemma 4 follows immediately.

THEOREM 1. If $\lfloor n/2 \rfloor \lceil n/2 \rceil \leq 2^{d-1}$, the $H(n)$ requires d tests and is minimax.

Proof. The nodes on $H(n)$ except the root can be classified into three types. Type R nodes consist of those on the rightmost path, having the form $\lceil \sum_{i=1}^k a_i \rceil$. Type L nodes consist of those that are left sons of type R nodes; and type D nodes consist of those that are descendants of some type L nodes. By Lemma 4(ii) the regularity of a level- i type L node implies the regularity of a level- $(i+1)$ type L node. Since the hypothesis of Theorem 1 specifies the regularity of the level-1 type L node $\lfloor \lfloor n/2 \rfloor \times \lceil n/2 \rceil \rfloor$, all type L nodes are regular. Since every type R node has a type L node as its left brother, by Lemma 4(i) the regularity of the latter implies the regularity of the former. Finally, by Lemma 3 the regularity of a type L node implies the regularity of all its descendants, hence all type D nodes are regular.

Let $H_P(n)$ denote the number of tests required by $H(n)$. Clearly, $H_P(n)$ and $T_P(n)$ are both nondecreasing in n since with $n-1$ items we can always throw in an additional good item and follow the procedure for n items. Therefore it makes sense to define $h(t)$ and $p(t)$ as the largest values of n for which $H_P(n) = t$ and $T_P(n) = t$, respectively. Note that $h(t) \leq p(t) \leq I(t)$ where $I(t)$ is the information-theoretic bound determined by

$$\log_2 \binom{I(t)}{2} \leq t < \log_2 \binom{I(t)+1}{2}.$$

THEOREM 2. (i) $h(2l+1) = p(2l+1) = I(2l+1)$ for $l = 0, 1, \dots$.

(ii) If $I(2l) = 2I(2l-1) = \dots = 2^j I(2l-j) < 2^{j+1} I(2l-j-1)$ for some $j \geq 0$ and $l \geq 3$, then $h(2l) = p(2l) = I(2l)$.

(iii) If $I(2l) = 2I(2l-1) = \dots = 2^j I(2l-j) > 2^{j+1} I(2l-j-1)$ for some $j \geq 0$ and $l \geq 1$, then $h(2l) = I(2l) - 1$.

Proof. (i) It is easily verified that $I(2l+1) = 2^{l+1}$. On the other hand, let $n = 2^{l+1}$ and $d = \lceil \log_2 \left(\frac{n}{2}\right) \rceil = 2l+1$. Then $\lfloor n/2 \rfloor \lceil n/2 \rceil = 2^{2l}$. By Theorem 1, $H_P(2^{l+1}) = 2l+1$ or $h(2l+1) = 2^{l+1} = I(2l+1)$.

(ii) It is easily verified that the positive real solution of y in the equation $\binom{y}{2} = 2^{2l}$ is $y = \sqrt{2^{2l+1} + \frac{1}{4}} + \frac{1}{2}$. Therefore $I(2l) = \lfloor y \rfloor$ and $d = \lceil \log_2 \left(\binom{2l}{2}\right) \rceil = 2l$. Since

$$\frac{I(2(l-j))}{2} < I(2(l-j-1)),$$

we have

$$\left\lceil \frac{I(2(l-j))}{2} \right\rceil \leq I(2(l-j-1))$$

and

$$\left\lfloor \frac{I(2(l-j))}{2} \right\rfloor \leq I(2(l-j-1)) - 1.$$

Hence

$$\left\lceil \frac{I(2(l-j))}{2} \right\rceil \left\lfloor \frac{I(2(l-j))}{2} \right\rfloor \leq 2 \binom{I(2(l-j-1))}{2} \leq 2^{2l-2j-1}.$$

If $I(2(l-j))$ is even, then clearly, $I^2(2(l-j)) \leq 2^{2l-2j+1}$. If $I(2(l-j))$ is odd, then since the product of two consecutive positive integers is never a power of two except for 1×2 ,

$$1 + \left\lceil \frac{I(2(l-j))}{2} \right\rceil \left\lfloor \frac{I(2(l-j))}{2} \right\rfloor \leq 2^{2l-2j-1} \quad \text{for } l-j > 1.$$

Thus $I^2(2(l-j)) \leq 2^{2l-2j+1}$ for $l-j > 1$. Let $n = I(2l)$ for $l \geq 3$ where $I(2l)$ satisfies the condition of (ii). Since $I(6) = 11 < 2I(4) = 12$, we have $l-j > 1$. Hence

$$\left\lfloor \frac{n}{2} \right\rfloor \left\lceil \frac{n}{2} \right\rceil = 2^{2(j-1)} I^2(2(l-j)) \leq 2^{2l-1}.$$

By Theorem 1, $h(2l) = I(2l)$.

(iii) Similarly, if

$$\frac{I(2(l-j))}{2} > I(2(l-j-1)),$$

then

$$\left\lfloor \frac{n}{2} \right\rfloor \left\lceil \frac{n}{2} \right\rceil = 2^{2(j-1)} I^2(2(l-j)) > 2^{2l-1}.$$

Hence $h(2l) < I(2l)$. On the other hand, let $n = I(2l) - 1$ and $d = \lceil \log_2 \left(\frac{n}{2}\right) \rceil = 2l$. Then

$$\left\lfloor \frac{n}{2} \right\rfloor \left\lceil \frac{n}{2} \right\rceil \leq \frac{I(2l)-1}{2} \cdot \frac{I(2l)}{2} = \frac{1}{2} \binom{I(2l)}{2} \leq 2^{2l-1}.$$

By Theorem 1, $h(2l) = I(2l) - 1$.

Define $U = \{n: n = I(2l) = 2I(2(l-1)) = \dots = 2^j I(2(l-j)) > 2^{j+1} I(2(l-j-1)) \text{ for some } j \geq 0\}$.

COROLLARY. $H(n)$ is minimax except for $n \in U$, then $H_P(n) \leq T_P(n) + 1$.

4. Some questions about the set U . The first question is can we determine U explicitly? For $l \leq 20$ those l -values such that $I(2l) \in U$ are 2, 4, 6, 12, 15, 16, 17, 18 (the corresponding n values are 6, 23, 91, 5793, 46341, 92682, 185364, 370728). Can we estimate the proportion of l such that $I(2l) \in U$? Here we give a proof that there are infinitely many $I(2l) \notin U$.

THEOREM 3. *There exist infinitely many $I_{2l} \notin U$.*

Proof. Suppose to the contrary that there exists an $I(2(k+1)) = 2I(k) + 1$ such that for all $l \geq k$, $I(2(l+1)) \geq 2I(2l)$.

Let $y_{2l} = \sqrt{2^{2l+1}} + \frac{1}{4} + \frac{1}{2}$ be the positive solution of $(\frac{y}{2}) = 2^{2l}$. Define

$$\varepsilon_l = \sqrt{2^{2l+1}} + \frac{1}{4} - \sqrt{2^{2l+1}}$$

and $n_l = \sqrt{2^{2l+1}} - \lfloor \sqrt{2^{2l+1}} \rfloor$. Then $0 < \varepsilon_l < \frac{1}{2}$, $\varepsilon_l \xrightarrow{l \rightarrow \infty} 0$, $0 < n_l < 1$, and

$$y_{2l} = \lfloor \sqrt{2^{2l+1}} \rfloor + n_l + \varepsilon_l + \frac{1}{2}.$$

Since

$$\sqrt{2^{2(l+1)+1}} = 2\sqrt{2^{2l+1}} = 2\lfloor \sqrt{2^{2l+1}} \rfloor + 2n_l,$$

$$y_{2(l+1)} = 2\lfloor \sqrt{2^{2l+1}} \rfloor + 2n_l + \varepsilon_{l+1} + \frac{1}{2}$$

and

$$y_{2(l+m)} = 2^m \lfloor \sqrt{2^{2l+1}} \rfloor + 2^m n_l + \varepsilon_{l+m} + \frac{1}{2}.$$

Note that $I(2(k+1)) = 2I(k) + 1$ implies $I(2k) = \lfloor \sqrt{2^{2k+1}} \rfloor$ and $n_k < \frac{1}{2}$, for if $I(2k) = \lfloor \sqrt{2^{2k+1}} \rfloor + 1$, then $I(2(k+1)) = 2\lfloor \sqrt{2^{2k+1}} \rfloor + 3 \geq \lfloor \sqrt{2^{2k+3}} \rfloor + 2$, a contradiction. Now

$$y_{2(k+m)} = 2^m \lfloor \sqrt{2^{2k+1}} \rfloor + 2^m n_k + \varepsilon_{k+m} + \frac{1}{2}$$

$$\geq 2^{m-1} y_{2(k+1)} = 2^m y_{2k} + 2^{m-1}$$

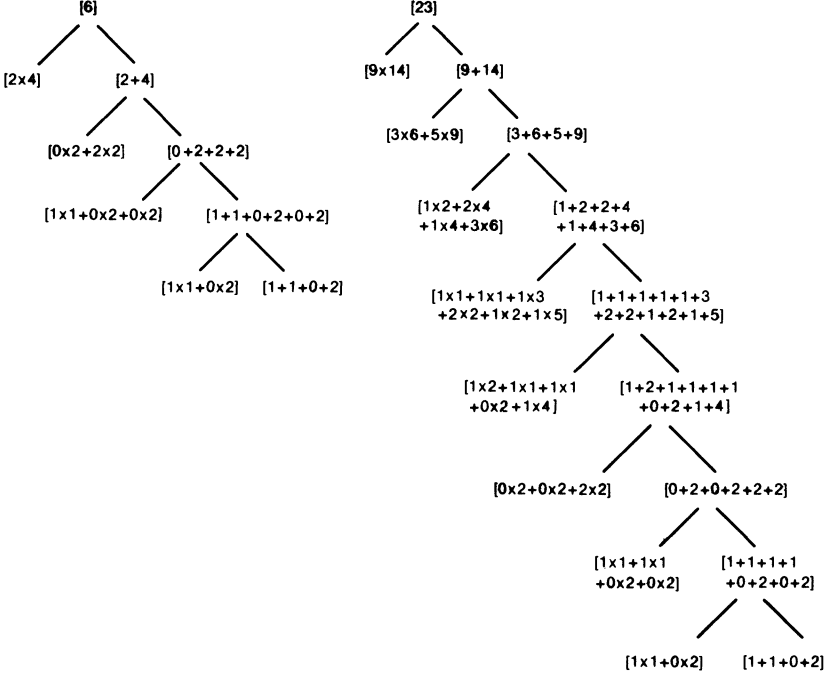
$$\geq 2^m \lfloor \sqrt{2^{2k+1}} \rfloor + 2^{m-1} \quad \text{by the contrary assumption}$$

or

$$(\varepsilon_{k+m} + \frac{1}{2})/2^m > \frac{1}{2} - n_k.$$

Since $n_k < \frac{1}{2}$ is fixed and $\varepsilon_{k+m} + \frac{1}{2} < 1$, the inequality cannot hold for m large enough. Therefore the contrary assumption is wrong, i.e., there exists $l \geq k$ such that $I(2(l+1)) < 2I(2l)$.

For $n = I(12) = 91$, $(\frac{n}{2}) = 4095 = 2^{12} - 1$. It is easily verified that there does not exist an x , $0 \leq x \leq n$, such that $\max \{x(n-x), (\frac{n}{2}) - x(n-x)\} \leq 2^{2l-1}$. Hence $p(12) = I(12) - 1 = h(12)$. Yet for other values of $I(2l)$ there exists x such that $\max \{x(n-x), (\frac{n}{2}) - x(n-x)\} \leq 2^{2l-1}$. For example, $x = 2$ for $n = I(4) = 6$ and $x = 9$ for $n = I(8) = 23$. Furthermore, Fig. 2 shows procedures with these initial tests that achieve the information-theoretic bound. How do we determine for which $I(2l) \in U$ the desirable x exists? Is it true that whenever such an x exists then there always exists a procedure achieving the information-theoretic bound? These are interesting questions to be studied in the future (see the Appendix).


 FIG. 2. Two procedures for $n = 6$ and $n = 23$.

In conclusion we note that even if $I(2l) \in U$ for all integer l , the density of U is still only $(\log_2 n)/2n$ which tends to zero as n gets large. Hence the halving procedure is indeed almost surely minimax.

Appendix. Recently, P. Erdős (through private communication) pointed out that for n large enough there always exists an x such that

$$\max \{x(n-x), \binom{n}{2} - x(n-x)\} \leq 2^{2l-1}.$$

His remarks, with minor editing, are as follows:

From a theorem of Roth [3], Schinzel deduced that the number of solutions of

$$2^{2l} - \binom{n}{2} < n^{1-\epsilon}$$

is finite for every $\epsilon > 0$. So in particular we have

$$2^{2l} - \binom{n}{2} \geq 2\sqrt{n} + 2 \quad \text{for every } n > n_0.$$

Consider $n > n_0$. For convenience we assume that n is even. Suppose first that $n^2/4 \leq 2^{2l-1}$. Set $x = n/2$. Clearly

$$x(n-x) = \left(\frac{n}{2}\right)^2 = \left(\frac{n^2}{4}\right) \leq 2^{2l-1}$$

and

$$\binom{n}{2} - x(n-x) = \frac{n^2 - n}{2} - \frac{n^2}{4} = \frac{n^2}{4} - \frac{n}{2} \leq 2^{2l-1}.$$

Suppose that $n^2/4 > 2^{2l-1}$. Note that

$$\frac{n^2}{4} - 2^{2l-1} = \frac{1}{2} \left[\binom{n}{2} - 2^{2l} \right] + \frac{n}{4} < \frac{n}{4}.$$

Set $x = n/2 + \lceil \sqrt{n}/2 \rceil$ where $\lceil x \rceil$ is the smallest integer not less than x . Then

$$x(n-x) = \frac{n^2}{4} - \left\lceil \frac{\sqrt{n}}{2} \right\rceil^2 \leq 2^{2l-1}.$$

Furthermore

$$x(n-x) \leq \frac{n^2}{4} - \left(\frac{\sqrt{n}}{2} + 1 \right)^2 = \frac{n^2}{4} - \frac{n}{4} - \sqrt{n} - 1.$$

Hence

$$\binom{n}{2} - x(n-x) \leq \frac{n^2 - n}{2} - \left(\frac{n^2}{4} - \frac{n}{4} - \sqrt{n} - 1 \right) = \frac{1}{2} \left[\binom{n}{2} + 2\sqrt{n} + 2 \right] \leq 2^{2l-1} \quad \text{for } n > n_0.$$

REFERENCES

- [1] C. CHRISTEN, *Optimal detection of two complementary defectives*, SIAM J. Algebraic Discrete Methods, 4 (1983), pp. 101–110.
- [2] F. K. HWANG, *A tale of two coins*, Amer. Math. Monthly, 94 (1987), pp. 121–129.
- [3] K. F. ROTH, *Rational approximations to algebraic numbers*, Mathematika, 2 (1955), pp. 1–20; Corrigendum, p. 168.

ON THE FRACTIONAL COVERING NUMBER OF HYPERGRAPHS*

F. R. K. CHUNG†, Z. FÜREDI‡, M. R. GAREY§, AND R. L. GRAHAM§

Abstract. The *fractional covering number* τ^* of a hypergraph $H = (V, E)$ is defined to be the minimum possible value of $\sum_{x \in V} t(x)$ where t ranges over all functions $t: V \rightarrow \mathbb{R}$ which satisfy $\sum_{x \in e} t(x) \geq 1$ for all edges $e \in E$. In the case of ordinary graphs G , it is known that $2\tau^*(G)$ is always an integer. By contrast, it is shown (among other things) that for any rational $p/q \geq 1$, there is a 3-uniform hypergraph H with $\tau^*(H) = p/q$.

Key words. hypergraphs, covering number

AMS(MOS) subject classification. 05C65

1. Notation. A *hypergraph* $\mathcal{H}$ is a pair $(V(\mathcal{H}), E(\mathcal{H}))$ where $V(\mathcal{H})$ is a finite set (called *vertices*) and $E(\mathcal{H})$ is a family of subsets of $V(\mathcal{H})$ (called *edges*). The *rank* of $\mathcal{H}$ is the maximum size of an edge, $r(\mathcal{H}) := \max \{|E|: E \in E(\mathcal{H})\}$. If every edge has r elements then $\mathcal{H}$ is called an r -uniform hypergraph, or r -graph for short. The 2-uniform hypergraphs are called (simple) *graphs*. The *matching number* $\nu(\mathcal{H})$ of $\mathcal{H}$ is the maximum number of pairwise disjoint edges in $E(\mathcal{H})$, i.e.,

$$\nu(\mathcal{H}) = \max \{w: \text{there exists } E_1, \dots, E_w \in E(\mathcal{H}), E_i \cap E_j = \emptyset \text{ for } i \neq j\}.$$

The *covering number* $\tau(\mathcal{H})$ of $\mathcal{H}$ is the minimum cardinality of a *cover* T , where $T \subset V(\mathcal{H})$ is a cover if $T \cap E \neq \emptyset$ for all $E \in E(\mathcal{H})$. If $\emptyset \in E(\mathcal{H})$ then $\nu = \tau = \infty$. The great importance of these notions is supported by the fact that virtually all combinatorial problems can be reformulated as the determination of the covering or matching number of an appropriate hypergraph. The calculation of τ and ν for an arbitrary hypergraph is an NP-hard problem. Thus, any result that gives estimates, at least for a certain class of hypergraphs, is especially valuable. One of the simplest estimates can be obtained from the *linear programming* bound, in other words, from the real relaxations of τ and ν . A *fractional matching* of $\mathcal{H}$ is a function $w: E(\mathcal{H}) \rightarrow \mathbb{R}$ (i.e., a vector in $\mathbb{R}^{E(\mathcal{H})}$) satisfying $w(E) \geq 0$ for every edge $E \in E(\mathcal{H})$ and

$$\sum \{w(E): x \in E \in E(\mathcal{H})\} \leq 1 \quad \text{for every } x \in V(\mathcal{H}).$$

The *value* of the fractional matching w is defined to be $|w| = \sum \{w(E): E \in E(\mathcal{H})\}$. The maximum of $|w|$ when w ranges over all fractional matchings is called the *fractional matching number* and is denoted by

$$\nu^*(\mathcal{H}) = \max \{|w|: w \text{ is a fractional matching of } \mathcal{H}\}.$$

Similarly, the *fractional covering number* is the minimum value of fractional covers of $\mathcal{H}$, i.e.,

$$\tau^*(\mathcal{H}) = \min \left\{ \sum_{x \in V(\mathcal{H})} t(x): t: V(\mathcal{H}) \rightarrow \mathbb{R}, t(x) \geq 0, \text{ and } \sum_{x \in E} t(x) \geq 1 \text{ for all } E \in E(\mathcal{H}) \right\}.$$

The determination of the fractional matching and covering number is a linear programming problem. This is a dual pair, so by the Duality Principle of linear programming we have $\tau^*(\mathcal{H}) = \nu^*(\mathcal{H})$ for every hypergraph $\mathcal{H}$. In general for every $\mathcal{H}$ and for every

* Received by the editors March 9, 1987; accepted for publication (in revised form) September 11, 1987.

† Bell Communications Research, Inc., Morristown, New Jersey 07960.

‡ Massachusetts Institute of Technology, Cambridge, Massachusetts 02139. Present address, AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

§ AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

fractional cover t and matching w we have

$$|w| \leq |t|,$$

so that if we have $|w| = |t|$ then both are optimal.

A subhypergraph $\mathcal{H}'$ is formed by a subset of edges of $\mathcal{H}$, i.e., $E(\mathcal{H}') \subset E(\mathcal{H})$, $\cup \{E \in E(\mathcal{H}')\} \subset V(\mathcal{H}') \subset V(\mathcal{H})$.

2. Fractional matchings in graphs. Edmonds [E] pointed out that an old theorem of Tutte [T] implies that $2\tau^*(\mathcal{G})$ is always an integer for a graph $\mathcal{G}$. Balinski [B], Balinski and Spielberg [BS], and Nemhauser and L. Trotter [NT] proved that even much more is true. To state their results, define the *fractional matching polytope* of the hypergraph $\mathcal{G}$, denoted by $\text{FMP}(\mathcal{H})$, as the set of all fractional matching vectors in $\mathbb{R}^{|E(\mathcal{H})|}$, i.e.,

$$\text{FMP}(\mathcal{H}) = \{\mathbf{w} \in \mathbb{R}^{|E(\mathcal{H})|} : \{w(E)\}_{E \in E(\mathcal{H})} \text{ is a fractional matching of } \mathcal{H}\}.$$

Analogously, the fractional covering polytope (FCP) of $\mathcal{H}$ is

$$\text{FCP}(\mathcal{H}) = \{\mathbf{t} \in \mathbb{R}^{|V(\mathcal{H})|} : \{t(x)\}_{x \in V(\mathcal{H})} \text{ is a fractional cover of } \mathcal{H}\}.$$

These are obviously polyhedra. If we can effectively describe all their vertices and facets, then in a certain sense we can solve any optimization problem concerning fractional matchings and covers. This description was given in [B], [BS], and in [NT] (a discussion of this and more graph theoretical background can be found in Lovász [L79]). Their results imply:

- (2.1) all the vertices of the polytopes $\text{FMP}(\mathcal{G})$ and $\text{FCP}(\mathcal{G})$ have coordinate values which are 0, $\frac{1}{2}$, or 1.

3. 3-graphs with arbitrary denominator. It is obvious that for arbitrary hypergraphs, a statement similar to (2.1) is not true. For every rational number $r = p/q$ (≥ 1), there exists a hypergraph $\mathcal{H}$ with $\tau^*(\mathcal{H}) = p/q$ (e.g., the complete q -graph on p elements). Lovász [L75] proved that for every choice of integers $1 \leq \nu \leq \tau$ and rational number $r > 1$ satisfying $\nu \leq r \leq \tau$ there exists a hypergraph $\mathcal{H}$ with $\nu(\mathcal{H}) = \nu$, $\tau^*(\mathcal{H}) = r$, and $\tau(\mathcal{H}) = \tau$. (If $\nu = \tau = 1$ then necessarily $\tau = 1$.) However, his hypergraphs have large ranks. In this section we prove that a similar statement holds even for hypergraphs of rank 3. For a real number x , denote by $\{x\}$ its *fractional part*, i.e., $\{x\} = x - \lfloor x \rfloor$.

THEOREM 3.1. *Let $0 \leq r < 1$ be a rational number. Then there exists a hypergraph $\mathcal{H}$ of rank 3 with $\{\tau^*(\mathcal{H})\} = r$.*

For the proof we are going to use the following constructions.

Example 3.2. (A hypergraph of rank 3 with $4k + 2$ edges, and with $\{\tau^*\} = 2^k/(2^{k+1} - 1)$.) We define $\mathcal{H}^2(k)$ as follows.

$$V(\mathcal{H}^2(k)) = \{x_1, x_2, \dots, x_{3k}\} \cup \{a_1, a_2, \dots, a_k\} \cup \{e_1, e_2\}.$$

Let $A_{2i-1} = \{x_{3i-2}, a_i\}$, $A_{2i} = \{x_{3i-1}, a_i\}$, $B_{2i-1} = \{x_{3i-2}, x_{3i-1}, x_{3i}\}$ for $1 \leq i \leq k$, and $B_{2i} = \{x_{3i}, x_{3i+1}\}$ for $1 \leq i \leq k-1$, and $B_0 = \{x_1, x_{3k}, e_1\}$, $E_0 = \{x_{3k}, e_0\}$, $E_1 = \{e_0, e_1\}$ (see Fig. 1).

To find $\tau^*(\mathcal{H}^2(k))$ consider the following fractional matching $\lambda: E(\mathcal{H}^2(k)) \rightarrow \mathbb{R}$ and cover t .

Denote $2^{k+1} - 1$ by N . Let

$$\lambda(A_{2i-1}) = \lambda(B_{2i-1}) = 2^{k-i}/N \quad \text{for } 1 \leq i \leq k,$$

$$\lambda(A_{2i}) = (N - 2^{k-i})/N \quad \text{for } 1 \leq i \leq k,$$

$$\lambda(B_{2i}) = (N - 2^{k-i})/N \quad \text{for } 0 \leq i \leq k-1,$$

$$\lambda(E_1) = 2^k/N \quad \text{and} \quad \lambda(E_0) = (2^k - 1)/N.$$

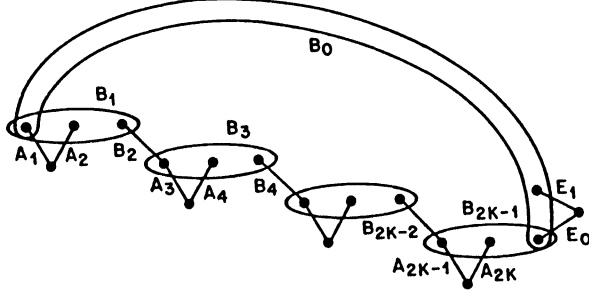


FIG. 1

Then $|\lambda| = 2k + 2^k/N$.

Define

$$\begin{aligned} t(a_i) &= (N - 2^{i-1})/N \quad \text{for } 1 \leq i \leq k, \\ t(x_{3i-2}) &= t(x_{3i-1}) = 2^{i-1}/N \quad \text{for } 1 \leq i \leq k, \\ t(x_{3i}) &= (N - 2^i)/N \quad \text{for } 1 \leq i \leq k, \\ t(e_1) &= (2^k - 1)/N, \quad t(e_0) = 2^k/N. \end{aligned}$$

Then t is a fractional cover with $|t| = 2k + 2^k/N$.

Example 3.3. (A hypergraph of rank 3 with $2k - 1$ edges and with $\{\tau^*\} = (\text{odd integer})/2^{k-1}$). Define $\mathcal{H}^3(k)$ as follows. $V(\mathcal{H}^3(k)) = \{x_1, \dots, x_k, y_1, \dots, y_k\}$,

$$E(\mathcal{H}^3(k)) = \{\{x_k, y_k\}, A_i, B_i (1 \leq i \leq k-1)\}$$

where $A_i = \{x_i, y_i, x_{i+1}\}$, $B_i = \{x_i, y_i, y_{i+1}\}$. Then

$$(3.1) \quad \tau^*(\mathcal{H}^3(k)) = \frac{2k}{3} + \frac{2}{9} + \frac{(-1)^{k-1}}{9 \cdot 2^{k-1}}.$$

To prove (3.1) consider the following (optimal) fractional matching w and cover t :

$$w(A_i) = w(B_i) = \frac{1}{3} + \frac{(-1)^{i-1}}{3 \cdot 2^i}, \quad (1 \leq i \leq k-1)$$

$$w(\{x_k, y_k\}) = \frac{2}{3} + \frac{(-1)^{k-1}}{3 \cdot 2^{k-1}},$$

and

$$t(x_i) = t(y_i) = \frac{1}{3} + \frac{(-1)^{k-i}}{3 \cdot 2^{k+1-i}}.$$

Proof of Theorem 3.1. For nonnegative integers $n_1, \dots, n_s$ and hypergraphs $\mathcal{H}_1, \dots, \mathcal{H}_s$, we denote the disjoint union of n_i copies of $\mathcal{H}_i$ by $\sum n_i \mathcal{H}_i$. Let $0 \leq r < 1$ be rational, $r = p/q$, $(p, q) = 1$. Let $q = 2^a b$ where b is odd. Choose nonnegative integers A and B such that

$$p/q = \left\{ \frac{A}{2^a} + \frac{B}{b} \right\}.$$

If a is even, define

$$\mathbf{H} := 9A\mathbf{H}^3(a+1) + 2B((2^{\phi(b)} - 1)/b)\mathbf{H}^2(\phi(b) - 1)$$

where ϕ denotes the Euler ϕ -function (i.e., $\phi(m)$ is the number of integers t , $1 \leq t < m$, which are relatively prime to m).

If a is odd, define

$$H := 18AH^3(a+2) + 2B((2^{\phi(b)} - 1)/b)H^2(\phi(b) - 1).$$

Then an easy calculation shows that $\{\tau^*(\mathcal{H})\} = r$, as required. $\square$

4. An upper bound on the denominator. Let $N_r = \{\tau^*(\mathcal{H}) : \mathcal{H} \text{ has rank at most } r\}$. Then $N_2 = \{1, \frac{3}{2}, 2, \frac{5}{2}, 3, \dots\}$ and $\bigcup_{r \geq 2} N_r$ consists of all rationals not smaller than 1.

THEOREM 4.1. *If $u/v \in N_r$, $(u, v) = 1$, then $u/v \geq (2 \log v)/(r \log r)$. In particular, the set N_r is a discrete sequence.*

The proof of this result is based on the following ideas. A hypergraph $\mathcal{H}$ is called τ^* -critical if $\tau^*(\mathcal{H}') < \tau^*(\mathcal{H})$ holds for each subhypergraph $\mathcal{H}'$ of $\mathcal{H}$, i.e., we cannot delete an edge without changing (decreasing) the value of τ^* .

LEMMA 4.2 (Füredi [Fü81]). *If $\mathcal{H}$ is τ^* -critical, then $|E(\mathcal{H})| \leq |\bigcup \{E \in E(\mathcal{H})\}|$, i.e., $\mathcal{H}$ has no more edges than nonisolated vertices.*

Other (more general) versions of this lemma are well known in the theory of linear programming. This lemma just means that the number of constraints of a linear program can be reduced to the number of variables without changing the optimal value.

LEMMA 4.3. *If $\mathcal{H}$ is τ^* -critical of rank r then $|E(\mathcal{H})| \leq r\tau^*$.*

Proof of 4.3. There exists an optimal fractional matching w_0 (i.e., $|w_0| = \tau^*$) which is a vertex of the fractional matching polytope $\text{FMP}(\mathcal{H})$. Thus, the vector $\{w_0(E)\}_{E \in E(\mathcal{H})}$ is contained in at least $|E(\mathcal{H})|$ facets of $\text{FMP}(\mathcal{H})$, say

$$\begin{aligned} w(E) &= 0 & \text{if } E \in \mathcal{E}_0 \subset E(\mathcal{H}), \\ \sum_{x \in E} w(E) &= 1 & \text{if } x \in V_0 \subset V(\mathcal{H}), \end{aligned}$$

where $|V_0| + |\mathcal{E}_0| \geq |E(\mathcal{H})|$. We have

$$|V_0| = \sum_{x \in V_0} \left(\sum_{x \in E} w(E) \right) = \sum_{E \in E(\mathcal{H})} w(E) |E \cap V_0| \leq \sum w(E) r = \tau^* r.$$

Hence $|E(\mathcal{H}) - \mathcal{E}_0| \leq r\tau^*$. Let $E(\mathcal{H}') = E(\mathcal{H}) - \mathcal{E}_0$. Then $\tau^*(\mathcal{H}') \geq |w_0|$, since w_0 is a fractional matching of $\mathcal{H}'$. However, $\tau^*(\mathcal{H}') \leq \tau^*(\mathcal{H})$, i.e., $\tau^*(\mathcal{H}') = \tau^*(\mathcal{H})$. $\mathcal{H}$ is τ^* -critical, so we have $\mathcal{H} = \mathcal{H}'$, $|E(\mathcal{H})| \leq r\tau^*$. $\square$

Applications of these lemmas can be found in [Fü86], [FF]. We now move to the proof of Theorem 4.1.

Proof of Theorem 4.1. Let $\mathcal{H}$ be a hypergraph of rank r with $\tau^*(\mathcal{H}) = u/v$. We can assume that $\mathcal{H}$ is τ^* -critical. Hence, $|E(\mathcal{H})| \leq (ru)/v$ by Lemma 4.3. The value of τ^* can then be obtained (by Cramer's rule) as a ratio of two 0-1 determinants of size at most $(ru)/v$. Every row contains at most r 1's by Hadamard's upper bound,

$$v = |\text{denominator}| = |\det(0\text{-}1 \text{ matrix})| \leq r^{(1/2)ru/v}.$$

PROPOSITION 4.4. *The smallest seven elements of N_3 are $1, \frac{4}{3}, \frac{3}{2}, \frac{5}{3}, \frac{7}{4}, \frac{9}{5}, 2$.*

Proof. Figure 2 shows the incidence matrices of seven hypergraphs of rank 3 with these fractional matching numbers (optimal fractional matchings and coverings are also indicated). (By Lemmas 4.2 and 4.3, it is sufficient to consider hypergraphs having at most five edges and vertices.) A short case-by-case study shows that these are all values of N_3 which are not larger than 2. $\square$

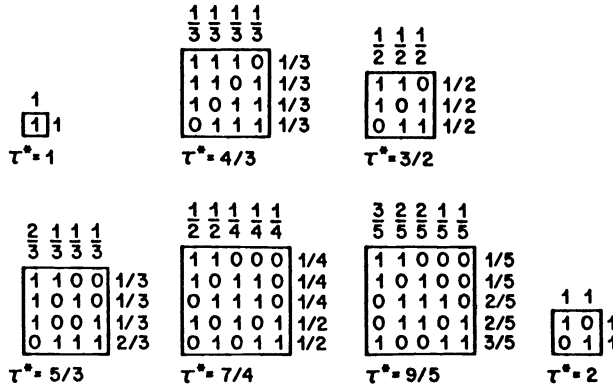


FIG. 2

5. Remarks.

Remark 5.1. One can prove in a way similar to that used for Lemma 4.3 that every vertex $\mathbf{c} \in \mathbb{R}^{|E(\mathcal{H})|}$ of FMP $(\mathcal{H})$ has at most $r\tau^*$ nonzero coordinates. This implies that there exists an integer $M = M(\mathbf{c}) \leq r^{(1/2)r\tau^*}$ such that $M\mathbf{c}$ is an integer point.

Remark 5.2. Let $N_r = \{t_1^{(r)}, t_2^{(r)}, \dots, t_i^{(r)}, \dots\}$, $t_i^{(r)} < t_{i+1}^{(r)}$. Although this is a discrete sequence, Theorem 3.1 implies that for $r \geq 3$,

$$\lim_{i \rightarrow \infty} (t_{i+1}^{(r)} - t_i^{(r)}) = 0$$

(since $\tau^*(\mathcal{H}') = 1 + \tau^*(\mathcal{H})$ if $\mathcal{H}'$ is formed from $\mathcal{H}$ by adding a single disjoint edge).

Remark 5.3. Define $d_r(n) = \max \{\text{denominator of } \tau^*(\mathcal{H}) : \mathcal{H} \text{ is an } r\text{-graph with } |E(\mathcal{H})| \leq n\}$. The examples in § 3 and Theorem 4.1 imply that

$$n \log \sqrt{2} + O(1) \leq \log d_3(n) \leq n \log \sqrt{3}.$$

It seems likely that $\lim_{n \rightarrow \infty} \log d_3(n)/n$ exists. If so, is it equal to $\frac{1}{2} \log 2$?

REFERENCES

- [B] M. L. BALINSKI, *Establishing the matching polytope*, J. Combin. Theory, 13 (1972), pp. 1–13.
- [BS] M. L. BALINSKI AND K. SPIELBERG, *Methods of integer programming: algebraic, combinatorial and enumerative*, in Progress in Operations Research, Vol. III, Relationship between Operations Research and the Computer, Wiley, New York, 1969, pp. 195–292.
- [Be] C. BERGE, *Packing problems and hypergraph theory: a survey*, Ann. Discrete Math., 4 (1979), pp. 3–37.
- [E] J. EDMONDS, *Maximum matching and a polyhedron with 0,1-vertices*, J. Res. Nat. Bur. Standards Sec. B, 69B (1965), pp. 125–130.
- [FF] P. FRANKL AND Z. FÜREDI, *Finite projective spaces and intersecting hypergraphs*, Combinatorica, 6 (1986), pp. 373–392.
- [Fü81] Z. FÜREDI, *Maximum degree and fractional matchings in uniform hypergraphs*, Combinatorica, 1 (1981), pp. 155–162.
- [Fü86] ———, *t-expansive and t-wise intersecting hypergraphs*, Graphs Combin., 2 (1986), pp. 67–80.
- [L75] L. LOVÁSZ, *On minimax theorems of combinatorics* (Doctoral thesis, in Hungarian), Mat. Lapok, 26 (1975), pp. 209–264.
- [L79] ———, *Graph theory and integer programming*, Discrete Optimization, (Proc. Adv. Res. Inst. Discrete Optimization and Systems Appl., Banff, Alberta, 1977), Ann. Discrete Math., 4 (1979), pp. 141–158.
- [NT] G. L. NEMHAUSER AND L. E. TROTTER, JR., *Properties of vertex packing and independence system polyhedra*, Math. Programming, 6 (1974), pp. 48–61.
- [T] W. T. TUTTE, *The 1-factors of oriented graphs*, Proc. Amer. Math. Soc., 4 (1953), pp. 922–931.

EXTREMUM PROPERTIES OF HEXAGONAL PARTITIONING AND THE UNIFORM DISTRIBUTION IN EUCLIDEAN LOCATION*

M. HAIMOVICH† AND T. L. MAGNANTI‡

Abstract. A zero-sum game with a maximizer that selects a point x in given polygon R in the plane and a minimizer that selects K points $c_1, c_2, \dots, c_K$ in the plane is considered; the payoff is the Euclidean distance from x to the closest of the points c_j , or any monotonically nondecreasing function of this quantity. Lower and upper bounds on the value of the game are derived by considering, respectively, the maximizer's strategy of selecting a uniformly distributed random point in R and the minimizer's strategy of selecting K members of a (uniformly) randomly positioned grid of centers that induce a covering of R by K congruent regular hexagons. The analysis shows that these strategies are asymptotically optimal (for $K \rightarrow \infty$). For Euclidean location problems with uniformly distributed customers, the results imply that hexagonal partitioning of the region is asymptotically optimal and that the uniform distribution is asymptotically the worst possible.

Key words. facility location, K -median Problem, location games, planar partitioning

AMS(MOS) subject classifications. 51M20, 52A40, 90C27

1. Introduction. Let $x_1, x_2, \dots, x_N$ be a given set of points in the plane $\mathbf{R}^2$ and consider the problem of finding K other points ("centers," "medians") $c_1, c_2, \dots, c_K$ so as to minimize

$$(1.1) \quad \sum_{i=1}^N \min_{1 \leq j \leq K} \|x_i - c_j\|,$$

expressed in terms of the Euclidean distance $\|x_i - c_j\|$ between x_i and c_j . This problem, which is known as the Multi-Source Weber Problem [KS] or the Euclidean K -median Problem¹ [HM], generalizes naturally to the problem of finding a set $C = \{c_1, c_2, \dots, c_K\}$ so as to minimize

$$(1.2) \quad D(w, C) \equiv \int \min_{c \in C} \|x - c\| dw(x).$$

In this expression, w is a finite positive Borel measure with bounded support on $\mathbf{R}^2$. We will refer to w as the *demand*. Let $S(w)$ denote the *support* of w and let $|w| \equiv w(\mathbf{R}^2)$ be the *total demand*. (The choice $w = \sum_{i=1}^N \delta_{x_i}$, with $\delta_{x_i}(T)$ for any x_i and T defined to be 1 if $x_i \in T$ and 0 otherwise, will return us to (1.1).)

The generalized problem is of interest, in part, because continuous distributions often provide computationally useful approximations of large ensembles of points (see [Ha3] for further details). When normalized so that $w(\mathbf{R}^2) = \int dw = 1$, w may be viewed as a probability measure on $\mathbf{R}^2$ and $D(w, C)$ as the associated expected distance between a random point and the point of C closest to it. Of particular interest is the case in which w is uniform on a region R of the plane, i.e., $w = \bar{m}\mu_R$ with $\bar{m}$ a positive constant and μ_R the restriction of μ , Lebesgue measure, to R .

* Received by the editors December 2, 1985; accepted for publication (in revised form) September 9, 1987. This research was supported by grants ECS-7926625 and ECS-8316224 from the Systems Theory and Operations Research Division of the National Science Foundation.

† Graduate School of Business, University of Chicago, Chicago, Illinois 60637. Present address, Department of Statistics, University of Tel Aviv, Tel Aviv, Israel.

‡ Sloan School of Management, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

¹ Note that the variant of the K -median Problem considered in this development does not require c_j 's to be chosen from among the x_j 's.

Now let $D_K(w)$ denote the minimal value of $D(w, C)$ when C is restricted to contain no more than K points. Theorem 1 of [Ha2] implies that some positive constant γ_2 satisfies

$$(1.3) \quad \lim_{K \rightarrow \infty} K^{1/2} D_K(w) = \gamma_2 \left(\int m(w)^{2/3} d\mu \right)^{3/2}$$

for all w having $m(w)$ as the density of its absolutely continuous part. Moreover, the author shows that for any measurable set T , as $K \rightarrow \infty$, the number of points in $C \cap T$ is proportional to $\int_T m^{2/3} d\mu$. This result is rather general and is not tied to many of the particular characteristics of the two-dimensional Euclidean K -median Problem. Using these characteristics, in this paper we are able to consider finer details of the (asymptotic) solution structure. We show in § 2 that if w is uniform on some bounded region R , then, as $K \rightarrow \infty$, the minimizing $c_1, c_2, \dots, c_K$ tend to be configured like the centers of the hexagonal cells in a honeycomb covering R . As a consequence, we find that

$$(1.4) \quad \gamma_2 = \sqrt{2/(3\sqrt{3})} \left(\frac{1}{3} + \frac{1}{4} \ln 3 \right) \approx 0.377$$

which is the average distance between the center and a (uniformly distributed) random point in a regular hexagon with unit area.

Consider now another, seemingly remote, problem. What is the largest possible value of $D_K(w)$, given that the support $S(w)$ of w is contained in R and that $|w| = w(R) = 1$; moreover, what demand w yields this maximum value? Such maximin problems, which are the subject of [HaM], seem quite difficult for $K > 2$. By randomizing the positioning of a hexagonal center grid, in § 3 we show, however, that there is a positioning of such a grid for which the cost $D(w, C)$ (and therefore $D_K(w)$) is smaller than the D_K obtained for a demand w that is uniform on a hexagonal cover of R that is only slightly larger than R (when the perimeter is finite, the area of the difference is $O(K^{-1/2})$). This result helps to establish that

$$(1.5)^2 \quad \lim_{K \rightarrow \infty} K^{1/2} \sup_w \{D_K(w) : |w| = 1, S(w) \subset R\} = \gamma_2 \mu(R)^{1/2}$$

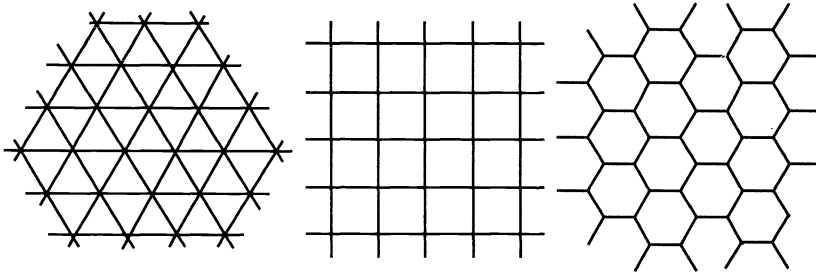
(where γ_2 is given by (1.4)), as well as the fact that as K grows, the uniform distribution becomes the worst possible (i.e., cost maximizing) demand.

The results contained in § 2 originated in [HA1]. They are similar to those established independently (and simultaneously) by Papadimitriou [Pa] and, as it turns out, quite earlier by Fejes-Toth [F1], [F2] and in other versions by Ballobás and Stern [BS] and Imre [I].³ Our results in this section differ from Papadimitriou's in two respects. First, he relies on a computer-aided proof to establish an essential convexity property (see Lemma 2 of this paper) underlying the analysis. On the other hand, our development is completely analytic. Furthermore, our proof provides more geometric insight and actually establishes a far more general result: any monotone function f of Euclidean distances can be used in the objective function (1.2) in place of the Euclidean distances themselves. Although not explicitly stated as such, Fejes-Toth's proof, which differs from ours, also implies this generalization.

Finally, it is pleasing to note that the maximin result (1.5) to be considered in § 3 also generalizes in this same fashion.

² Note that (1.5) is stronger than $\sup_w \{ \lim_{K \rightarrow \infty} K^{1/2} D_K(w) : |w| = 1, S(w) \subset R \} = \gamma_2 \mu(R)^{1/2}$ which (using Jensen's inequality) is an immediate consequence of (1.3) (see § 3).

³ We are grateful to J. M. Steele and one of the referees for bringing these earlier works to our attention.

FIG. 1. *The regular partitioning schemes.*

2. Hexagonal partitioning and lower bounds for uniform demand.

Preview. As motivation for the honeycomb⁴ form of the asymptotic solution, let us consider the following version of the problem. Suppose that for any particular K , we were free to move the demand points and choose any shape and size for the single facility subregions, provided only that the total area were equal to A . Then, as is easy to verify, each subregion would be circular with area A/K . Unfortunately, such a partition of R is not possible in general (it is never possible for more than one value of K). Intuitively, a close approximation to the K -circle partition seems to consist of K disjoint congruent circles of maximal size packed inside the region R and with the remainder of R divided between the circles so that each point is assigned to the center closest to it. Asymptotically this solution is equivalent to partitioning the region into K congruent regular hexagons.

Suppose, instead, that we consider partitioning R into congruent regular polygons. Figure 1 illustrates partitions into equilateral triangles, into squares, and into hexagons. Partitioning into congruent regular polygons with more than six edges is asymptotically impossible (see Lemma 4 to follow). Consequently, from among all regular partitions, the natural choice for approximating the nonachievable, ideal, circular, single-facility service regions would appear to be the hexagonal partitioning scheme.

Of course, it is not clear that a nonregular polygonal partitioning having some cells with more than six edges and some with fewer than six edges might not be preferable. As frequently is the case on such occasions, showing that the cost in an “average” cell (that is, one with an average number of edges and an average area) is lower than the average cost per cell involves application of a convexity argument. Indeed, our formal proof relies on such an argument.

Problem setting. As we will see, the hexagonal partitioning property applies to problems with cost structure broader than that of the K -median Problem, namely, whenever the cost is monotonically increasing (not necessarily linearly, as in the K -median Problem) in the Euclidean customer to facility distance.

Let $f: [0, \infty) \rightarrow [0, \infty]$ be monotonically increasing and consider the following generalizations of $D(w, C)$ and $D_K(w)$:

$$D^f(w, C) = \int f\left(\min_{c \in C} \|x - c\|\right) dw$$

⁴ Regular hexagonal partitioning structures are common in nature (see *On Growth and Form* by D’Arcy Thompson [Th]). They arise for a variety of reasons, sometimes as a solution to extremum problems. For example, they minimize (asymptotically) the total length of the walls of a partition of a planar region into K equal area subregions, thus providing an economic (wax saving) rationale to the hexagonal structure of honeycombs.

and

$$D_K^f(w) = \min \{D^f(w, C) : |C| \leq K\}.$$

D^f and D_K^f coincide with D and D_K whenever f is the identity, i.e., $f(r) = r$.

We shall require some additional notational conventions:

- Let $T(\alpha, A)$ denote the right angle triangle with area A , with a vertex of acute angle α at the origin, and with one edge on the horizontal axis.
- Let $\sigma_f(\alpha, A) = D^f(\mu_{T(\alpha, A)}, \{0\}) = \int_{T(\alpha, A)} f(\|x\|) d\mu$.
- For $y > 2$ and $A \geq 0$, let $\phi_f(y, A) = 2y \cdot \sigma_f(\pi/y, A/2y)$. Note that for $y = 3, 4, 5, \dots$, $\phi_f(y, A) = \int_{P_y(A)} f(\|x\|) d\mu$ is the cost of serving a regular y -gon $P_y(A)$ of area A by a facility located at its center.
- Let $\phi(y) = \phi_I(y, I)$, where I is the identity (i.e., $I(r) = r$). That is,

$$\phi(y) = 2y \int_{T(\pi/y, 1/2y)} \|x\| d\mu.$$

Calculation yields

$$\phi(y) = \frac{1}{3} \left(y \tan^3 \frac{\pi}{y} \right)^{-1/2} \left(\sec \frac{\pi}{y} \tan \frac{\pi}{y} + \ln \left(\sec \frac{\pi}{y} + \tan \frac{\pi}{y} \right) \right).$$

Note that for $y = 3, 4, 5, \dots$, $\phi(y)$ is the average distance to the center in a regular unit area y -gon.

Results. We are ready to state the main result of this section.

THEOREM 1 (Fejes-Toth). *Let $f: [0, \infty) \rightarrow [0, \infty)$ be monotonically increasing and let R be an n -gon of area A . Then for all $K \geq 1$,*

$$D_K^f(\mu_R) \geq K \phi_f \left(6 + \frac{n-6}{K}, \frac{A}{K} \right).$$

In particular (when f is the identity),

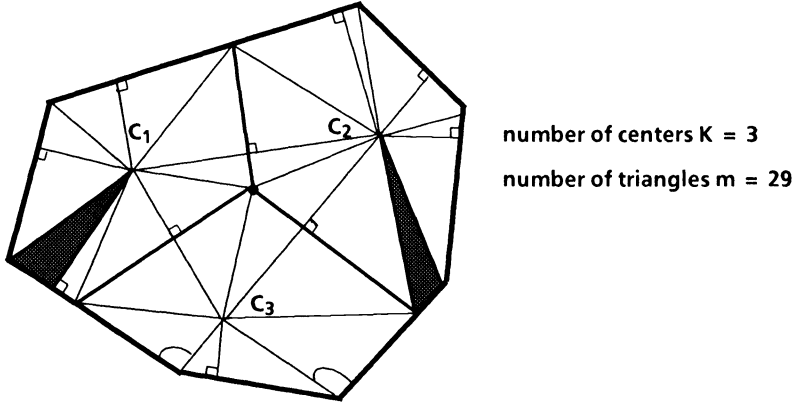
$$D_K(\mu_R) \geq \phi \left(6 + \frac{n-6}{K} \right) A \left(\frac{A}{K} \right)^{1/2}.$$

Remarks. (i) If $n \leq 6$ (e.g., R is rectangular), we may substitute $\phi_f(6, A/K)$ and $\phi(6)$ for $\phi_f(6 + (n-6)/K, A/K)$ and $\phi(6 + (n-6)/K)$ in the theorem.

(ii) If $K = 1$, the right-hand side of these expressions become $\phi_f(n, A)$ and $\phi(n) A^{3/2}$. Thus, *for the location of a single center, the regular n -gon gives the smallest cost from among all n -gons.*

(iii) While the proof of Theorem 1 (specifically Lemma 4) considers only convex regions R , the theorem remains valid not only for polygons, as stated, but even for bounded disconnected sets with a piecewise linear boundary: in this setting, if l is the number of components of R and h is the number of “holes” in R , the term $6 + (n - 6(l - h))/K$ replaces $6 + (n - 6)/K$; this extension follows (with some added difficulties that arise because service regions might not be simply connected) from a simple modification of Lemma 4 (see footnote 6).

The asymptotic optimality of hexagonal partitions is an obvious corollary of Theorem 1. Observing that ϕ_f is continuous in both arguments, that $6 + (n - 6)/K \rightarrow 6$ as $K \rightarrow \infty$, and that R can be covered by K congruent regular hexagons of an area that converges to A as $K \rightarrow \infty$, we have the following.

FIG. 2. Partition of R into triangles.

COROLLARY 1.1. *If R is a polygon, and f is monotone, then,*

$$\lim_{K \rightarrow \infty} \frac{D_K^f(\mu_R)}{K \phi_f(6, \mu(R)/K)} = 1.$$

In particular, $\lim_{K \rightarrow \infty} (D_K(\mu_R)/K^{-1/2} \phi(6) \mu(R)^{3/2}) = 1$ and thus, recalling Theorem 1 of [Ha2] (see also (1.3)), we conclude that $\gamma_2 = \phi(6)$, i.e., we have the following.

COROLLARY 1.2. *For any demand distribution w in the plane,*

$$\lim_{K \rightarrow \infty} K^{1/2} D_K(w) = \phi(6) \left(\int m^{2/3} d\mu \right)^{3/2}$$

where $\phi(6) = \sqrt{2/(3\sqrt{3})}(\frac{1}{3} + \frac{1}{4} \ln 3)$ is the average distance of points from the center of a regular unit area hexagon.

The rest of this section is devoted to the proof of Theorem 1. The proof is based on the convexity of σ_f , the (decreasing) monotonicity of ϕ_f in its first argument, y , and the fact that the average number of edges in a single facility service cell is no more than $6 + (n - 6)/K$. We establish each of these facts as intermediate results. The proof can be viewed as constructive in the sense that it is based on a sequence of mappings of R (and C) that reshape R while (i) preserving its area and (ii) not increasing the distance of any point in R to the closest center in C .

Proof of Theorem 1. Though the assertion holds as stated, we restrict ourselves to situations in which R is convex. (The extension to the nonconvex case is tedious and not very illuminating.) We may assume that $C \subset R$, since the projection on R of any point $c \in C - R$ lies closer than c to any point in R , and thus reduces $D^f(\mu_R, C)$.

Let $R_j = \{x \in R: \|x - c_j\| \leq \|x - c_i\| \text{ for all } i = 1, 2, \dots, K\}$. $R_1, R_2, \dots, R_K$ constitutes a partition⁵ of R into K polygons. Draw lines from each center c_j to the vertices of its associated polygon R_j and draw perpendicular lines from each c_j onto the edges of R_j (provided they lie inside R_j). The result is a “center” partition of R into triangles $T_1, T_2, \dots, T_m$. Each triangle T_i has one of the centers as a vertex and has another vertex incident to a right or obtuse angle (see Fig. 2).

For $i = 1, 2, \dots, m$, let A_i denote the area of triangle T_i , and let α_i denote the acute angle of T_i that is adjacent to a center.

⁵ Strictly speaking, these sets do not give a proper partition, since the common boundaries (that have a null area) of the polygons are counted more than once.

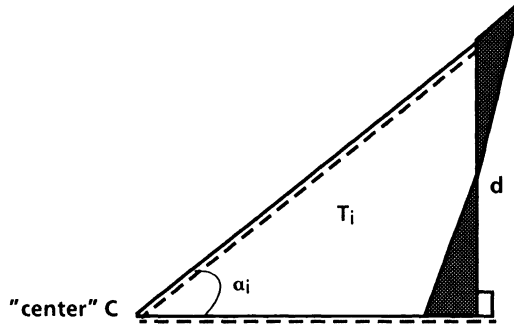


FIG. 3. Comparison between obtuse and right angle triangles.

Consider the following intermediate results.

LEMMA 1. $D^f(\mu_{T_i}, C) \geq \sigma_f(\alpha_i, A_i)$ for all $i = 1, \dots, m$.

Proof. If T_i is a right angle triangle, then by definition and by the invariance of Euclidean distances and of areas under displacement $D^f(\mu_{T_i}, C) = \sigma_f(\alpha_i, A_i)$. If, on the other hand, T_i has an obtuse angle then, as depicted in Fig. 3, we compare it to a right angle triangle with the same angle at the “center” vertex.

Since any point in the lower shaded area in Fig. 3 except d is closer to the “center” vertex than any point in the upper shaded area, we conclude that

$$D^f(\mu_{T_i}, C) > \sigma_f(\alpha_i, A_i).$$

□

LEMMA 2. $\sigma_f(\beta, B)$ is convex on the region $(0, \pi/2) \times [0, \infty)$.

Proof. Since σ_f is continuous, it suffices to show that $\sigma_f(\beta_1, B_1) + \sigma_f(\beta_2, B_2) \geq 2\sigma_f((\beta_1 + \beta_2)/2, (B_1 + B_2)/2)$ whenever $0 < \beta_1, \beta_2 < \pi/2$ and $B_1, B_2 \geq 0$. We compare a pair of right angle triangles with areas B_1, B_2 and angles β_1, β_2 at a “center” vertex, with a pair of congruent right angle triangles each with area $\frac{1}{2}(B_1 + B_2)$ and an angle $\frac{1}{2}(\beta_1 + \beta_2)$ at the “center” vertex (see Fig. 4).

The comparison, depicted in Fig. 4, is carried out in two stages. We first compare the original pair of right angle triangles (leftmost in Fig. 4) to an intermediate pair of right angle triangles (middle of Fig. 4) of the same total area and total angle at the

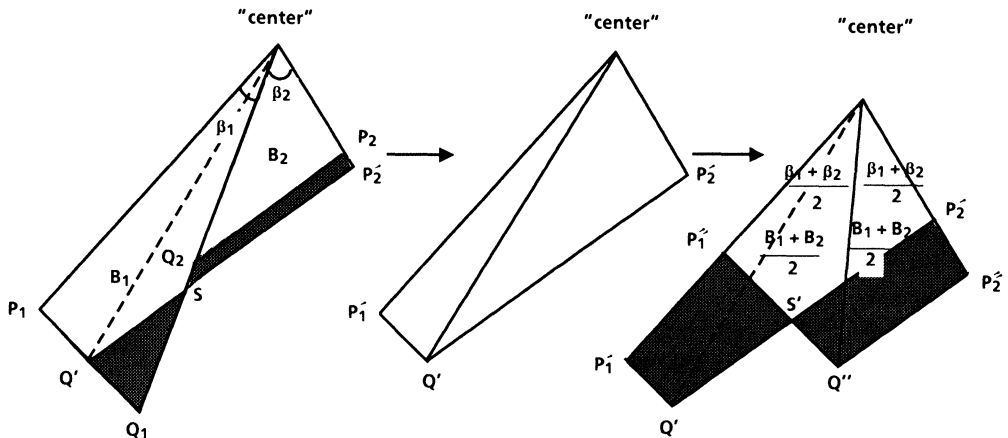


FIG. 4. Comparison between pairs of right angle triangles.

“center” vertex, and with a common hypotenuse. Then we compare the intermediate pair to the pair of congruent right angle triangles (rightmost in Fig. 4) with the same total area and total angle at the “center” vertex. It is enough to show that, when we move from left to right in Fig. 4, the area that is at distance r or more from the center is decreasing.

That this assertion is true for the first comparison follows simply from the fact that every point in the trapezoid $Q_2P_2P'_2S$ other than S is closer to the center than any point in the (equivalent in area) triangle Q_1Q_2S . (This argument is similar to that employed in the proof of Lemma 1.)

The situation in the second comparison is not so straightforward. It is not the case that every point in the trapezoid $S'P'_2P''_2Q''$ lies closer to the center than any point in the trapezoid $S'P'_1P'_1Q'$. We can, however, carry out a point-by-point comparison or, precisely speaking, demonstrate an area (i.e., Lebesgue measure) preserving and distance (to center) decreasing mapping of $S'P'_1P'_1Q'$ into $S'P'_2P''_2Q''$. One such mapping can be constructed as follows (see Fig. 5): slice trapezoid $S'P'_1P'_1Q'$ into very thin (almost rectangular) strips, using lines parallel to the basis $S'P'_1$. Then select the first strip (that is adjacent to $S'P'_1$) and “stretch” it so that its length expands to the length of $S'P'_2$, while its width shrinks so as to preserve the area. Place the stretched strip in the trapezoid $S'P'_2P''_2Q''$ so that the point that was adjacent to P'_1 is now adjacent to P'_2 . To complete the mapping, we proceed in the same manner, selecting the next strip, “stretching” it and placing it along the previously transformed strip and so on until the whole trapezoid is transformed. More precisely, if X is a point in $S'P'_1P'_1Q'$, then there is $0 \leq \lambda(X) \leq 1$ so that $X = \lambda\bar{X} + (1 - \lambda)\underline{X}$ where $\underline{X}$ lies on $\overline{P'_1P'_1}$, $\bar{X}$ lies on $\overline{S'Q'}$ and the segment $\underline{X}\bar{X}$ (passing through X) is parallel to $\overline{P'_1S'}$. Let $A(X)$ be the area of the subtrapezoid $S'P'_1\underline{X}\bar{X}$. Next construct a subtrapezoid $S'P'_2\underline{Y}\bar{Y}$ of $S'P'_2P''_2Q''$ (see Fig. 5) that has the same area $A(X)$ and let $Y = \lambda\bar{Y} + (1 - \lambda)\underline{Y}$. Consider the 1-1 mapping of $S'P'_1P'_1Q'$ into $S'P'_2P''_2Q''$ defined by the correspondence $X \rightarrow Y$. The mapping is obviously area (measure) preserving, and it is a straightforward exercise (see Lemma 5 in the Appendix) to verify that it is distance decreasing.

We conclude, then, that for all monotonically increasing f ,

$$2\sigma_f((\beta_1 + \beta_2)/2, (B_1 + B_2)/2) \geq \sigma_f(\beta_1, B_1) + \sigma_f(\beta_2, B_2)$$

or, that σ_f is convex in $(0, \pi/2) \times [0, \infty)$. $\square$

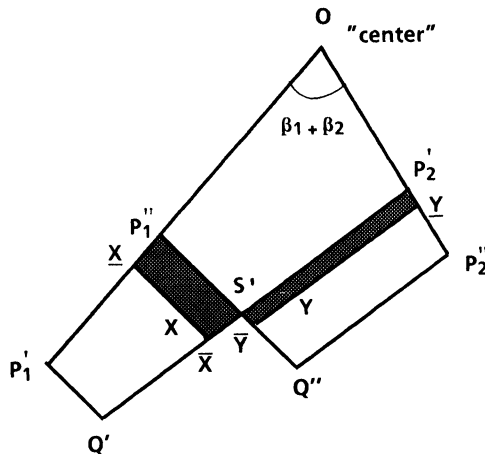


FIG. 5. Area preserving, distance decreasing mapping.

LEMMA 3. For fixed A and f , $\phi_f(y, A)$ is monotonically decreasing in y over the interval $(2, \infty)$.

Proof. By Lemma 2, we know that σ_f is convex, and thus, for any positive scalar λ , $\sigma_f(\alpha, \lambda\alpha)$ is convex in α . Furthermore, $\sigma_f(0, \lambda 0) = \sigma_f(0, 0) = 0$. Consequently (by the 3-chord lemma, for example), $\sigma_f(\alpha, \lambda\alpha)/\alpha$ is monotonically increasing in α for $0 < \alpha < \pi/2$. Recalling that $\phi_f(y, A) = 2\pi\sigma_f(\pi/y, A/2\pi \cdot \pi/y)/(\pi/y)$ completes the proof. $\square$

LEMMA 4. The number of triangles m in the “center” partition of R does not exceed $2K(6 + (n - 6)/K)$.

Proof. By our construction, $m \leq 2 \sum_{j=1}^K n_j$ where n_j is the number of edges of the polygon R_j .

Consider the vertices of the polygons $R_1, R_2, \dots, R_K$; there are three types of vertices:

- (1) those that are vertices of R ,
- (2) those that lie on the relative interior of edges of R , and
- (3) those that lie in the interior of R .

There are (by definition) n type 1 vertices. Let N_i for $i = 2$ and 3 denote the number of type i vertices.

Each type 2 vertex is a vertex of at least two polygons and each type 3 vertex is adjacent to at least three polygons. Furthermore, it is easy to see that each such vertex must be a vertex of all the polygons adjacent to it. (That is, no two edges adjacent to a type 3 vertex are collinear.)

Consequently, as depicted in Fig. 6, the number of subregional edges adjacent to each type 1 (respectively, type 2 and 3) vertex is at least two (respectively, four and six). (Note that any edge of two adjacent subregions is counted twice.) Since the total number of edges in R_j is n_j , summing over all the vertices gives the subregional edge count

$$(2.1) \quad \sum_{j=1}^K n_j \geq \frac{1}{2}(2n + 4N_2 + 6N_3).$$

The division by two of the right-hand side reflects the fact that each edge is counted twice in the vertex adjacent count, once for each of its two vertices.

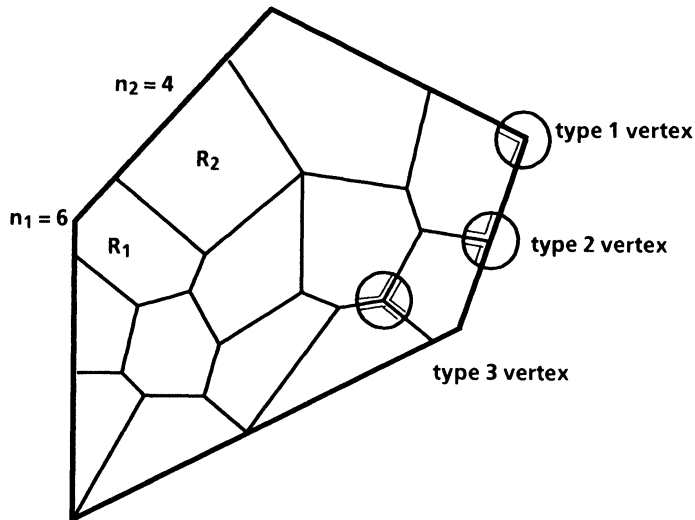


FIG. 6. Partition of R into polygonal service regions.

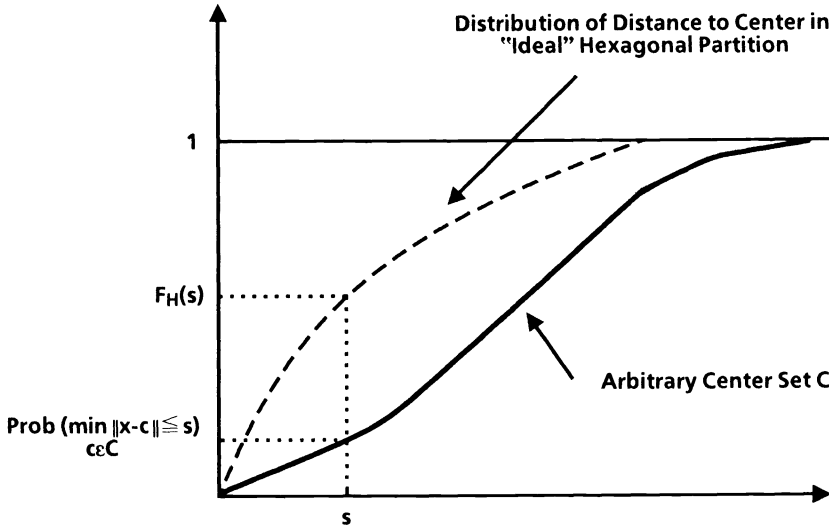


FIG. 7. Stochastic dominance of hexagonal partitions.

Summing the angles over the vertices and over the polygons yields

$$(2.2)^6 \quad \sum_{j=1}^K \pi(n_j - 2) = \pi(n - 2) + \pi N_2 + 2\pi N_3$$

which implies that $N_3 = \frac{1}{2}(\sum_{j=1}^K n_j - N_2 - n - 2(K - 1))$. Substituting this expression in (2.1) and rearranging gives

$$\sum_{j=1}^K n_j \leq 6(K - 1) + n - N_2 \leq 6(K - 1) + n = K \left(6 + \frac{n - 6}{K} \right)$$

and, thus, $m \leq 2 \sum_{j=1}^K n_j \leq 2K(6 + (n - 6)/K)$. $\square$

We are now ready to complete the proof of Theorem 1. Using the fact that $\sum_{i=1}^m \alpha_i = 2\pi K^7$ and $\sum_{i=1}^m A_i = A$, and Lemmas 1-4, we have

$$\begin{aligned} D^f(\mu_R, C) &= \sum_{i=1}^m D^f(\mu_{T_i}, C) \geq \sum_{i=1}^m \sigma_f(\alpha_i, A_i) \geq m \sigma_f \left(\frac{\sum_{i=1}^m \alpha_i}{m}, \frac{\sum_{i=1}^m A_i}{m} \right) \\ &= m \sigma_f \left(\frac{2\pi K}{m}, \frac{A}{m} \right) = K \phi_f \left(\frac{m}{2K}, \frac{A}{K} \right) \geq K \phi_f \left(6 + \frac{n - 6}{K}, \frac{A}{K} \right). \end{aligned}$$

The three inequalities in this expression follow, respectively, from Lemma 1, Lemma 2, and Lemma 3 together with Lemma 4. Since this result is valid for any C , the theorem is valid. $\square$

To conclude this section, we note that the proof of Theorem 1 actually establishes (by a sequence of measure preserving, distance decreasing transformations) a stochastic

⁶ This equation, which could be obtained from Euler's Lemma for planar connected graphs, may also be extended to the more general case in which R is a bounded set with piecewise linear boundary having l components and h "holes," but each subregion R_i for $i = 1, 2, \dots, K$ is still simply connected; in this instance, $\pi(n - 2(l - h))$ replaces $\pi(n - 2)$ in the right-hand side of (2.2).

⁷ This equality is obvious when all centers are in the interior of R . However, if c_j lies on an edge or a vertex of R , we can add degenerate null area triangles to justify $\sum \alpha_i = 2\pi$ at that center.

dominance property of hexagonal partitioning. Let F_H denote the cumulative distribution of distances from a (uniformly) random point in a hexagon with area $\mu(R)/K$ to its center. This distance distribution is achievable for the distances of points to their nearest center if the region R can be partitioned into K congruent regular hexagons. (This “ideal” partition is in general achievable approximately, with negligible boundary effects, only for large values of K .) Next, suppose, for simplicity, that the region R is a square and that C is any specified center set of K points. Then, as shown in Fig. 7, F_H stochastically dominates the cumulative distribution of distances of points in R to their nearest center in C . This dominance relationship is, incidentally, the form of Theorem 1 as stated by Fejes-Toth [F2].

The proof of Theorem 1 implicitly uses this dominance property to show that hexagonal partitioning minimizes $\int \min_{1 \leq j \leq K} f(\|x - c_j\|) d\mu(x)$ for any monotone function f . In addition, the dominance property also implies that hexagonal partitioning is asymptotically cost minimizing when other cost functionals are used in place of the integral. For example, hexagonal partitioning asymptotically minimizes

$$\max_{x \in R} \min_{1 \leq j \leq K} \|x - c_j\|,$$

as shown by Zemel [Z].

3. Uniform demand and upper bounds for hexagonal partitioning. In this section we derive an upper bound on $D_K^f(w)$ (and $D_K(w)$) for demand distributions w whose support is contained in a bounded region R with a rectifiable boundary. The bound depends only on the area A and the perimeter p of R . In contrast to Theorem 1, though, the bound is not restricted to uniform demand.

This upper bound and the lower bound of Theorem 1 imply that the uniform distribution on a region R is in some “strong” asymptotic sense (that will be explained) the worst possible (i.e., cost maximizing) distribution in that region.

THEOREM 2. *Let R be a connected region in the plane with finite area A and with perimeter p . Then for any demand w with support in R and for any measurable real-valued function f defined on $[0, \infty)$,*

$$D_K^f(w) \leq |w| \phi_f\left(6, \frac{\bar{A}}{K}\right) \left(\frac{\bar{A}}{K}\right)^{-1}.$$

In particular, if f is the identity, then

$$D_K(w) \leq \phi(6) \cdot |w| \left(\frac{\bar{A}}{K}\right)^{1/2}.$$

In these expressions,

$$\bar{A} = A \left(\frac{1}{1 - aK^{-1}} + \frac{2bK^{-1/2}}{1 - aK^{-1}} \sqrt{\frac{1}{1 - aK^{-1}} + \left(\frac{b}{1 - aK^{-1}}\right)^2} \right)$$

where

$$a \equiv \frac{8\pi}{3\sqrt{3}} \quad \text{and} \quad b \equiv \sqrt{\frac{2}{3\sqrt{3}}} \frac{p^2}{A}.$$

Since ϕ_f is continuous in its second argument and since $\bar{A} \rightarrow A$ as $K \rightarrow \infty$ ($\bar{A} = A + o(K^{-1/2})$), we immediately have (in conjunction with Theorem 1) the following.

COROLLARY 2.1. *Assume the conditions of Theorem 2. Then,*

$$\lim_{K \rightarrow \infty} \left(\sup_w \left\{ \frac{D_K^f(w)}{|w|\phi_f(6, A/K)(A/K)^{-1}} : S(w) \subset R, w \neq 0 \right\} \right) = 1.$$

Proof. That the limit is at most one follows immediately from Theorem 2. Selecting w as uniform over R and invoking Theorem 1, we conclude that the limit is at least one. $\square$

Remark. Corollary 2.1, as well as the subsequent Corollary 2.2, holds for any Lebesgue measurable bounded R with $\mu(R) = A$, but we will not pursue this matter in this paper.

Recalling Theorem 1 (or Corollary 1.1), we have the following.

COROLLARY 2.2. *Assume the conditions of Theorem 2 and let $\bar{m} = |w|/A$ (i.e., $\bar{m}\mu(R) = |w|$). Then,*

$$\lim_{K \rightarrow \infty} \left(\sup_w \left\{ \frac{D_K^f(w)}{D_K^f(\bar{m}\mu_R)} : S(w) \subset R, w \neq 0 \right\} \right) = 1.$$

That is, among all demand distributions in a given region with some fixed total demand, the uniform distribution is asymptotically worst (i.e., cost maximizing).

It is interesting to compare this last property with a slightly weaker asymptotic maximality result. Consider the equality

$$\lim_{K \rightarrow \infty} K^{1/2} D_K(w) = \phi(6) \left(\int m^{2/3} d\mu \right)^{3/2}$$

of Corollary 1.2. Applying Jensen's inequality to the convex function $g(x) = x^{3/2}$ gives

$$\left(\frac{1}{\mu(R)} \int m^{2/3} d\mu \right)^{3/2} \leq \frac{1}{\mu(R)} \int (m^{2/3})^{3/2} d\mu,$$

that is,

$$\left(\int m^{2/3} d\mu \right)^{3/2} \leq \left(\int m d\mu \right) \mu(R)^{1/2} \leq |w| \mu(R)^{1/2}$$

with equalities if and only if m is constant almost everywhere in R , and is equal to $\bar{m} = |w|/\mu(R)$. In other words,

$$(*) \quad \sup_w \left\{ \lim_{K \rightarrow \infty} \frac{D_K(w)}{D_K(\bar{m}\mu_R)} : S(w) \subset R, w \neq 0 \right\} = 1.$$

This result is substantially weaker than Corollary 2.2, in the sense that here we maximize with a single demand for the whole tail of the sequence, where in Corollary 2.2 we are free to maximize with different demands for the individual K 's. (Recall that in general $\sup(\lim(\cdot)) \leq \lim(\sup(\cdot))$.) It is interesting to note that if we consider a variation of the K -median Problem that restricts the centers to lie within the support of the demand (i.e., $C \subset S(w)$), then $(*)$ will be still valid ([Ha2, § 4]), while Corollary 2.2 will not ([HaM, § 7]).

The proof of Theorem 2, which concludes this section, is based on a randomized hexagonal covering argument. This randomized covering can be interpreted as a randomized strategy in a zero-sum game with payoff $\min_{c \in C} f(\|x - c\|)$. The maximizer in this game chooses $x \in R$ and the minimizer chooses K points C from $\mathbf{R}^2$. With this interpretation, w is a randomized strategy for the maximizer (note that in our original problem there is no loss of generality in assuming that $|w| = 1$), while the randomized

hexagonal covering is a randomized strategy for the minimizer. Our conclusion is that asymptotically (as $K \rightarrow \infty$) the uniform distribution for x together with a uniformly randomized positioning of a regular hexagonal cover constitute a saddle point of this game. A companion paper [HaM] treats this result and other so-called “location games” more thoroughly.

Proof of Theorem 2. Partition the plane into congruent regular hexagons (reference hexagons) each with area $\bar{A}/K$. Take the grid of the centers of the hexagons and, without changing its orientation, position it (as a whole) randomly so that each single center is distributed uniformly in a regular hexagon of area $\bar{A}/K$.

By Lemma 6 of the Appendix, R is covered, for *any* positioning of this grid, by no more than K of the hexagons; that is, we may assume that for any positioning of the grid, every demand point in R is served by the grid point closest to it.

Consider now an arbitrary point $x \in R$ and any particular random grid G of centers generated as specified. Draw a regular hexagon of area $\bar{A}/K$ around x (Fig. 8). This hexagon always contains exactly one grid point from G , which is the center serving the demand point x from this grid of centers. By our choice of randomization, this center has a uniform distribution in this hexagon. Thus, if there is discrete demand mass $\Delta w(x)$ at point x , the average (expected) cost paid by the demand there is

$$\frac{\Delta w(x)}{\bar{A}/K} \phi_f\left(6, \frac{\bar{A}}{K}\right).$$

(Recall that $(1/(\bar{A}/K))\phi_f(6, \bar{A}/K) = (1/\mu(H)) \int_H f(\|x\|) d\mu$, where H is a regular hexagon of area $\bar{A}/K$, is the average cost per unit demand at our point.)

Summation (or integration in the nondiscrete case) over all randomized grid solutions yields an average total cost $|w|\phi_f(6, \bar{A}/K)(\bar{A}/K)^{-1}$. Since each of these randomized grid solutions is feasible (i.e., use K or fewer centers), the value of the minimum cost solution must be no more than $|w|\phi_f(6, \bar{A}/K)(\bar{A}/K)^{-1}$ as well. $\square$

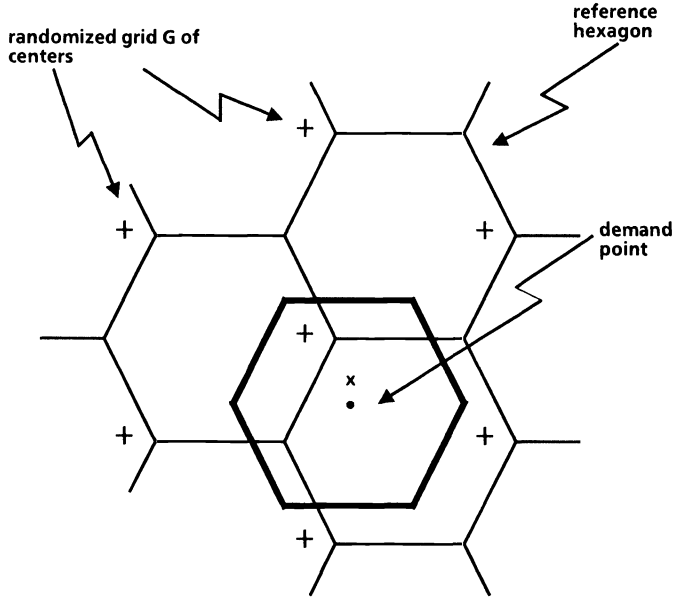


FIG. 8. A single customer amid a randomized center grid.

4. Summary. We have derived a lower bound on the optimal cost of K -median Problems (or similar monotone-in-distance K -median Problems) with demand uniformly distributed in a given bounded region R . On the other hand, using a randomly positioned hexagonal grid we have established an upper bound for the optimal cost that is valid for all possible demand distributions in R . These two bounds converge (in ratio), as K tends to infinity, to the cost for demand uniformly distributed over K regular hexagons of area $\mu(R)/K$ each with their centers chosen for $c_1, c_2, \dots, c_K$. Therefore, as $K \rightarrow \infty$, if the demand is distributed uniformly, a hexagonal partitioning scheme is asymptotically optimal.

As mentioned in [HaM], we have essentially found a saddlepoint of a zero-sum game in which the maximizer chooses a point in R , the minimizer chooses K points in R , and the payoff is the distance between the maximizer's choice to the closest of the minimizer's choices (or some nondecreasing function of this distance).

Appendix.

LEMMA 5. In Fig. 5, $\overline{OY}$ is shorter than $\overline{OX}$.

(The construction of Fig. 5 is described in Lemma 2 in § 2.)

Proof. (Let $\overline{MN}$ denote the length of segment MN and let $\angle MNL$ denote the angle MNL .) Since the trapezoids $S'P_1'\overline{X}\overline{X}$ and $S'P_2'\overline{Y}\overline{Y}$ are equal in area, while the bases of the first are shorter than those of the second, their altitudes are inversely ordered, i.e., $\overline{P_1'\overline{X}} \geq \overline{P_2'\overline{Y}}$. Noting now that $\angle P_1'S'\overline{X} = \angle P_2'S'\overline{Y}$ we get $S'\overline{X} \geq S'\overline{Y}$. This result together with the fact $\angle OS'\overline{X} \geq \angle OS'\overline{Y}$ implies $\overline{OX} \geq \overline{OY}$.

Also, $\psi_1 = \angle \overline{XOX} \leq (\beta_1 + \beta_2)/2 \leq \angle \overline{YOY} = \psi_2$. Consequently,

$$\begin{aligned} \overline{OY} &= \overline{OY} \sqrt{\cos^2 \psi_2 + \lambda^2 \sin^2 \psi_2} = \overline{OY} \sqrt{1 - (1 - \lambda^2) \sin^2 \psi_2} \leq \overline{OX} \sqrt{1 - (1 - \lambda^2) \sin^2 \psi_1} \\ &= \overline{OX}. \end{aligned} \quad \square$$

LEMMA 6. Let R be a connected region in the plane with finite area A and perimeter p , and let $K \geq 5$. Then any partition of the plane into congruent regular hexagons, each with area $\bar{A}/K$, where

$$(A.1) \quad \bar{A} = A \left(\frac{1}{1 - aK^{-1}} + \frac{2bK^{-1/2}}{1 - aK^{-1}} \sqrt{\frac{1}{1 - aK^{-1}} + \left(\frac{b}{1 - aK^{-1}} \right)^2} \right)$$

$$a = \frac{8\pi}{3\sqrt{3}} \quad \text{and} \quad b = \sqrt{\frac{2}{3\sqrt{3}}} \frac{p^2}{A}$$

will use no more than K hexagons to cover R .

(Remark. If R is not connected, but has l components and h "holes," then the result still holds with $a = (8\pi/3\sqrt{3})(l - h)$.)

Proof. Consider any partition of the plane into congruent regular hexagons of diameter d (i.e., edges each have length $d/2$). Then, the set

$$\bar{R}(d) = \{x + y: x \in R, \|y\| \leq d\}$$

contains any hexagon that covers some point in A . Hence, if

$$(A.2) \quad \mu(\bar{R}(d)) \leq K \frac{3\sqrt{3}}{8} d^2$$

(($3\sqrt{3}/8$) d^2 being the area of a single hexagon), then there are no more than K such hexagons.

By Lemma 7 to follow, we know that

$$(A.3) \quad \mu(\bar{R}(d)) \leq A + pd + \pi d^2$$

(with $(l - h)\pi d^2$ instead of πd^2 if R is not connected and has l components and h holes).

So, to guarantee no more than K hexagons in a cover, it is sufficient to have

$$(A.4) \quad A + pd + \pi d^2 \leq K \frac{3\sqrt{3}}{8} d^2.$$

So, let $\bar{d}$ be the smallest solution of (A.4), that is the positive solution of

$$(K(3\sqrt{3}/8) - \pi)d^2 - pd - A = 0$$

(a positive solution exists for $K \geq 5$), and let

$$\bar{A} = K(3\sqrt{3}/8)\bar{d}^2 = A + p\bar{d} + \pi\bar{d}^2.$$

A simple calculation yields (A.1) for $\bar{A}$. $\square$

LEMMA 7. Let R be a connected region in the plane, with a finite perimeter p , then for $d \geq 0$

$$\mu(\bar{R}(d)) \leq \mu(R) + pd + \pi d^2 \quad (\bar{R}(d) \equiv \{x + y : x \in R, \|y\| \leq d\}).$$

(Remarks. This inequality becomes an equality if R is convex. If R is not connected and has l components and h “holes,” the inequality is valid with $(l - h)\pi d^2$ instead of d^2 .)

Proof. We consider only a polygonal set R depicted in Fig. A1 (the validity for other sets follows from simple continuity arguments). Let α_i be the (internal) corner angle at

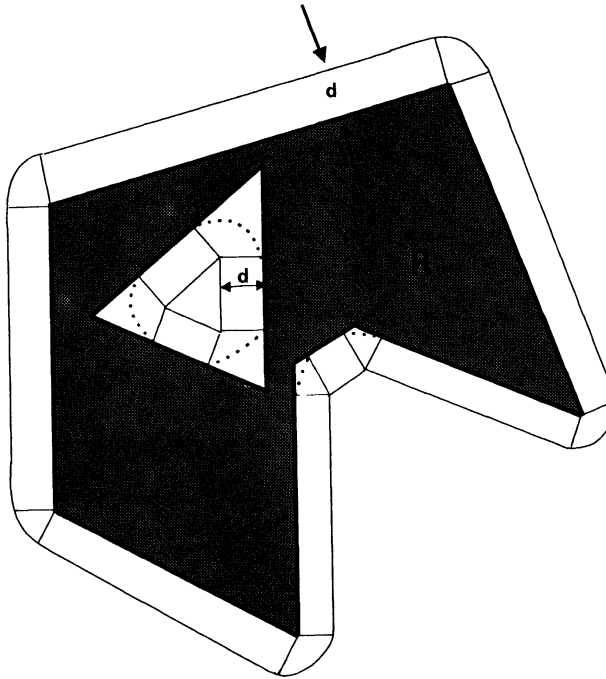


FIG. A1. A d -wide boundary strip.

the i th vertex of R . Consider the d -width boundary strip, $\bar{R}(d) \setminus R$. The strip can be partitioned into rectangles, circular sectors (at vertices with positive polar angles), and rhombuses (at vertices with negative polar angles). Let p_1 be the total length of the rectangular parts and let p_2 be the total length of the boundary of R along the rhombuses. Clearly, $p = p_1 + p_2$. Now the area of the rectangular parts is $p_1 d$, while the area of the rhombuses is $\frac{1}{2} p_2 d$. Noting that the area of the circular sectors is $\sum_{\{i: \alpha_i < \pi\}} \frac{1}{2} (\pi - \alpha_i) d^2$, the area of the strip is then

$$p_1 d + \frac{1}{2} p_2 d + \frac{1}{2} d^2 \sum_{\{i: \alpha_i < \pi\}} (\pi - \alpha_i) = p d + \frac{1}{2} d^2 \sum_{\{i: \alpha_i < \pi\}} (\pi - \alpha_i) - \frac{1}{2} p_2 d.$$

Now it is not hard to see in Fig. A1 that the boundary along the rhombus is longer than the corresponding dashed circular arcs, that is, $p_2 \geq \sum_{\{i: \alpha_i > \pi\}} (\alpha_i - \pi) d$ (with equality only in the case of a convex region R , for which $p_2 = 0$). We may easily conclude that the area of the strip is at most $p d + \frac{1}{2} \sum (\pi - \alpha) d^2$ (the summation is over all vertices).

Observe now that in general $\sum_i (\pi - \alpha_i) = 2\pi(l - h)$, where l is the number of connected components of R and h is the number of holes in R . And since $l = 1$, we have $\sum_i (\pi - \alpha_i) \leq 2\pi$, and the area of the strip is at most $p d + \pi d^2$. $\square$

Acknowledgments. We are grateful to one referee for alerting us to early references by Ballobás and Stern and by Imre and to another referee for a number of helpful suggestions for improving the paper.

REFERENCES

- [BS] B. BALLOBÁS AND N. STERN, *The optimal structure of market areas*, J. Econom. Theory, 4 (1972), pp. 174–179.
- [F1] L. FEJES-TOTH, *Sur la representation d'une population infinie par un nombre fini d'elements*, Acta. Math. Acad. Sci. Hungar., 10 (1959), pp. 299–304.
- [F2] ———, *Lagerungen in der Ebene, auf der Kugel and im Raum*. Berlin-Göttingen-Heidelberg, 1953.
- [Ha1] M. HAIMOVICH, *On the asymptotic behavior of large random K-median Problems in the plane*, unpublished notes, 1979.
- [Ha2] ———, *Large scale geometric location problems I: deterministic limit properties*, Working Paper OR 130-85, Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, January 1985.
- [Ha3] ———, *Large scale geometric location problems II: deterministic limit properties*, Working Paper OR 131-85, Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, January 1985.
- [HaM] M. HAIMOVICH AND T. L. MAGNANTI, *Location games and bounds for median problems*, Working Paper OR 133-85, Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, January 1985.
- [HM] G. HANDLER AND P. B. MIRCHANDANI, *Location on Networks, Theory and Algorithms*. MIT Press, Cambridge, MA, 1979.
- [I] M. IMRE, *Kreislagerungen auf flächen konstanter krümmung*, Acta. Math. Acad. Sci. Hungar., 15 (1971), pp. 115–121.
- [KS] R. E. KUENNE AND R. M. SOLAND, *Exact and approximate solutions to the multi-source Weber problem*, Math. Programming, 3 (1972), pp. 193–204.
- [Pa] C. H. PAPADIMITRIOU, *Worst case and probabilistic analysis of a geometric location problem*, SIAM J. Comput., 10 (1981), pp. 542–557.
- [Th] D'A. W. THOMPSON, *On Growth and Form*, Cambridge University Press, Cambridge, England, 1917, 1942.
- [Z] E. ZEMEL, *Probabilistic analysis of geometric location problems*, SIAM J. Algebraic Discrete Methods, 6 (1985), pp. 189–200.

SEARCHING FOR TWO OBJECTS BY UNDERWEIGHT FEEDBACK*

KER-I KO†

Abstract. A group testing problem with two irregular coins and with a test device that detects only an underweight coin is considered. The maximum number of tests required to identify two irregulars, one overweight and one underweight, is known to have a lower bound of $2 \log n$.¹ We present a procedure that gives an upper bound of $2 \log n + O(\log \log n \sqrt{\log n})$.

Key words. group testing, upper bound

AMS(MOS) subject classification. 68P10

1. Introduction. In a recent paper, Hwang [2] pointed out that in combinatorial search theory it is often straightforward to find an optimal procedure to search for one object while it is more difficult to search optimally for two objects. Hwang reviewed a number of two-object searching problems that use test-type devices. Among all models of searching problems studied in [2], the following problem, called Model U, has a big gap between the best known upper bound and the best known lower bound.

MODEL U (The Underweight Model). *Let x and y be two irregulars in a set A of n coins. A test of a subset S of coins yields the feedback $F(S) = 1$ if $y \in S$ and $x \notin S$, and yields the feedback $F(S) = 0$ otherwise. Find a procedure to minimize the maximum number of tests needed to identify both x and y with the above feedback function F .*

This is called the underweight model because this feedback function arises from the testing devices that detect only underweight coins. (In the above, x is an overweight coin and y an underweight coin.) The best known lower bound for this problem is the information-theoretic bound $\log [n(n-1)]$, or approximately $2 \log n$. This bound comes from the observation that the initial solution space (the set of all pairs of candidates for x and y) has size $n(n-1)$ and that each test can split a solution space into only two new solution spaces. The best known upper bound, given by Hwang [2], is $3 \log n - 1$.

In this paper, we present a procedure that requires, in the worst case, $2 \log n + O(\log \log n \sqrt{\log n})$ tests to identify x and y . We do not know whether this upper bound can be further improved, although intuitively a slightly better procedure exists. Our results suggest that the optimal procedure for Model U is difficult to find, and it supports the general phenomenon observed by Hwang and other researchers.

2. The main result. We first describe a basic procedure that provides an upper bound approximately $3 \log n$. The basic procedure is then applied recursively to obtain a better procedure.

We define the solution space associated with a set of tests $\{S_1, \dots, S_t\}$ and their feedbacks to be the set of all pairs (u, v) such that $x = u$ and $y = v$ are consistent with the tests and feedbacks; i.e., the solution space is $\{(u, v) | u \notin S_i \text{ and } v \in S_i \text{ for all } S_i \text{ with } F(S_i) = 1; \text{ and } u \in S_i \text{ or } v \notin S_i \text{ for all } S_i \text{ with } F(S_i) = 0\}$.

In our procedure, we consider two types of solution spaces. In the first type, there exist $2p$ pairwise disjoint subsets $B_1, \dots, B_p, C_1, \dots, C_p$ of A , each B_i of size m and

* Received by the editors March 10, 1986; accepted for publication (in revised form) September 14, 1987. This research was supported in part by the National Science Foundation under grants DCR83-12472 and CCR-8696135. Part of the work was done while the author was visiting the Department of Mathematics, University of California, Santa Barbara, California 93106.

† Department of Computer Science, State University of New York at Stony Brook, Stony Brook, New York 11794.

¹ Throughout the paper, we write $\log n$ to denote $\log_2 n$.

each C_i of size l , such that $x \in \bigcup_{i=1}^p B_i$ and that $x \in B_i$ if and only if $y \in C_i$ for all $i = 1, \dots, p$. We denote this type of solution spaces as $p(m, l)$. The second type consists of k pairwise disjoint subsets $B_1, \dots, B_p$ of A , each of size m , such that $x \in \bigcup_{i=1}^p B_i$ and that $x \in B_i$ if and only if $y \in B_i$ for all $i = 1, \dots, p$. We denote the second type of solution spaces as $p(m)$. The initial solution space is simply (n) .

We now discuss how to handle each type of solution spaces. The first type deals with two irregular coins known to be in two disjoint sets and so a simple binary search suffices.

LEMMA 1. *Given a solution space $p(m, l)$, there is a procedure to identify x and y in $\lceil \log m \rceil + \lceil \log (pl) \rceil$ tests.*

Proof. Assume that $B_1, \dots, B_p, C_1, \dots, C_p$ are pairwise disjoint subsets of A , each B_i of size m and each C_i of size l , such that $x \in \bigcup_{i=1}^p B_i$ and that $x \in B_i$ if and only if $y \in C_i$ for all $i \leq p$. Note that a test $S \subseteq \bigcup_{i=1}^p C_i$ receives feedback $F(S) = 1$ if and only if $y \in S$. So we can identify y by binary search in $\lceil \log (pl) \rceil$ tests. When y is found, a test $T' = T \cup \{y\}$ for any $T \subseteq \bigcup_{i=1}^p B_i$ receives feedback $F(T') = 0$ if and only if $x \in T$. So, $\lceil \log m \rceil$ more tests suffice to identify x . $\square$

Remark. The above procedure is not necessarily the optimal, but it differs from the optimal procedure by at most one test. It is interesting to observe that for the solution space $p(m, l)$, the problem of Model U is dual to the problem of Model G (group testing) discussed in [2]. More precisely, let $B_1, \dots, B_p, C_1, \dots, C_p$ be defined as in the above proof. Then, for any sets $S_1 \subseteq \bigcup_{i=1}^p B_i$ and $S_2 \subseteq \bigcup_{i=1}^p C_i$, we have $F_U(S_1 \cup S_2) = 1$ if and only if $F_G(S_1 \cup (\bigcup_{i=1}^p C_i - S_2)) = 0$, where F_U and F_G are the feedback functions for Model U and Model G, respectively. Chang and Hwang [1] have done a careful analysis for the problem of Model G with this type of solution spaces. In particular, they gave an optimal procedure for the solution space (m, l) which uses exactly $\lceil \log (ml) \rceil$ tests.

For the second type of solution spaces, we first consider a special case of space (n) , where $n = \binom{t}{k}$ for some $t > k > 0$.

LEMMA 2. *Let t, k be integers such that $0 < k < t$. Also let A have size $n = \binom{t}{k}$. Then, there exist t subsets $B_1, \dots, B_t$ of A such that*

- (i) *each B_i , $1 \leq i \leq t$, has size $\binom{t-1}{k-1}$,*
- (ii) *each $z \in A$ occurs in exactly k B_i 's, and*
- (iii) *for any two distinct elements a, b in A , there is a B_i , $1 \leq i \leq t$, such that $a \notin B_i$ and $b \in B_i$.*

Proof. Identify A with the collection of k -subsets of $\{1, \dots, t\}$, and for each i , $1 \leq i \leq t$, take B_i to be the collection of k -subsets of $\{1, \dots, t\}$ containing the element i . $\square$

Example. Let $t = 4$ and $k = 2$. Then, $n = 6$. For a set $A = \{a_1, \dots, a_6\}$, Lemma 2 shows that there are four tests such that for at least one of them the feedback is equal to 1. These four tests are $\{a_1, a_2, a_3\}$, $\{a_1, a_4, a_5\}$, $\{a_2, a_4, a_6\}$, and $\{a_3, a_5, a_6\}$.

Lemmas 1 and 2 together give a procedure that identifies x and y from $n = \binom{t}{k}$ coins in at most $2 \log n + t - 1$ tests. To be more precise, let A have size n and $B_1, \dots, B_t$ subsets of set A satisfying properties of Lemma 2. We first make $t - 1$ tests: $B_1, \dots, B_{t-1}$, and either get one feedback equal to 1 or infer that B_t has feedback equal to 1. Then, the solution space is contained in one of the type

$$\left(\binom{t-1}{k}, \binom{t-1}{k-1} \right) = \left(\frac{t-k}{t} \cdot n, \frac{k}{t} \cdot n \right).$$

By Lemma 1, at most $2 \log n$ more tests are required to identify x and y .

For any t and k , $t \geq \log \binom{k}{t}$, and so the upper bound $2 \log n + t - 1$ above is actually weaker than the one given by Hwang [2]. A more careful analysis of the above procedure shows that the actual upper bound of the above procedure is about $2.27t$ [3]. However, if we apply Lemma 2 recursively, the upper bound can be improved more drastically, and so we will not give the complete analysis. The next lemma generalizes Lemma 2 to more complicated solution spaces and prepares us for the recursive procedure.

LEMMA 3. *Let $0 < k < t$ and $m = \binom{k}{t}$. Given a solution space of type $p(ml)$, there exist t tests $S_1, \dots, S_t$ such that*

(i) *if $F(S_j) = 1$ for some $j = 1, \dots, t$, then the new solution space is contained in a space of type $p((t - k)/t \cdot ml, k/t \cdot ml)$, and*

(ii) *if $F(S_j) = 0$ for all $j = 1, \dots, t$, then the new solution space is of type $pm(l)$.*

Proof. Assume that $A_1, \dots, A_p$ are pairwise disjoint sets each of size ml and satisfying the properties that $x \in \bigcup_{i=1}^p A_i$ and that $x \in A_i$ if and only if $y \in A_i$ for all $i = 1, \dots, p$. Then, divide each A_i , $1 \leq i \leq p$, into m pairwise disjoint subsets of size l : $A_i = \bigcup_{r=1}^m C_{i,r}$. Treating, for each $r = 1, \dots, m$, the set $\bigcup_{i=1}^p C_{i,r}$ as a single element and applying Lemma 2 to these m elements, we obtain t tests satisfying the properties (i) and (ii). We leave the details to the reader. $\square$

Lemma 3 may be applied to obtain a recursive procedure that outperforms the basic procedure. First let us look at an example.

Example. We consider the set $A = \{1, 2, \dots, 36\}$ which gives an initial solution space of the type (36) . Let $t = 4$ and $k = 2$. We first apply Lemma 3 (with $m = l = 6 = \binom{k}{t}$ and $p = 1$) to get the following four tests:

$$\begin{aligned} &\{1, \dots, 18\}, \\ &\{1, \dots, 6, 19, \dots, 30\}, \\ &\{7, \dots, 12, 19, \dots, 24, 31, \dots, 36\}, \\ &\{13, \dots, 18, 25, \dots, 36\}. \end{aligned}$$

Assume that all tests receive feedback 0. Then the solution space is of the type $6(6)$, defined by six sets: $\{1, \dots, 6\}$, $\{7, \dots, 12\}$, $\dots$, $\{31, \dots, 36\}$. Now apply Lemma 3 again (with $m = 6$, $l = 1$, and $p = 6$). We consider four more tests:

$$\begin{aligned} &\{6i + 1, 6i + 2, 6i + 3 \mid i = 0, \dots, 5\}, \\ &\{6i + 1, 6i + 4, 6i + 5 \mid i = 0, \dots, 5\}, \\ &\{6i + 2, 6i + 4, 6i + 6 \mid i = 0, \dots, 5\}, \\ &\{6i + 3, 6i + 5, 6i + 6 \mid i = 0, \dots, 5\}. \end{aligned}$$

These four tests must receive at least one feedback 1, and so we need only perform the first three tests to get a solution space of type $6(3, 3)$. Then, an ad hoc binary search procedure can identify x and y in six more tests (cf. Chang and Hwang [1]).

On the other hand, if for instance, at the first step, we get

$$F(\{1, \dots, 6, 19, \dots, 30\}) = 1,$$

then the new solution space is contained in

$$\{7, \dots, 18, 31, \dots, 36\} \times \{1, \dots, 6, 19, \dots, 30\}.$$

Using Chang and Hwang's [1] optimal procedure, we only need nine more tests to identify x and y .

Altogether, 13 tests are sufficient. This improves the lower bound $3 \log n - 1$ of Hwang [2] by 4.

Now we give the general procedure. It can be seen that for a fixed t , the best choice of k is $\lfloor t/2 \rfloor$.

Procedure P_t ($t \geq 1$). Let $k = \lfloor t/2 \rfloor$ and $m = \binom{t}{k}$. Assume that we are given a set A of size n with $m^{\alpha-1} < n \leq m^\alpha$. Then we add dummy members to make the initial solution space to be of the type (m^α) . We will describe 2α steps, starting with Step 0.

Step p ($0 \leq p < \alpha$). Assume that the current solution space is $m^p(m^{\alpha-p})$. Then, apply Lemma 3 to find t tests $S_1, \dots, S_t$, satisfying (i) and (ii) of Lemma 3. If $F(S_j) = 1$ for some $j = 1, \dots, t$, then obtain a space of type

$$m^p(((t-k)/t)m^{\alpha-p}, (k/t)m^{\alpha-p})$$

that contains the new solution space, and go to Step p' . If $F(S_j) = 0$ for all $j = 1, \dots, t$, then obtain the new solution space (of the type $m^{p+1}(m^{\alpha-p-1})$), and go to Step $(p+1)$.

Step p' ($0 \leq p < \alpha$). Assume that the current solution space is contained in a space of the type $m^p(((t-k)/t)m^{\alpha-p}, (k/t)m^{\alpha-p})$. Apply Lemma 1 to identify x and y .

End of Procedure P_t .

We note that Lemma 3 guarantees that the assumption made at the beginning of each step is correct. Furthermore, if we begin at Step 0 and reach Step $(\alpha-1)$, then next we must enter Step $(\alpha-1)'$. This is because, at the beginning of Step $(\alpha-1)$, we have a solution space $m^{\alpha-1}(m)$, and Lemma 2 proves that by this step there must be at least one feedback equal to 1. So, Lemmas 1 and 3, together with the above observation, imply that Procedure P_t always identifies x and y correctly.

Now we analyze the number of tests made in Procedure P_t . First, we need an inequality.

LEMMA 4. Let $t \geq 2$ and $m = \binom{t}{\lfloor t/2 \rfloor}$. Then,

$$t - \frac{1}{2} \log t - 1 \leq \log m \leq t - \frac{1}{2} \log t.$$

Assume that we enter Procedure P_t at Step 0, and continue to Step p , and then go to Step p' . Then, the number of tests performed is at most

$$\begin{aligned} M_{t,p}(n) &\leq t(p+1) + \left\lceil \log \left(\frac{t-k}{t} \cdot m^{\alpha-p} \right) \right\rceil + \left\lceil \log \left(\frac{k}{t} \cdot m^\alpha \right) \right\rceil \\ &\leq t(p+1) + \lceil \log m^{\alpha-p} \rceil + \lceil \log m^\alpha \rceil - 1. \end{aligned}$$

Since $t > \log m$, the above value increases as p increases. So, the worst case occurs when $p = \alpha - 1$. Thus, the maximum number of tests required by Procedure P_t on input n is

$$M_t(n) \leq t\alpha + \lceil \log m \rceil + \lceil \log m^\alpha \rceil - 1 \leq (2\alpha + 1)t.$$

Now, by the relation $m^{\alpha-1} < n \leq m^\alpha$, we have

$$\alpha < \frac{\log n}{\log m} + 1.$$

So,

$$M_t(n) \leq \frac{2t}{\log m} \log n + 3t \leq (2 + \varepsilon_t) \log n + 3t,$$

where

$$\varepsilon_t = 2 \left(\frac{t}{\log m} - 1 \right) \approx \frac{\log t}{\log m}.$$

Note that $\varepsilon_t \rightarrow 0$ as $t \rightarrow \infty$. So, the asymptotic bound of Procedure P_t can be arbitrarily close to the information-theoretic bound by choosing sufficiently large t . In practice, however, we must choose, for given input n , the value t with $t < 2 \log n$, since $t \geq 2 \log n$ implies $\binom{t}{k} > n$. So, for a given n , what is the best procedure among all P_t 's? From the analysis of $M_t(n)$ above, we find that letting t be approximately $\sqrt{\log n}$ produces a nice upper bound. This suggests the following procedure.

Procedure R. For given input n , let $t = \lceil \sqrt{\log n} \rceil$ and then apply Procedure P_t .

THEOREM 5. *The maximum number of tests required by Procedure R on input n is*

$$M_R(n) = 2 \log n + O(\log \log n \sqrt{\log n}).$$

Proof. From the analysis of $M_t(n)$ and Lemma 4, we get

$$M_R(n) \leq M_t(n) \leq 2 \log n + \frac{\log t + 2}{\log m} \log n + 3t.$$

Note that $\log m \approx t - (\log t)/2$ and t was chosen to be about $\sqrt{\log n}$. Substituting these values into the above inequality and simplifying the formula, we get

$$M_R(n) = 2 \log n + O(\log \log n \sqrt{\log n}). \quad \square$$

3. Discussion. It can be seen that even though Procedure R gets an upper bound quite close to the information-theoretic lower bound, it does not appear to be an optimal procedure. Intuitively, the number t at each step of the search in Procedure R should be determined dynamically by the size of the current solution space. In other words, the following procedure, with a carefully chosen function f , should behave better than Procedure R.

Procedure $D(f)$. Assume that the input is a set of size n .

repeat

if the current solution space is of the type $p(m)$

then begin

 choose $t = f(m)$;

 apply Lemma 3 to find t tests $S_1, \dots, S_t$;

if $F(S_j) = 1$ for some $j = 1, \dots, t$,

then set up the new solution space (of type $p(l_1, l_2)$) and go to 1

else set up the new solution space (of type $p(l)$) and continue

end

end repeat

1: apply Lemma 1 to identify x and y

End of Procedure $D(f)$.

It seems very difficult to give a precise analysis of the maximum number of tests made by Procedure $D(f)$, even for a simple function such as $f(m) = \lceil \sqrt{\log m} \rceil$. It remains open whether there is a function f which makes Procedure $D(f)$ optimal and, if such

an optimal function exists, what the maximum number of tests required by Procedure $D(f)$ is.

Acknowledgments. The author thanks Dr. Hwang for helpful discussions. The simple proof of Lemma 2 is suggested by Dr. Hwang as well as the referees.

REFERENCES

- [1] G. J. CHANG AND F. K. HWANG, *A group testing problem on two disjoint sets*, SIAM J. Algebraic Discrete Methods, 2 (1981), pp. 35–38.
- [2] F. K. HWANG, *A tale of two coins*, Amer. Math. Monthly, 94 (1987), pp. 121–129.
- [3] ———, personal communication, 1986.

GRAPHS THAT SPLIT ENTROPIES*

J. KÖRNER† AND K. MARTON†

Abstract. The entropy of a graph is a functional depending both on the graph itself and on a probability distribution on its vertex set. This concept is at the core of a new bounding technique for graph covering problems and has furnished the best known bounds for the problem of perfect hashing.

The basis of the technique is the sub-additivity of graph entropy with respect to the union of graphs. The tightness of the bounds depends on whether or not we have equality rather than just sub-additivity. As a first step in this analysis, we are investigating whether for a given graph G the entropies of G and $\bar{G}$ add up to the entropy of the complete graph on the same vertex set, i.e., the entropy of the underlying probability distribution.

We shall prove that for a bipartite graph G and an arbitrary probability distribution P on its vertex set the entropies of G and $\bar{G}$ add up to the entropy of P . Related problems will be discussed.

The results have interesting connections with the Ford–Fulkerson theory of network flows.

Key words. graph entropy, graph covering, perfect graphs, network flows

AMS(MOS) subject classifications. primary 05C70; secondary 68R10, 90B10, 94A15, 05C75

1. Introduction. Graph entropy can be viewed as a probabilistically refined concept of the chromatic number of a graph. It was introduced by the first author [1] as a characterization of the theoretical limit of the coding performance in a somewhat esoteric information theoretic context. Some of its basic properties have been established in the papers [2], [3]. Graph entropy reentered the scene in a paper by Fredman and Komlós [4] on the minimum number of perfect hash functions of a given range that hash all the k -element subsets of a set of a given size. Although only a simple special case of graph entropy is hidden in [4], in a streamlined version [5] of the Fredman–Komlós proof the first author pointed out that the basis of their proof technique is the sub-additivity of graph entropy with respect to the union of graphs. This observation has led us to an improvement on the original Fredman–Komlós bounds [6]. The understanding of the cases of additivity of graph entropy might lead to a further refinement of this surprisingly efficient combinatorial bounding technique. More importantly, the conditions for additivity of graph entropy seem to have interesting connections with the Ford–Fulkerson theory of flows in networks [7] and the theory of perfect graphs [8]. But let us proceed in a systematic manner and give the necessary definitions. A minimum amount of prerequisites from information theory is needed. We will be dealing with probability distributions on finite sets. Given a distribution P on the finite set $\mathcal{X}$, i.e., a weight distribution P with $P(x) \geq 0$, $x \in \mathcal{X}$, $\sum_{x \in \mathcal{X}} P(x) = 1$, the entropy of P is

$$H(P) \triangleq - \sum_{x \in \mathcal{X}} P(x) \cdot \log P(x).$$

Here and in the sequel all exp's and log's are to the base 2. Given a random variable X with range $\mathcal{X}$, the entropy of X is the entropy of the probability distribution (PD) P_X , i.e.,

$$H(X) \triangleq - \sum_{x \in \mathcal{X}} \Pr(X=x) \cdot \log \Pr(X=x).$$

The mutual information of the RV's X and Y with respective (finite) ranges $\mathcal{X}$ and $\mathcal{Y}$ is

$$I(X \wedge Y) \triangleq H(X) + H(Y) - H(X, Y),$$

* Received by the editors June 9, 1986; accepted for publication March 12, 1987.

† Mathematical Institute of the Hungarian Academy of Sciences, 1053 Budapest, Hungary.

where $H(X, Y)$ is the entropy of the RV (X, Y) , also called the joint entropy of (X, Y) . It is well known and easy to prove that $I(X \wedge Y)$ is nonnegative, and equals zero if and only if X and Y are independent. (Our information theoretic terminology and notation come from the book [9].)

The paper is not entirely self-contained. All the definitions are included, but some facts will be borrowed from [5] without proof.

A graph will always mean a simple undirected graph with no loops or multiple edges.

DEFINITION 1. Let P be an arbitrary probability distribution on the vertex set of the graph G . Let $\mathcal{Y}(G)$ denote the family of all the maximal independent sets of G . The entropy of the pair (G, P) is defined as

$$H(G, P) \triangleq \min_{\substack{X \in Y \in \mathcal{Y}(G) \\ P_X = P}} I(X \wedge Y).$$

(An independent set is an edge-free induced subgraph of G . The notation $X \in Y \in \mathcal{Y}(G)$ means that Y is a RV ranging over the maximal independent sets of G while X ranges over the vertices of G in such a manner that with probability 1 the point X is contained in the set Y .)

Given the graphs F and G with the same vertex set $\mathcal{X}$, their union $F \cup G$ is a graph also having vertex set $\mathcal{X}$. The edges of $F \cup G$ are all the edges contained in either F or G . It was proved in [5] that graph entropy is a sub-additive functional with respect to the union of graphs, i.e.,

LEMMA K. *For arbitrary graphs F and G with common vertex set $\mathcal{X}$ and any distribution P on $\mathcal{X}$,*

$$(1) \quad H(F \cup G, P) \leq H(F, P) + H(G, P).$$

We say that the graph F covers G , in symbols $G \subset F$, if the two graphs have the same vertex set and their union is F . Several interesting combinatorial problems including that of perfect hashing can be formulated in this language: a given graph G has to be covered by a possibly small number t of graphs G_i , $i = 1, 2, \dots, t$, where the G_i 's are to be chosen from a given family of graphs, i.e.,

$$G \subset \bigcup_{i=1}^t G_i.$$

A lower bound on t (depending on G and the family) can be obtained using (1), provided that we are able to compute the entropies $H(G, P)$, $H(G_i, P)$ for a suitable P . This technique is applied to perfect hashing in [5] and [6]. Although the bounds obtained are the best ones around, their tightness has not been established. Obviously, a bound based on the sub-additivity of graph entropies can be tight only if we have equality in (1). Hence our interest in this question. Far from being able to answer it, we will address a special case. Later on, we shall see that this special case is important also for quite different reasons than those cited above.

Let us denote by $\bar{G}$ the complement of the graph G . ($\bar{G}$ has the same vertex set as G while two distinct vertices are adjacent in $\bar{G}$ if and only if they are not so in G .) Clearly, the union of G and $\bar{G}$ is the complete graph on the vertex set of G . One easily sees from the definition that the entropy of the complete graph equals the entropy of the underlying PD. Hence (1) becomes

$$(2) \quad H(P) \leq H(G, P) + H(\bar{G}, P).$$

In this paper, we shall be dealing with the case of equality in the last inequality. This problem was first raised in [2] from an information-theoretic viewpoint. We shall need some of the concepts and results introduced there and in the follow-up paper [3].

2. Graphs that split entropies.

DEFINITION 2. The graph G is *weakly splitting* if for at least one nowhere vanishing PD P on its vertex set we have equality in (2).

The graph G is *strongly splitting* if (2) holds for *every* PD P on its set of vertices.

We would like to characterize the above two categories of graphs. To this end, let us introduce the following definition.

DEFINITION 3. The graph G is *normal* if the vertex set of G has a covering by independent sets of G and another one by complete subgraphs of G so that every pair of subsets from the two different coverings has a nonvoid intersection.

By definition, G is normal if and only if so is $\bar{G}$. It was proved in [3] that every perfect graph is normal, but the converse is not true, since every odd cycle of length at least 9 is normal without being perfect. On the other hand, odd cycles of length 5 and 7 are not even normal. It was shown in [2] that normal graphs are weakly splitting. In fact, the converse is also true, so that we have the following.

PROPOSITION 1. *A graph G is weakly splitting if and only if it is normal.*

Proof. For the sake of completeness, we shall prove both the “if” and the “only if” parts of the statement.

1. If G is normal, then it is weakly splitting. Let G be a normal graph. Then there is a family of independent sets $\mathcal{Y}'$ of G and a family of independent sets $\mathcal{Z}'$ of $\bar{G}$ so that any pair of members from the two different families have a nonvoid intersection. Without loss of generality, we can suppose that every independent set involved is maximal. Let us introduce arbitrary PD's on $\mathcal{Y}'$ and $\mathcal{Z}'$ and let the RV's Y and Z ranging over $\mathcal{Y}'$ and $\mathcal{Z}'$, respectively, have these distributions. Let Y and Z be independent RV's. By our hypothesis, $Y \cap Z$ is a nonvoid subset of the vertex set $\mathcal{X}$ of G and $\bar{G}$. Since this set must be an independent set both in G and $\bar{G}$, it has a single element. Let us define the random variable X with range $\mathcal{X}$ by

$$X \triangleq Y \cap Z.$$

Set $P = P_X$. Then, by (2) and the definition of graph entropy,

$$H(X) \leq H(G, P) + H(\bar{G}, P) \leq I(X \wedge Y) + I(X \wedge Z).$$

Now, by the so-called Data-Processing Lemma (cf. [9])

$$I(X \wedge Y) \leq I(XZ \wedge Y),$$

and hence

$$H(X) \leq I(XZ \wedge Y) + I(X \wedge Z).$$

It can be seen from the very definition of mutual information that the right-hand side of the last inequality is symmetric in X , Y and Z and thus

$$H(X) \leq I(YZ \wedge X) + I(Y \wedge Z).$$

Now, since X is a deterministic function of the RV YZ , we see that $I(YZ \wedge X) = H(X)$ while $I(Y \wedge Z) = 0$ since Y and Z are independent. Hence all our inequalities become equalities and thus G is indeed weakly splitting for the given P .

The converse can be proved more or less along the same lines.

2. If G is weakly splitting, then it is normal. In fact, let us have

$$H(P) = H(G, P) + H(\bar{G}, P)$$

for some nowhere vanishing P on the vertex set $\mathcal{X}$ of G . Let the joint distribution of X and Y attain the minimum in the definition of $H(G, P)$ and, likewise, let the joint distribution of X and Z attain the minimum for $H(\bar{G}, P)$. Since this puts no further condition on the joint distribution of X , Y and Z , we can suppose that Y is independent of Z given X . Then

$$\begin{aligned} H(X) &= H(G, P) + H(\bar{G}, P) = I(X \wedge Y) + I(X \wedge Z) \\ &= I(ZX \wedge Y) + I(X \wedge Z) = I(YZ \wedge X) + I(Y \wedge Z) \\ &= H(X) + I(Y \wedge Z). \end{aligned}$$

Comparing the two ends we see that $I(Y \wedge Z) = 0$, i.e., Y and Z are independent. This means, however, that their ranges determine two coverings as required by the definition of a normal graph. $\square$

The above proof yields a simple way of calculating graph entropy in certain cases.

LEMMA 1. *If $\mathcal{Y}$ is the family of the maximal independent sets of G , $\mathcal{Z}$ is the family of its maximal complete subgraphs and there exist independent RV's Y and Z with respective ranges $\mathcal{Y}$ and $\mathcal{Z}$ for which the RV $X = Y \cap Z$ has distribution P , then*

$$H(G, P) = I(X \wedge Y). \quad \square$$

Next we would like to characterize the class of strongly splitting graphs. However, this does not seem to be an easy task and only partial results are available. First of all, let us see that not every weakly splitting graph is strongly splitting. This will follow immediately from the next theorem if we recall that by Proposition 1 and the aforementioned results [3] on normal graphs every cycle of length ≥ 9 is weakly splitting.

THEOREM 1. *No odd cycle of length strictly greater than 3 is strongly splitting.*

Proof. We claim that if G is an odd cycle of length strictly greater than 3 and P is the uniform distribution on G , then (2) cannot hold with equality. Let us proceed indirectly. By the same argument as in the second part of the proof of Proposition 1, if (2) holds with equality for P , then there must be an RV Y with distribution concentrated on the members of a family of maximal independent sets of G and a RV Z with distribution concentrated on the members of a family of maximal complete subgraphs of G so that the ranges of both Y and Z cover the vertex set $\mathcal{X}$, the RV's Y and Z are independent, and we have

$$(3) \quad P(x) = \sum_{x \in y} \Pr \{Y = y\} \cdot \sum_{x \in z} \Pr \{Z = z\}.$$

The maximal complete subgraphs of G are edges. Let $\mathcal{Z}$ denote the family of those edges which are taken by Z with nonzero probability. For the sake of better understanding, let us give the cycle an arbitrary orientation. It is clear that the edges in $\mathcal{Z}$ are partitioned into classes such that within each of them the edges which are consecutive in the cyclic orientation are adjacent, while two consecutive classes are separated by just one edge, lying outside of $\mathcal{Z}$. Let us call such a class of edges an "island."

Let us now look at the covering of the vertex set by those independent sets of G which are taken by Y with nonzero probability. Let $\mathcal{Y}$ denote their family. For brevity, let us call the members of $\mathcal{Y}$ *colors*. Consider two arbitrary consecutive islands. These must have two adjacent endpoints. Let us call them x_1 and x_2 , in the order of the given orientation.

We claim that

$$(4) \quad \Pr \{x_i \in Y\} \geq \frac{1}{2}, \quad i = 1, 2.$$

In fact, since x_2 is the endpoint of an island, it is covered by a single edge, z , say. Then, by (3),

$$\Pr \{x_2 \in Y\} \cdot \Pr \{Z = z\} \geq \Pr \{x \in Y\} \cdot \Pr \{Z = z\},$$

where x is that point of our island which is adjacent to x_2 . Hence

$$\Pr \{x_2 \in Y\} \geq \Pr \{x \in Y\}.$$

Noticing that x_2 and x are the two points of an edge from our covering, we see that they must contain all the colors, and since they are adjacent, they must have disjoint sets of colors. Thus

$$\Pr \{x_2 \in Y\} + \Pr \{x \in Y\} = 1.$$

This and the last inequality establish (4) for $i = 2$, and, clearly, the case $i = 1$ is the very same.

On the other hand, the points x_1 and x_2 are adjacent in G , and therefore

$$\Pr \{x_1 \in Y\} + \Pr \{x_2 \in Y\} = 1.$$

In other words, we must have equality in (4). This means, however, that the total probability of the set of edges covering x must be the same as the probability of z , or, which is the same that x is covered by just one element of $\mathcal{Z}$.

We conclude that every island consists of a single edge. However, in this case G cannot be an odd cycle. Hence the desired contradiction. $\square$

Let us turn to strongly splitting graphs. A trivial example is the complete graph. Less trivial examples are to be found in [2].

3. Bipartite graphs are strongly splitting. We have not been able to characterize the class of strongly splitting graphs. The next theorem cannot be regarded but as a first attempt. The proof heavily relies on the classical Gale–Ryser theorem on network flows. However, there are nontrivial classes of graphs other than bipartite graphs that are strongly splitting. Finding the proper way to tackle the problem in the general case might be a challenge for those interested in extensions of the Ford–Fulkerson theory of network flows.

THEOREM 2. *Every bipartite graph is strongly splitting.*

Proof. Let G be a bipartite graph with vertex set $\mathcal{X}$. Then there exist two disjoint independent sets, $\mathcal{A}$ and $\mathcal{B}$, say, the union of which is $\mathcal{X}$. Given a set $\mathcal{C} \subset \mathcal{X}$, let $\Gamma \mathcal{C}$ denote the set of those vertices of G which are adjacent to at least one vertex in $\mathcal{C}$.

1. We claim that if for every $\mathcal{A}' \subset \mathcal{A}$,

$$(5) \quad \frac{P(\mathcal{A}')}{P(\mathcal{A})} \leq \frac{P(\Gamma \mathcal{A}')}{P(\mathcal{B})}$$

then G and P satisfy the equality in (2) and we have $H(G, P) = H((P(\mathcal{A}), P(\mathcal{B})))$ where the latter is the entropy of the binary distribution $(P(\mathcal{A}), P(\mathcal{B}))$.

This claim is an immediate consequence of the Gale–Ryser theorem, cf. [7] or [10]. In fact, we are looking for a probability distribution on the set of maximal complete subgraphs $\mathcal{Z}$ of G such that if the RV Z on $\mathcal{Z}$ has this distribution, and the RV Y takes its values in $\{\mathcal{A}, \mathcal{B}\}$, then (3) holds. By our assumption

$$\Pr \{Y = \mathcal{A}\} = P(\mathcal{A}) = 1 - \Pr \{Y = \mathcal{B}\},$$

and thus we are looking for some PD R on $\mathcal{Z}$, the set of the edges of G , such that for every $x \in \mathcal{A}$,

$$\sum_{x \in z} R(z) = \frac{P(x)}{P(\mathcal{A})}$$

and for every $x' \in \mathcal{B}$,

$$\sum_{x' \in z} R(z) = \frac{P(x')}{P(\mathcal{B})}.$$

The question of the existence of such a PD R is equivalent to the Gale–Ryser problem and (5) is just the sufficient condition of Gale and Ryser for such an R to exist. This establishes our first claim.

However, (5) is also a necessary condition for the requested R to exist, and thus if there is no R as needed, then (5) fails to hold for at least one $\mathcal{A}' \subset \mathcal{A}$. In this case we will construct a decomposition of G .

2. We claim that there is a partition of $\mathcal{A}$ and a corresponding partition of $\mathcal{B}$,

$$\mathcal{A} = \bigcup_{i=1}^r \mathcal{A}_i, \quad \mathcal{B} = \bigcup_{i=1}^r \mathcal{B}_i$$

such that

$$(6) \quad \mathcal{B}_i \triangleq \Gamma \mathcal{A}_i \setminus \bigcup_{j < i} \Gamma \mathcal{A}_j,$$

$$\frac{P(\mathcal{A}_1)}{P(\mathcal{B}_1)} \geq \frac{P(\mathcal{A}_2)}{P(\mathcal{B}_2)} \geq \dots \geq \frac{P(\mathcal{A}_r)}{P(\mathcal{B}_r)} \geq 1$$

and

$$(7) \quad \frac{P(\mathcal{A}')}{P(\mathcal{A}_i)} \leq \frac{P(\Gamma \mathcal{A}' \cap \mathcal{B}_i)}{P(\mathcal{B}_i)} \quad \text{for every } \mathcal{A}' \subset \mathcal{A}_i.$$

The last inequality means that the Gale–Ryser conditions hold for the graph induced by G on $\mathcal{A}_i \cup \mathcal{B}_i$ and the corresponding measure. (Although the latter is not a PD any more, notice that the original conditions (5) are independent of normalization.)

The claim is obviously true with $r = 1$ if (5) holds for every $\mathcal{A}' \subset \mathcal{A}$. If (5) fails to hold, let $\mathcal{A}_1$ be the set maximizing

$$\frac{P(\mathcal{A}')}{P(\Gamma \mathcal{A}')}.$$

Clearly, with such a choice, (7) will be satisfied for every $\mathcal{A}' \subset \mathcal{A}_1$. If $\mathcal{A}_1, \dots, \mathcal{A}_{i-1}$ and the corresponding $\mathcal{B}_j$'s have been constructed, choose $\mathcal{A}_i$ to be $\mathcal{A} \setminus \bigcup_{j < i} \mathcal{A}_j$ if the maximum of

$$(8) \quad \frac{P(\mathcal{A}')}{P(\mathcal{A} \setminus \bigcup_{j < i} \mathcal{A}_j)} \cdot \frac{P(\mathcal{B} \setminus \bigcup_{j < i} \mathcal{B}_j)}{P(\Gamma \mathcal{A}' \setminus \bigcup_{j < i} \mathcal{B}_j)}$$

within $\mathcal{A}' \subset \mathcal{A} \setminus \bigcup_{j < i} \mathcal{A}_j$ does not exceed 1, else choose $\mathcal{A}_i$ to be the maximizing $\mathcal{A}'$.

First of all, this is obviously a well-described algorithm that does construct a partition of $\mathcal{A}$ and hence of $\mathcal{B}$. Further, it is immediate from the definition of $\mathcal{A}_i$ that it satisfies (7). It remains to establish (6). Since $\mathcal{A}_i$ maximizes (8) within $\mathcal{A} \setminus \bigcup_{j < i} \mathcal{A}_j$, it follows that

$$\frac{P(\mathcal{A}_i)}{P(\mathcal{B}_i)} \geq \frac{P(\mathcal{A}_i \cup \mathcal{A}_{i+1})}{P(\mathcal{B}_i \cup \mathcal{B}_{i+1})} = \frac{P(\mathcal{A}_i) + P(\mathcal{A}_{i+1})}{P(\mathcal{B}_i) + P(\mathcal{B}_{i+1})}.$$

Comparing the two ends of this, we get

$$\frac{P(\mathcal{A}_i)}{P(\mathcal{B}_i)} \geq \frac{P(\mathcal{A}_{i+1})}{P(\mathcal{B}_{i+1})}.$$

This decomposition of $\mathcal{A}$ and $\mathcal{B}$ satisfying (6) and (7) leads to the proper choice of the PD's of Y and Z for (3) to hold.

Let us denote by G_i the induced subgraph of G having vertex set $\mathcal{A}_i \cup \mathcal{B}_i$. By (7) this bipartite graph satisfies the Gale–Ryser conditions (5) for every i . Hence there is a PD R_i on the edges of G_i such that for every i and every $x \in \mathcal{A}_i$

$$\sum_{\substack{z: x \in z \\ z \in \mathcal{Z}(G_i)}} R_i(z) = \frac{P(x)}{P(\mathcal{A}_i)}$$

where $\mathcal{Z}(G_i)$ is the set of edges of G_i , and likewise, for every $x' \in \mathcal{B}_i$

$$\sum_{\substack{z: x' \in z \\ z \in \mathcal{Z}(G_i)}} R_i(z) = \frac{P(x')}{P(\mathcal{B}_i)}.$$

We define a PD R on $\mathcal{Z}$ by

$$R(z) = \begin{cases} [P(\mathcal{A}_i) + P(\mathcal{B}_i)] \cdot R_i(z) & \text{if } z \in \mathcal{Z}(G_i), \\ 0 & \text{else.} \end{cases}$$

Next we will construct a PD Q on $\mathcal{Y}$. Let us denote

$$y_i \triangleq \left(\bigcup_{j < i} \mathcal{A}_j \right) \cup \left(\bigcup_{k \geq i} \mathcal{B}_k \right).$$

Clearly, the sets $y_i \in \mathcal{Y}$ are independent and cover $\mathcal{A} \cup \mathcal{B}$. We claim that there is a PD Q concentrated on y_i , $i = 1, 2, \dots, r+1$ such that the above R and this Q satisfy (3). In order to see this, note that if $x \in \mathcal{B}_i$, then x is covered by $y_1, \dots, y_i$. It is an easy consequence of the monotonicity of $P(\mathcal{A}_i) \cdot [P(\mathcal{B}_i)]^{-1}$, (cf. (6)) that there exists a PD Q on the y_i 's such that

$$(9) \quad \sum_{j=1}^i Q(y_j) = \frac{P(\mathcal{B}_i)}{P(\mathcal{A}_i) + P(\mathcal{B}_i)} \quad \text{for every } i \leq r.$$

Notice that for such a Q (9) implies

$$(10) \quad \sum_{j=i+1}^{r+1} Q(y_j) = \frac{P(\mathcal{A}_i)}{P(\mathcal{A}_i) + P(\mathcal{B}_i)} \quad \text{for every } i \geq 1,$$

and $y_{i+1}, \dots, y_{r+1}$ are those independent sets among the y_j 's which cover an $x \in \mathcal{A}_i$. Comparing (9) and (10) with the definition of R it is immediate that the independent RV's Y and Z with PD's $P_Y = Q$, $P_Z = R$ satisfy (3). $\square$

Discussion. Hidden in the proof of Theorem 2 is an interesting information theoretic sufficient and necessary condition for a maximum flow in a bipartite transportation network to saturate all the sink arcs (cf. [10, p. 84]).

For our purposes, a bipartite transportation network is given by a bipartite graph G with vertex set $\mathcal{X} = \mathcal{A} \cup \mathcal{B}$ where the disjoint sets $\mathcal{A}$ and $\mathcal{B}$ are independent sets in G , and a nonnegative measure (not necessarily a probability measure) Q on the vertices of G . A flow is then a measure R on the edge set $\mathcal{Z}$ of G . The measure R “saturates the sink arcs” if

$$(11) \quad \sum_{\substack{z: x \in z \\ z \in \mathcal{Z}}} R(z) = Q(x)$$

for every $x \in \mathcal{X}$. The Gale–Ryser theorem [10, Cor. p. 84] gives the sufficient and necessary condition

$$Q(\mathcal{A}') \leq Q(\Gamma \mathcal{A}') \quad \forall \mathcal{A}' \subseteq \mathcal{A}$$

for a saturating flow to exist. We claim the following.

PROPOSITION 2. *In a bipartite transportation network $(G, \mathcal{A}, \mathcal{B}, Q)$ a saturating flow R exists if and only if for the PD*

$$(12) \quad P(x) \triangleq Q(x) \cdot [Q(\mathcal{X})]^{-1}$$

on $\mathcal{X} = \mathcal{A} \cup \mathcal{B}$ the graph G has maximum entropy among all the graphs admitting the bipartition $(\mathcal{A}, \mathcal{B})$, i.e.,

$$(13) \quad H(G, P) = H((P(\mathcal{A}), P(\mathcal{B}))).$$

Proof. If for the given $(G, \mathcal{A}, \mathcal{B}, Q)$ there is a saturating flow R , i.e., (11) holds, then we have (13) by the first part of the proof of Theorem 2.

Conversely, if for the bipartite graph $(G, \mathcal{A}, \mathcal{B})$ and the PD P on the vertex set of G derived from Q by (12) we have (13), then we claim that there is a saturating flow R , i.e., a measure R satisfying (11) on the set of edges of G . In fact, by Theorem 2 we have $H(P) = H(G, P) + H(\bar{G}, P)$. By (13), $H(G, P) = I(X \wedge Y)$ where $Y \in \{\mathcal{A}, \mathcal{B}\}$ is a deterministic function of X . Let X, Z achieve the minimum in the definition of $H(\bar{G}, P)$. Then, as in the second part of the proof of Proposition 1, we see that Y and Z must be independent if we suppose as we may that they are conditionally independent given X . Hence (11) follows by choosing R to be the suitably normalized distribution of the RV Z . $\square$

It is to be expected that the relationship between strongly splitting graphs and saturating flows carries over to other classes of graphs and networks.

From a purely graph-theoretic viewpoint strongly splitting graphs are hoped to be in an interesting relationship with perfect graphs. We have some reasons to believe in the following conjecture.

Conjecture 1. Every perfect graph is strongly splitting.

Finally, let us observe that this and the strong perfect graph conjecture of Berge (cf. [8]) would imply by our Theorem 1 the following conjecture:

Conjecture 2. A graph is strongly splitting if and only if it is perfect.

Were Conjecture 2 valid, one could say that perfection is a relationship between G and $\bar{G}$ characterized by equality in (2). This relationship could then be generalized to one between 3 edge-disjoint graphs G_1, G_2, G_3 such that their union is the complete graph on their common vertex set and for every PD P on the latter

$$H(P) = \sum_{i=1}^3 H(G_i, P).$$

REFERENCES

- [1] J. KÖRNER, *Coding of an information source having ambiguous alphabet and the entropy of graphs*, in Transactions of the 6th Prague Conference on Information Theory, Academia, Prague, 1973, pp. 411–425.
- [2] J. KÖRNER AND G. LONGO, *Two-step encoding of finite memoryless sources*, IEEE Trans. Inform. Theory, 19 (1973), pp. 778–782.
- [3] J. KÖRNER, *An extension of the class of perfect graphs*, Studia Sci. Math. Hungar., 8 (1973), pp. 405–409.

- [4] M. FREDMAN AND J. KOMLÓS, *On the size of separating systems and perfect hash functions*, SIAM J. Algebraic Discrete Methods, 5 (1984), pp. 61–68.
- [5] J. KÖRNER, *Fredman–Komlós bounds and information theory*, SIAM J. Algebraic Discrete Methods, 7 (1986), pp. 560–570.
- [6] J. KÖRNER AND K. MARTON, *New bounds for perfect hashing via information theory*, European J. Combin., submitted.
- [7] E. L. LAWLER, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.
- [8] M. C. GOLUBIC, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York, 1980.
- [9] I. CSISZÁR AND J. KÖRNER, *Information Theory: Coding Theorems for Discrete Memoryless Systems*, Academic Press, New York, 1982.
- [10] C. BERGE, *Graphs and Hypergraphs*, 2nd edition, North-Holland, Amsterdam, 1976.

UNIVERSAL CHAINS AND WIRING LAYOUTS*

M. S. PATERSON†

Abstract. A universal chain of order m for a metric space S is a sequence of $m - 1$ real numbers such that any multiset of m points from S can be sequenced so that the distances between successive points are no larger than the corresponding numbers in the chain. The length of a chain is the sum of its numbers. Upper and lower bounds are proved for the lengths of universal chains for the unit interval, unit square and for higher dimensions. Applications to wiring layouts in circuit boards are presented.

Key words. routing, layout, permutations, circuit board, sequencing

AMS(MOS) subject classifications. 68R10, 68Q35, 05C35, 94C15

1. Definitions. A finite sequence of nonnegative real numbers is a **chain**. The **length** of a chain is the sum of its numbers. A chain $a_1, \dots, a_{m-1}$ **covers** a multiset of m points in a metric space if there is an ordering $x_1, \dots, x_m$ of these points such that for $i = 1, \dots, m - 1$ the distance $d(x_i, x_{i+1})$ is less than or equal to a_i . A **universal chain of order m for a metric space S** is a chain which covers each multiset of size m in S .

The results obtained here concern universal chains for the unit interval, unit square and unit hypercubes in higher dimensions. For the sake of our main application we will use the L_∞ metric, $d(\langle u_1, u_2, \dots \rangle, \langle v_1, v_2, \dots \rangle) = \max \{|u_1 - v_1|, |u_2 - v_2|, \dots\}$. Other L -metrics merely yield different constants in some of our theorems.

We define $u_d(m)$ to be the minimum length of a universal chain of order m for the d -dimensional hypercube, $[0, 1]^d$, over the L_∞ metric. We use $\log$ to mean $\log_2$ throughout.

2. Main results. We present our combinatorial results in order of increasing dimension: one, two, many. The applications appear in § 3. The upper bound in Theorem 1 was proved by Tom Leighton, improving the result of an earlier version of this paper.

THEOREM 1 (Leighton and Paterson).

$$\Omega((\log m)^2) \leq u_1(m) \leq O((\log m)^3).$$

Proof. For the upper bound we construct a universal chain recursively. The resulting path visiting the elements of a multiset may be described roughly as follows. A rather compact multiset of elements with a binary tree-like structure is identified and these elements are visited in sequence using a suitable chain. Then the remaining multiset is visited recursively. The details of this construction are given below.

We shall prove the following bound:

$$u_1(m) \leq c \cdot (\log m)^3 + 1 \quad \text{where } c = (\ln 2)/3.$$

Since there must exist some pair of points in the interval with distance apart no more than $1/(m - 1)$, the chain $1/(m - 1), 1, 1, \dots, 1$ of total length $(m - 2) + 1/(m - 1)$ is universal. This suffices to prove the bound for all $m \leq 31$, and yields a base for our induction. The inductive step which follows actually holds for all $m \geq 10$.

Let the integer d be such that

$$d \cdot 2^d < m \leq (d + 1) \cdot 2^{d+1}.$$

* Received by the editors February 1, 1987; accepted for publication February 13, 1987.

† Department of Computer Science, University of Warwick, Coventry, CV4 7AL, England. This author was supported by an Information Technology Senior Fellowship from the Science and Engineering Research Council of the United Kingdom.

Working through the unit interval from left to right we identify the sequence of disjoint pairs of elements whose distance apart is at most 2^{-d} . Since each element which is unpaired by this process, except perhaps the rightmost, is followed by an empty interval of size greater than 2^{-d} , the number of such elements is at most 2^d . Next we identify in a similar way disjoint pairs of these pairs whose distance apart is at most 2^{-d+1} . The number of unpaired pairs is at most 2^{d-1} . Continuing in this way we identify disjoint pairs of (2^{r-1}) -tuples with separating distance at most 2^{-d+r-1} , for $r = 1, 2, \dots, d$. The total number of elements which are excluded at some stage of this pairing process is at most

$$2^d + 2^d + \dots + 2^d = d \cdot 2^d < m.$$

Hence the process generates at least one 2^d -tuple satisfying the given distance constraints. The elements of this 2^d -tuple can be visited in left-to-right order using the following chain:

$$2^{-d}, 2^{-d+1}, 2^{-d}, 2^{-d+2}, 2^{-d}, 2^{-d+1}, 2^{-d}, 2^{-d+3}, \dots$$

Half the links have size 2^{-d} , one quarter have size 2^{-d+1} , one eighth have size 2^{-d+2} , and so on. The total length is therefore exactly $d/2$. Our universal sequence begins in this way, then has a unit link and continues recursively with a universal sequence for the remaining $m - 2^d$ elements. We have therefore shown the following inequality:

$$u_1(m) \leq d/2 + 1 + u_1(m - 2^d).$$

Now,

$$u_1(m) \leq d/2 + 1 + c \cdot (\log(m - 2^d))^3 + 1 \quad \text{by the inductive hypothesis}$$

$$\leq d/2 + 1 + c \cdot (\log m)^3 + 1 - 2^d \cdot c \cdot 3(\log m)^2 / (m \cdot \ln 2)$$

since the derivative of $(\log m)^3$ is a decreasing function of m

$$\leq c \cdot (\log m)^3 + 1 + (d+2)/2 - (\log m)^2 / (2(d+1))$$

$$\text{using } c = (\ln 2)/3 \text{ and } m \leq (d+1) \cdot 2^{d+1}$$

$$\leq c \cdot (\log m)^3 + 1 \quad \text{since } (d+1)(d+2) \leq (\log m)^2 \text{ for } m \leq 10.$$

This establishes the upper bound claimed.

For the lower bound, suppose C is a minimal length universal chain. We can assume that no element of C has length greater than 1, and for $i \geq 0$ we define k_i to be the number of elements of C with length in the range $[2^{-i}, 2^{-i+1})$. Let F_0 be the family of $m+1$ multisets $s_0, \dots, s_m$ where s_j has j of its elements at 0 and $m-j$ at 1. When using C to cover members of F_0 the only choices that can be made are at which end of the interval to start and then for each member of C whether to remain at the current end or move to the other, a total of $2^{(k_0+1)}$ possibilities. Since C is universal for F_0 we must have:

$$2^{(k_0+1)} \geq m+1.$$

This already gives a lower bound of $\Omega(\log m)$. Now we define F_t for all $t \geq 0$ to be the family of multisets of size m where each element is at one of the points $j \cdot 2^{-t}$ for $j = 0, \dots, 2^t$. In using C to cover members of F_t we have for each element of length less than 2^{-i+1} , a choice of at most $2^{t-i+2} - 1$ points to visit, i.e., at most $2^{t-i+1} - 1$ points on either side of the current point. Arguing as above that the total number of choices in using C for F_t must be at least as great as the cardinality of F_t , we obtain the following inequality:

$$(2^t + 1) \prod_{0 \leq i \leq t} (2^{t-i+2} - 1)^{k_i} \geq \binom{m+2^t}{2^t}.$$

Recognizing that $(T + 1)! < T^T$ for $T \geq 3$ and taking logarithms, we derive:

$$2^{-t} \sum_{0 \leq i \leq t} (t - i + 2)k_i \geq \log m - t$$

for $t \geq 2$. The same inequality may be verified explicitly for $t = 0$ and 1. Summing each side of these inequalities for $t = 0, \dots, \log m$, we may deduce:

$$(2 + 3 \cdot 2^{-1} + 4 \cdot 2^{-2} + 5 \cdot 2^{-3} + \dots) \sum_{0 \leq i \leq \log m} (2^{-i} k_i) \geq (\log m)^2 / 2.$$

The series on the left has a limit of 6, so we have a lower bound on the total length of C given by $(\log m)^2 / 12$. $\square$

For two dimensions our upper and lower bounds are closer.

THEOREM 2.

$$\Omega(\sqrt{m}) \leq u_2(m) \leq O(\sqrt{m}).$$

Proof. We shall show a bound of the form: $u_2(m) \leq a\sqrt{m} - 1$. If $m \leq a^2$ then a chain consisting only of unit links is universal (in the L_∞ metric). Assume therefore that $m > a^2$ and that there is a chain of length at most $a\sqrt{r} - 1$ for r points in the square where $r < m$. Let $n = \lceil \sqrt{m}/7 \rceil$ and $p = 1/n$. Divide the unit square into an $n \times n$ array of $p \times p$ squares. A **cluster** is defined to be any multiset of 10 points lying within one such subsquare. Identify a maximal number of disjoint clusters. The number of points not included within some cluster is at most $9n^2$. Select $q = \lceil (m - 9n^2)/10 \rceil$ clusters and consider the multiset Q consisting of the centers of the subsquares containing each cluster. Given a covering chain for Q we can modify this by increasing each link by an amount p and inserting 9 links of size p for each cluster. The resulting chain can cover all points in the clusters. Our universal chain is obtained by appending one further link of size 1 and then a universal chain for the remaining points. Thus

$$\begin{aligned} u_2(m) &\leq u_2(q) + p(q - 1) + 9pq + 1 + u_2(m - 10q) \\ &\leq a\sqrt{q} - 1 + p(q - 1) + 9pq + 1 + a\sqrt{m - 10q} - 1 \\ &< a\sqrt{q'} + 10q'/n + a\sqrt{m - 10q'} - 1 \end{aligned}$$

for $a \geq 10$, where $q' = (m - 9n^2)/10$. To show this result to be less than or equal to $a\sqrt{m} - 1$ as required, we shall verify that:

$$a \cdot (1 - \sqrt{(1 - 9v^2)/10} - 3v) \geq 1/v - 9v$$

where $v = n/\sqrt{m}$. We now select the value $a = 21$ and observe that since $\sqrt{m} > a$ the choice of n ensures that v lies in the interval $[1/7, 4/21]$. For this range of v the above inequality holds and the result is proved.

The lower bound is obvious from consideration of a set of m points which is distributed as a $\sqrt{m} \times \sqrt{m}$ square array with distance $1/\sqrt{m}$ between adjacent points. $\square$

Finally in this section we consider the natural extension to higher dimensions and derive bounds which tighten asymptotically as the dimension tends to infinity.

THEOREM 3. For all $d \geq 2$ there is a constant $c(d)$ such that

$$m^{1 - 1/d} \gtrsim u_d(m) \gtrsim c(d) \cdot m^{1 - 1/d}.$$

Furthermore $c(d) \rightarrow 1$ as $d \rightarrow \infty$.

Proof. The lower bound comes from consideration of an $m^{1/d} \times \dots \times m^{1/d}$ array of points with a spacing of $m^{-1/d}$.

For the upper bound we follow the pattern in the proof of Theorem 2, but now take **clusters** to be multisets of $k = \lceil b^d/2 \rceil$ points lying within one $b \cdot m^{-1/d} \times \dots \times b \cdot m^{-1/d}$

subhypercube in a regular $m^{1/d}/b \times \dots \times m^{1/d}/b$ array. The number excluded from a maximal set of clusters is at most $(k-1)m/b^d$ which is less than $m/2$, so for our construction we select $q = \lceil m/(2k) \rceil$ clusters. The resulting recurrence, following the pattern from Theorem 2, is:

$$u_d(m) \lesssim f(q) + kqb \cdot m^{-1/d} + u_d(m - qk).$$

The claimed upper bound is shown by the following inequalities:

$$\begin{aligned} c \cdot q^{1-1/d} + kqb \cdot m^{-1/d} + c \cdot (m - qk)^{1-1/d} &\lesssim m^{1-1/d} (c/b^{d-1} + b/2 + c/2^{1-1/d}) \\ &\lesssim c \cdot m^{1-1/d} \end{aligned}$$

which hold provided that

$$c = c(d) \geq b/(2 - 2/b^{d-1} - 2^{-1/d}).$$

Any large enough constant b is adequate here. For the claim that $c(d) \rightarrow 1$ we choose $b = b(d)$ such that $b \rightarrow 1$ but $b^{d-1} \rightarrow \infty$ as $d \rightarrow \infty$. $\square$

3. Applications. Although the above results are offered as being of purely combinatorial interest, the motivation for our research comes from the design of “semi-custom” integrated circuits. Here the main components of a layout are designed and positioned in advance leaving various wiring details and local choices to be made for particular applications. As an example of a fragment of such a design consider Fig. 1. On the circuit board shown in Fig. 1, there are contacts for n wires to enter from the left edge and for $n - m$ of these to leave at the right edge. Some m of these, which will be specified in the particular application, are to be connected in an arbitrary order to the m “vias” (connections between the front and back faces) which are shown as black squares. Furthermore no wires are allowed to touch or cross. Figure 2 shows the form of a simple **partial weave** layout method which can always be used if there is enough space between successive vias. The asterisks show which subset of the inputs is to terminate at vias in this instance. The wires that pass above a via run in a contiguous parallel group at the top of the rectangle, and similarly for those passing below. If two inputs separated by r inputs on the left edge are chosen to connect to adjacent vias then a spacing between these vias which allows r wires to pass through is certainly sufficient for all the required transfers

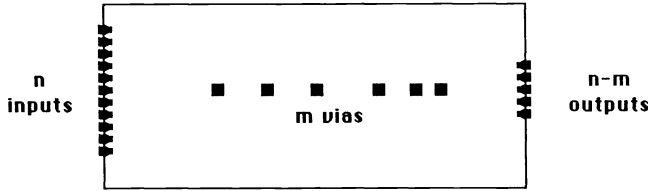


FIG. 1. Input/output board with drop-offs.

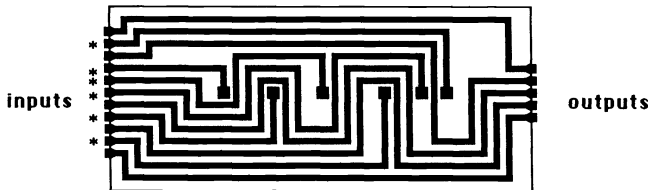


FIG. 2. Partial weave layout.

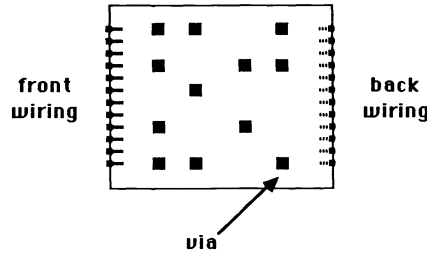


FIG. 3. Two-sided permutation board.

between top and bottom. Extra spaces of size n and $n - m$ are allowed at the left and right ends, respectively. As a final design stage the height of the layout is reduced as much as possible.

THEOREM 4. *The layout problem posed above can be solved within a rectangle as shown in Fig. 1 of height $O(n)$ and width $O(n(\log m)^3)$. If the vias are constrained to lie in a horizontal line, the layout requires height $\Omega(n)$ and width $\Omega((n - m)(\log m)^2)$.*

Proof. The spacing of vias used corresponds to a universal chain for sets of size m within an interval of size either n or $n - m$ for the upper or lower bound, respectively. $\square$

A **two-sided permutation board of order n** is a square with two faces described as follows. Along one edge of the front face are n input terminals labelled $1, \dots, n$, while along one edge of the back face are n output terminals labelled $1, \dots, n$. In addition there are n vias arranged in the interior of the square. See Fig. 3. The board is to have the property that, for any n -permutation, there is a planar layout on the front face of n disjoint wires from the inputs to the vias and a similar layout from vias to outputs on the back face, such that the connections from inputs to outputs realize the permutation.

THEOREM 5. *There is a two-sided permutation board of order n with area $O(n^{5/2})$.*

Proof. Consider a **weave layout** as shown in Fig. 4. This is just a special case of the partial weave layout used above, where *all* of the inputs are terminated at vias.

Labelling the vias $v_1, \dots, v_n$ from left to right, we see that it is enough for the horizontal spacing between vias to satisfy:

$$d(v_i, v_{i+1}) \geq \max \{ |\pi_i - \pi_{i+1}|, |\pi'_i - \pi'_{i+1}| \} \quad \text{for } i = 1, \dots, n-1$$

where via j is connected to input π_j and output π'_j for all j .

Represent some permutation to be realized on the permutation board as an $n \times n$ permutation matrix. Embed this matrix in a unit square and consider the set of n points defined by the 1's in the matrix. By Theorem 2 there is a fixed universal chain $a_1, \dots, a_{n-1}$ of length $O(\sqrt{n})$ which covers any such set, visiting it in some order. Using this order for the ones of the matrix let the corresponding order for the rows and the columns

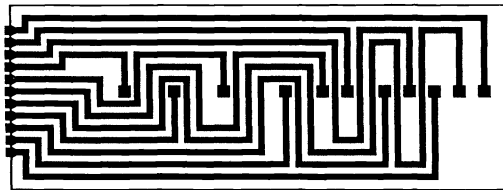
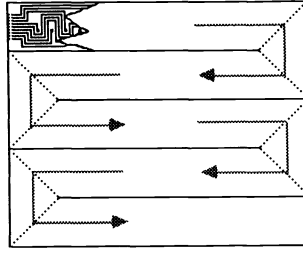


FIG. 4. Weave layout.

FIG. 5. *Folded weave layout.*

be $\pi_1, \dots, \pi_n$ and $\pi'_1, \dots, \pi'_n$, respectively. A weave layout with horizontal spacings $n \cdot a_1, \dots, n \cdot a_{n-1}$ can be made corresponding to the connection pattern given by the row order or the column order. A rectangular permutation board can therefore be made with the input and output terminals on the two faces of the left-hand edge and the vias arranged in a horizontal row with the spacings given above and will have height $O(n)$ and width $O(n^{3/2})$. This strip could then be “folded” as indicated in Fig. 5, and the terminals set in whichever edges are desired within a final square permutation board of area $O(n^{5/2})$. This folding technique is described in [4]. $\square$

4. Related work and conclusions. Cutler and Shiloach [2] showed that $O(n^3)$ area was sufficient for the layout of all permutations between n input and n output terminals in fixed locations. Their construction is like a single weave layout as in Fig. 4, but with the vias replaced by the output terminals spaced regularly with an interval of $O(n)$. In [1] a matching lower bound of $\Omega(n^3)$ is stated for this problem. This $\Omega(n^3)$ lower bound result has been extended [3] to permit the use of both sides of a circuit board and cn fixed vias, where $c < 1$. Our Theorem 5 shows that the area can be reduced to $O(n^{5/2})$ when $c = 1$. Of course if the locations of the vias are not fixed then a straightforward crossbar layout achieves an area of $O(n^2)$ using n vias.

It is an open problem to improve the bound in Theorem 5, though the lower bound from Theorem 2 suggests that this may be optimal. For the unit interval, the upper and lower bounds given in Theorem 1 leave in question the true value of u_1 .

REFERENCES

- [1] A. AGGARWAL, M. KLAWE, D. LICHTENSTEIN, N. LINIAL AND A. WIGDERSON, *Layer changes in VLSI*, Proc. 26th Annual IEEE Symposium on Foundations of Computer Science, 1985, pp. 186–196.
- [2] M. CUTLER AND Y. SHILOACH, *Permutation layout*, Networks, 8 (1978), pp. 253–278.
- [3] M. KLAWE, personal communication.
- [4] C. E. LEISERSON, *Area efficient graph layouts (for VLSI)*, Proc. 21st Annual IEEE Symposium on Foundations of Computer Science, 1980, pp. 270–281.

HOLE PROBLEMS FOR RECTANGLES IN THE PLANE*

GREGORY J. E. RAWLINS†, PETER WIDMAYER‡, AND DERICK WOOD§

Abstract. Given a set of n rectangles with sides parallel to the coordinate axes, we show how to determine whether their union, viewed as a set of disjoint polygons, has a hole. The algorithm presented needs no more than $O(n \log n)$ time and $O(n)$ space, which is shown to be optimal. However, in practice it is also necessary to know the locations of holes, if there are any. We present an algorithm to determine the locations of all h holes in $O(n \log n + h)$ time and $O(n)$ space, which is again optimal. The algorithm computes a point within each hole, representing the location of the hole. The efficiency of these and several other algorithms follows from some simple combinatorial arguments about sets of rectangles in the plane.

Key words. rectangles, holes, counting, reporting

AMS(MOS) subject classifications. 51-04, 68E99, 05B99

1. Introduction. The problem of gathering information about holes in the union of a set of rectangles with sides parallel to the coordinate axes is of theoretical interest and of importance as a step in the computation of the contour of Boolean combinations of sets of rectangles. Certainly, any algorithm reporting the contour of the union of a set of rectangles (see, e.g., [3], [2], [9]) will also report the contours of all holes. However, such an algorithm uses time $O(n \log n + p)$ for n rectangles and p edges in the contour, where $p = O(n^2)$ in the worst case. For some questions concerning holes, this may be too much information to be desirable. Specifically, we answer the following simpler problems more efficiently.

Given a set of n rectangles in the plane, with sides parallel to the coordinate axes,

- (1) Does the union of the rectangles have a hole?
- (2) Report one point in each hole of the union of the rectangles.

In order to solve problems (1) and (2) efficiently, we establish a few simple combinatorial properties of a set of rectangles in § 2. We would certainly like problem (1) to be decidable in time independent of the number of edges of the holes or even independent of the number of holes. We show in § 3 that $\Omega(n \log n)$ is a lower bound for problem (1), and we present an algorithm answering problem (1) in time $O(n \log n)$.

For problem (2), the number h of holes is the size of the output. From the lower bound of problem (1), a lower bound of $\Omega(n \log n + h)$ follows directly for problem (2). An algorithm to answer problem (2) in time $O(n \log n + h)$ is presented in § 4. Finally, we present some final remarks in § 5.

2. Basic definitions and combinatorial properties. We are given a set $\mathcal{R} = \{R_1, R_2, \dots, R_n\}$ of n rectangles in the plane. Each rectangle R_i , $1 \leq i \leq n$, is described by a quadruple $(x_l, x_r, y_b, y_t)_i$ of coordinates for its left x -, right x -, bottom y -, and top y -coordinate. We identify rectangle R_i with the two-dimensional interval

* Received by the editors November 10, 1986; accepted for publication June 8, 1987.

† Department of Computer Science, Indiana University, Bloomington, Indiana 47405. The work of this author was supported by the Natural Sciences and Engineering Research Council of Canada under grant A-5692.

‡ Institut für Angewandte Informatik und Formale Beschreibungsverfahren, Universität Karlsruhe, Postfach 6980, 7500 Karlsruhe, Federal Republic of Germany. The work of this author was supported by a grant from the Deutsche Forschungsgemeinschaft Wi 810/1-1.

§ Data Structuring Group, Department of Computer Science, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1. The work of this author was supported by the Natural Sciences and Engineering Research Council of Canada under grant A-5692.

$[x_l, x_r] \times [y_b, y_t]$ in the plane. A point $p = (x, y)$ in the plane is said to be *covered* by rectangle R_i if $x_l \leq x \leq x_r$ and $y_b \leq y \leq y_t$ for $R_i = (x_l, x_r, y_b, y_t)$; rectangle R_i is said to *cover* p . Let $\mathcal{U}$ denote the union of all points covered by some rectangle in $\mathcal{R}$. Without loss of generality, assume that $\mathcal{U}$ is a connected set (the connected components of $\mathcal{R}$ can be computed in time $O(n \log n)$; see [1]). If there are several connected components, one can lie inside a hole of another; because we are interested in holes with respect to the complexity of computations, we consider all components separately. $\mathcal{U}$ can be described as a polygon P with sides parallel to the coordinate axes. P divides the plane $\mathbb{R}^2$ into a number of connected components of maximal size. That is,

$$\mathbb{R}^2 - P = \{P_i | 0 \leq i \leq h\},$$

for $h \geq 0$, and each P_i is a polygon describing a connected set of points not covered by any rectangle in $\mathcal{R}$, and no two polygons P_i, P_j intersect for $i \neq j$. P_0 is defined to be the polygon containing point (∞, ∞) ; it is called the *exterior* of $\mathcal{U}$. Polygons $P_1, \dots, P_h$ are called the *holes* of $\mathcal{U}$.

For an example illustrating the definitions, see Fig. 1. Note that there need not be any holes in $\mathcal{U}$. To simplify the discussion, but with no loss of generality, we assume that the rectangles are in *general position*, that is, no two of the x -values (y -values) in the input data are the same.

Recall that n is the number of rectangles and h is the number of holes in $\mathcal{U}$. Let c , or equivalently, h_0 , be the number of edges of the exterior P_0 of $\mathcal{U}$, and let h_i be the number of edges of hole P_i , $1 \leq i \leq h$. Then we know the following from [3].

PROPOSITION 2.1 [3]. *If $\mathcal{U}$ has no holes, then $c \leq 8n - 4$.*

Widmayer and Wood [8] have generalized Proposition 2.1 to allow holes.

PROPOSITION 2.2 [8]. *$c \leq 8n - 4$, for n and c as defined above.*

Proof. We present a proof of this proposition that is different from the one in [8], with the aim of using the insights it provides in subsequent proofs.

Any set of $n + 1$ rectangles in the plane in one connected component can be obtained by adding a rectangle to a set of n rectangles in a connected component, for $n \geq 1$. For $n = 1$, the claim holds, because $c = 4$. Now suppose our claim holds for some $n \geq 1$. Consider a set of $n + 1$ rectangles. Remove a rectangle R_{n+1} . Then the intersection of the boundary of R_{n+1} with the uncovered area $\mathbb{R}^2 - P$ consists of line segments on the

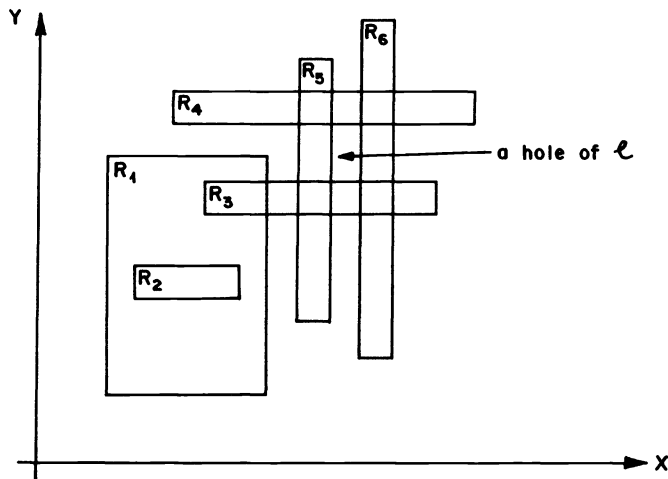


FIG. 1. *The basic situation.*

boundary of R_{n+1} (see Fig. 2). At most eight line segments contain a vertex of R_{n+1} ; these line segments may increase the number of edges of either the exterior P_0 or of holes (see Fig. 2, edges e_1 to e_5). Consider a line segment on the boundary of R_{n+1} obtained from the intersection with P_0 , say line segment l_0 . Then, either l_0 is an edge of the exterior P'_0 of $\mathcal{U}' = \mathcal{U} \cup R_{n+1}$, or l_0 is an edge of a newly created hole $P_j, j > h$ (see Fig. 2, edges e_6 to e_{12}). In the former case, the number of edges of P_0 does not increase, because l_0 replaces at least one edge in P_0 ; that is, at least one edge e_0 of P_0 is not an edge of P'_0 , because l_0 belongs to P'_0 , and e_0 either lies in $\mathcal{U}'$ or belongs to a hole in $\mathcal{U}'$. Therefore, c' , the number of edges of P'_0 , satisfies $c' \leq c + 8 \leq 8n - 4 + 8 = 8(n + 1) - 4$. $\square$

COROLLARY 2.3. For any hole P_i of $\mathcal{U}$, $h_i = 0$ for $n \leq 3$, $h_i \leq 4$ for $n = 4$, and $h_i \leq 8n - 32$ for $n \geq 5$.

Proof. The proof is shown by arguments similar to those for Proposition 2.2. $\square$

Note that all of the stated bounds are tight, that is, there exist examples for arbitrary n achieving the stated bounds exactly. For Proposition 2.2, the example can be constructed as in [3] (Fig. 3). For Corollary 2.3, Fig. 4 provides an example. We have seen so far that the exterior of $\mathcal{U}$, P_0 , as well as any hole P_i , $1 \leq i \leq h$, of $\mathcal{U}$, may have a number of edges that is linear in n . The example in Fig. 5 shows that the number of holes may be

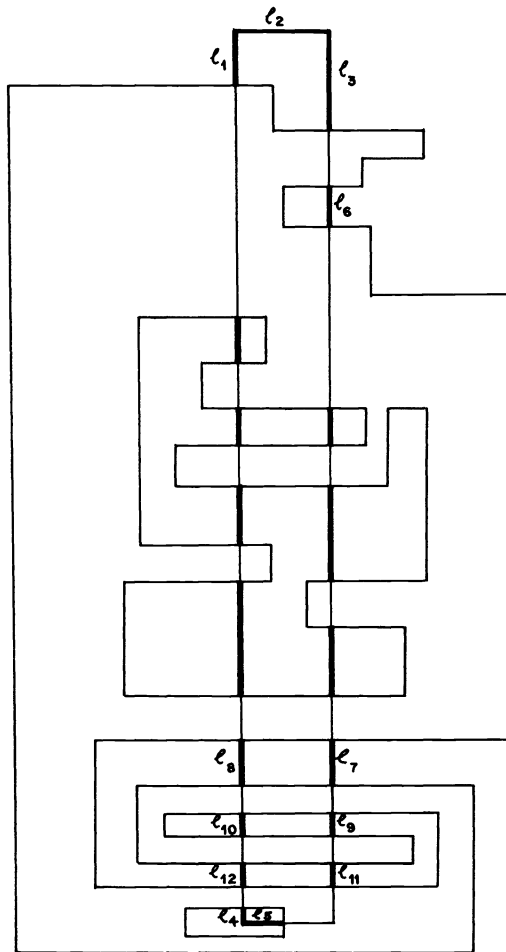


FIG. 2. Illustrating the proof of Proposition 2.2.

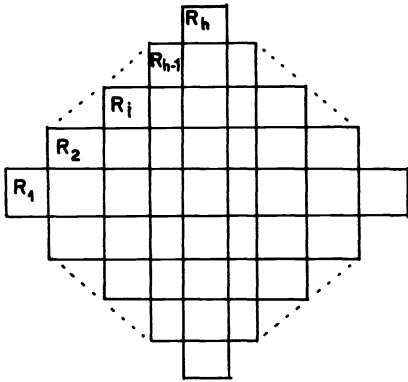


FIG. 3. The bound in Proposition 2.2 is tight.

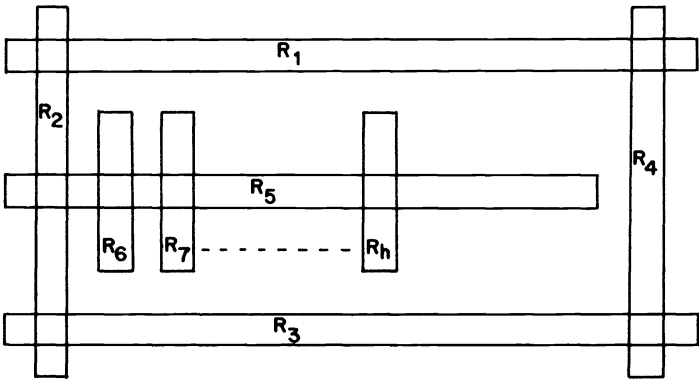


FIG. 4. The bounds in Corollary 2.3 are tight.

as high as $(n/2 - 1)^2$. However, the total number of edges in all holes and the exterior of $\mathcal{U}$ together, is much less than the combination of these numbers might suggest.

LEMMA 2.4. $e \leq 8n + 4h - 4$, where $e = \sum_{i=0}^h h_i$ is the total number of edges in the exterior and in all holes of $\mathcal{U}$.

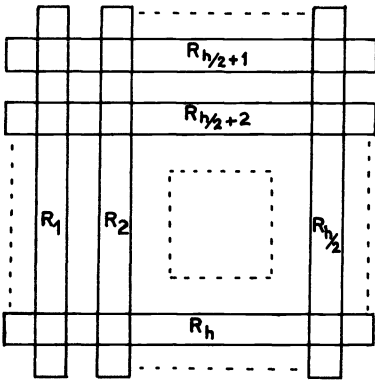


FIG. 5. The maximum number of holes.



FIG. 6. Splitting a hole (before and after).

Proof. Based on the proof of Proposition 2.2, consider the case where l_0 is not an edge of P_0 , but an edge of a hole P_j , $j > h$. Then P_j may have just been created, that is, before adding R_{n+1} , P_j was part of P_0 . In this case, l_0 accounts for one new hole. Another possibility is that P_j has been obtained by splitting a hole in two. In general, i edges on the boundary of R_{n+1} may create i new holes from an old one, that is, add $i - 1$ new holes. For $i = 2$, the increase in the number of edges by creating a new hole is at most 4 (see Fig. 6). For $i > 2$, the increase in the number of edges per new hole is less. Therefore, we get $e' \leq e + 8 + 4h'$, where h' is the additional number of holes created by R_{n+1} . Altogether, $e' \leq e + 8 + 4h' \leq 8n + 4h - 4 + 8 + 4h' = 8(n + 1) + 4(h + h') - 4$. $\square$

Next, we show that this bound is tight; that is, for any given n and h such that h holes can be formed by a set of n rectangles, there is a set of n rectangles forming h holes and having a total of e edges in the exterior and in all holes of $\mathcal{U}$. To this end, we first need to know the maximum possible number h of holes for a given n .

LEMMA 2.5. *The number of holes in any arrangement of n rectangles is at most $(\lceil n/2 \rceil - 1)(\lfloor n/2 \rfloor - 1)$, for $n \geq 1$.*

Proof. The proof is by induction on n . Let $h(n)$ denote the number of holes in an arrangement of n rectangles. We perform induction on odd and even n separately.

Consider a hole with leftmost left boundary, and a hole with topmost top boundary. Any rectangle bounding the former hole to the left is called a leftmost rectangle; any rectangle bounding the latter hole from the top is called a topmost rectangle. Let $d(n)$ denote the decrease in the number of holes of an arrangement of n rectangles, when a leftmost and a topmost rectangle are removed. That is, $d(n)$ is at most the number of holes with a left or a top hole edge from one of the two rectangles being removed. The edges of these holes are, hence, aligned along the two rectangles, in a purely linear fashion (see Fig. 7). Therefore, to separate two holes from each other, a rectangle is needed. This rectangle cannot separate any other two of the holes under consideration. As there are just $n - 2$ rectangles left, these can form at most $n - 3$ holes in the described way. Therefore, $d(n) \leq n - 3$. Now consider induction on even numbers of rectangles, that

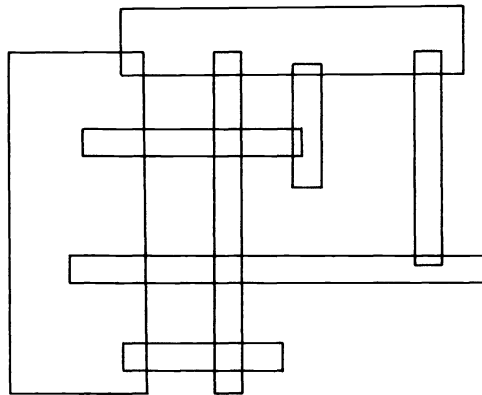


FIG. 7. Leftmost and topmost holes and rectangles.

is, $h(2n)$, for $n \geq 2$. The claim $h(2n) \leq (n-1)^2$ holds for $2n = 4$. Assume the claim holds for all even values up to some value $2n \geq 4$. We show that the claim then holds for $2(n+1)$. To this end, assume the contrary, that is, $h(2(n+1)) > n^2$. Because $d(2(n+1)) \leq 2n-1$ is the number of holes we have to subtract when removing two rectangles, we get $h(2n) \geq h(2(n+1)) - d(2(n+1)) > n^2 - 2n + 1 = (n-1)^2$, a contradiction to the inductive assumption that $h(2n) \leq (n-1)^2$.

For odd numbers of rectangles, the argument is similar. The claim $h(2n+1) \leq n(n-1)$ holds for $2n+1 = 5$ (we could also start with $2n+1 = 3$ and no holes). Assume the claim holds for all odd values up to some $2n+1 \geq 5$. To see that the claim also holds for $2(n+1)+1$, assume the contrary, that is, assume that $h(2(n+1)+1) > (n+1)n$. By removing the two rectangles to the left and on the top we get $h(2n+1) \geq h(2(n+1)+1) - d(2(n+1)+1) > (n+1)n - 2n = n(n-1)$, a contradiction. $\square$

The example in Fig. 5 shows how the upper bound given in Lemma 2.5 can be achieved for even n ; for odd n , just add one rectangle in either of the two ways. Any desired number h of holes can be achieved by shrinking rectangles R_i , $1 \leq i \leq \lfloor n/2 \rfloor$, in such a way that enough holes are connected to the exterior and the desired number of holes remains (see Fig. 8 for an example with six rectangles). If the corner points of all rectangles belong to the boundary of a hole or the exterior, that is, are not covered by

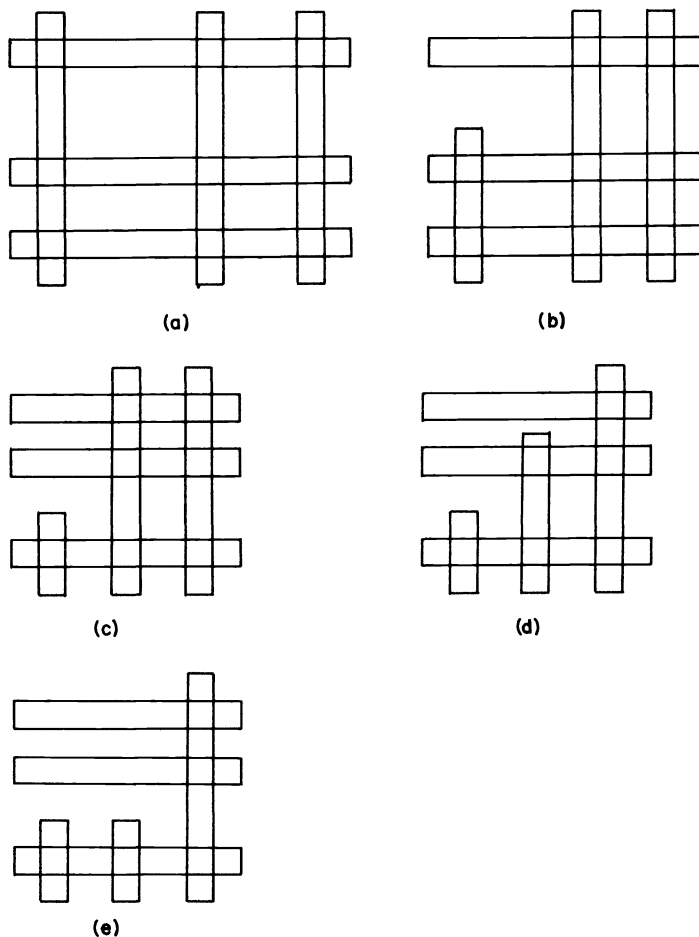


FIG. 8. Achieving all possible numbers of holes by shrinking.

some other rectangle, then the bound of Lemma 2.4 is realized exactly. Hence, for any n and h as described, this bound is tight.

For $k \geq 1$ connected components of the set $\mathcal{R}$ we obtain the following corollary.

COROLLARY 2.6. *For all $n \geq 1$ and $k \geq 1$:*

- (1) $c \leq 8n - 4k$;
- (2) $h_i = 0$ for $n - k + 1 \leq 3$,
 $h_i \leq 4$ for $n - k + 1 = 4$,
 $h_i \leq 8n - 8k - 24$ for $n - k + 1 \geq 5$;
- (3) $e \leq 8n + 4h - 4k$.

Proof. For k components having $n_j \geq 1$ rectangles in component j , $1 \leq j \leq k$, we observe the following:

$$(1) \ c \leq \sum_{j=1}^k (8n_j - 4) = 8 \sum_{j=1}^k n_j - 4k = 8n - 4k.$$

(2) The number of holes is maximized if all components apart from one consist of only one rectangle, that is, without loss of generality, $n_1 = n - k + 1$, $n_2 = n_3 = \dots = n_k = 1$. Substituting $n - k + 1$ in the bound of Corollary 2.3 yields the above bound.

(3) Let $h(j)$ be the number of holes in component j , $1 \leq j \leq k$; then $e \leq \sum_{j=1}^k (8n_j - 4h(j) - 4) = 8 \sum_{j=1}^k n_j - 4 \sum_{j=1}^k h(j) - 4k$. Hence $e \leq 8n - 4h - 4k$. $\square$

Finally, to ensure that our algorithm satisfies the claimed bounds on time and space, we need the following:

LEMMA 2.7. *For any vertical or horizontal line l , let $I \subseteq \{0, \dots, h\}$ be the set of indices of those polygons P_i (holes and exterior of $\mathcal{U}$) that intersect l , that is, $I = \{i | P_i \cap l \neq \emptyset\}$. Then $\sum_{i \in I} h_i \leq 16n - 4k$.*

Proof. In the given set $\mathcal{R}$ of n rectangles, line l intersects all rectangles in a subset $\mathcal{R}' \subseteq \mathcal{R}$. Let each rectangle in $\mathcal{R}'$ be cut into two pieces at l . Then we get two sets of rectangles: $\mathcal{R}_l$ to the left of l and $\mathcal{R}_r$ to the right of l . All edges of the exterior and of holes of $\mathcal{R}$ that have been intersected by l belong (partially) to the exterior of at least one of $\mathcal{R}_l$, $\mathcal{R}_r$. Let k, c, n with subscript l (r) be defined with respect to set $\mathcal{R}_l$ ($\mathcal{R}_r$). Then $k_l + k_r \geq k$, and $n_l + n_r \leq 2n$. By Corollary 2.6 (2.3) we get

$$\sum_{i \in I} h_i \leq c_l + c_r \leq 8n_l - 4k_l + 8n_r - 4k_r \leq 16n - 4k. \quad \square$$

Note that this bound is not tight in general, but it is sufficient to establish the runtime and space consumption arguments for the algorithms to be presented. The essence of Lemma 2.7 has been stated and proved already in [8]; we include it here for the sake of completeness.

3. The hole detection problem. Let us present a lower bound for the hole detection problem first and then describe an optimal solution.

3.1. A lower bound.

LEMMA 3.1. *Given a set $\mathcal{R}$ of n rectangles in the plane, $\Omega(n \log n)$ time is required to determine whether their union $\mathcal{U}$ has a hole. This holds even if the rectangles are given in sorted order according to one of the coordinates.*

Proof. The proof is shown by a reduction from the element uniqueness problem for positive integers.

Given a set $X = \{x_1, \dots, x_n\}$ of positive integers, define a set of $3n + 1$ rectangles in the plane as follows. Compute $x_{\max} = \max \{x_i | 1 \leq i \leq n\}$ in time $O(n)$ by a simple linear scan. Then $\mathcal{R} = \{R_0, \dots, R_{3n}\}$, where

$$R_0 = (-\frac{1}{4}, \frac{1}{4}, -\frac{1}{4}, x_{\max} + \frac{1}{4}),$$

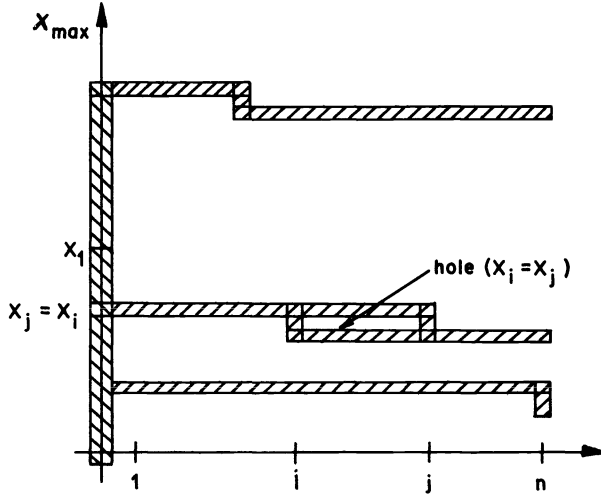


FIG. 9. A lower bound for hole detection.

and for $i = 1, \dots, n$,

$$R_{3i-2} = (-\frac{1}{4}, i + \frac{1}{4}, x_i, x_i + \frac{1}{4}),$$

$$R_{3i-1} = (i - \frac{1}{4}, i + \frac{1}{4}, x_i - \frac{1}{2}, x_i + \frac{1}{4}), \quad \text{and}$$

$$R_{3i} = (i - \frac{1}{4}, n + \frac{1}{4}, x_i - \frac{1}{2}, x_i - \frac{1}{4}).$$

For an illustration see Fig. 9. Note that we have deliberately violated the assumption that the rectangles are in general position, in order not to obscure the basic idea of the reduction. The reduction can be changed slightly to shift the rectangles into general position. The set of rectangles has the following properties: All rectangles form one connected component; there is a hole in $\mathcal{U}$ if and only if the elements of X are not unique.

It is immediate from the construction that Lemma 3.1 holds even if the rectangles are given in sorted order according to one of the coordinates.

This completes the proof of Lemma 3.1. $\square$

3.2. An algorithm for hole detection. Our general approach is to sweep the plane from left to right with a vertical sweep line l . The line stops at each left and right vertical edge of a rectangle. A data structure L keeps track of the horizontal edges of rectangles currently intersected by the sweep line; the *active* edges. Some of these edges separate a covered area from an area that is not covered, that is, they are on the boundary of $\mathcal{U}$. Whenever l intersects at least one rectangle there will be at least two edges that partially belong to the outside boundary of $\mathcal{U}$, that is, P_0 , and there may be edges that partially belong to holes P_i , $i > 0$. All of these edges are said to be *visible*. In addition, there may be active edges that do not currently belong to the boundary of $\mathcal{U}$; these are said to be *hidden*. Of the visible edges, some are known to belong to the outside boundary of $\mathcal{U}$, namely, at least the topmost and the bottommost ones; these edges are called *outer* edges. There may be other visible active edges that may or may not belong to a hole, as far as this can be judged from the current state of the plane sweep; these edges are called *tentative inner* edges with respect to the state of the plane sweep.

For an illustration of these concepts, see Fig. 10. At sweep position x_1 , e_1 , e_2 , e_3 and e_4 are all tentative inner edges, whereas e_5 and e_6 are outer edges. At sweep position

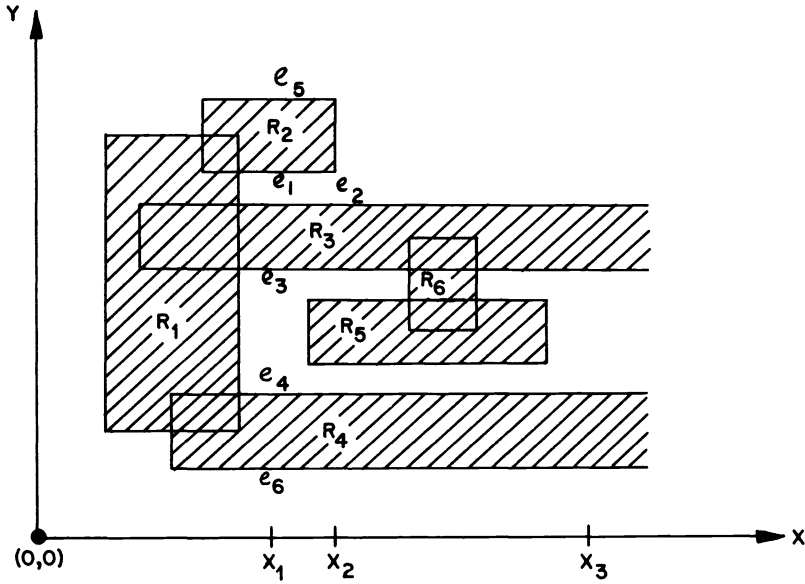


FIG. 10. Hole detection.

x_2 , e_1 becomes an outer edge, because it can be determined that e_1 will not contribute to a hole. At sweep position x_3 , it is still not known whether e_4 will contribute to a hole.

An uncovered area between two tentative inner edges is called a *tentative hole*. Any hole has to start in our plane sweep as a tentative hole. A tentative hole splits into several branches, when a left vertical edge of a rectangle falls entirely in between the two active edges of the hole. In our example, R_5 's left vertical edge falls entirely in between e_3 and e_4 . At a right vertical edge of a rectangle R_i , two areas meet that have been separated currently by R_i . If two tentative holes meet, it is recognized that they actually form one tentative hole. If a tentative hole meets the outside P_0 of $\mathcal{U}$, it is recognized that the tentative hole belongs to P_0 rather than forming a hole. A hole is detected only when a left vertical edge blocks all branches of a tentative hole.

Note that the tentative inner edges come in pairs of active edges with adjacent intersections with the sweep line. All of the above situations and transitions can be properly determined by keeping track of all pairs of tentative inner edges, of outer edges, and of hidden edges, in the way outlined above. Furthermore, in order to decide whether all branches of a tentative hole have been blocked, it is helpful to keep track of a set of branches for each tentative hole. Each tentative hole will start out with one branch. Later on, a branch may be added, whenever the tentative hole splits; a branch may be deleted when it ends; two tentative holes may be combined into one when two of their tentative branches meet; and, finally, a tentative hole is recognized to be a hole, when all of its branches end.

Then, more formally, the algorithm works as follows.

ALGORITHM DETECT HOLE.

{ Given a set $\mathcal{R}$ of rectangles, detect whether there is a hole in the contour of their union. }

1. Initialize all sweep-line data structures to be empty, and sort all x -values (vertical edges).

2. For all vertical edges e in ascending x -order do:
 - if e is a left vertical edge, then
 - terminate all branches of tentative holes in the y -range of e ;
 - if the set of branches for some tentative hole becomes empty, then report that a hole has been detected, and stop;
 - update all (at most 2) branches that are intersected by (but not contained in) e 's y -range;
 - if e falls within one branch of a tentative hole, then split this branch into two;
 - terminate all contour edges that lie within e 's y -range;
 - add the top and bottom edges of e 's rectangle to the set of active edges, and to the contour if they lie outside the area covered by the rectangles so far;
 - else { e is a right vertical edge }
 - if there are no edges in e 's y -range, to the right of e , that belong to the contour, then
 - combine the two areas adjacent to e above and below into one, by creating a new tentative hole from two old ones, by combining all branches, or by adding all branches of a tentative hole to the contour,
 - else { there are contour or hole edges to the right of e }
 - update the contour or holes above and below e accordingly;
 - start a new hole for every pair of horizontal edges in e 's y -range that have uncovered area between them.
 3. Report that there is no hole.
- end

3.3. An efficient implementation. In essence, to detect the existence of a hole in the union of a set of rectangles, the algorithm computes the contour implicitly and additionally keeps track of “topological” information about candidate holes. An optimal sweep-line algorithm for computing the contour (see [8]) uses as the central data structure the visibility tree (see [9]).

For any given sweep-line position, the active horizontal edges are represented by points on the sweep line. Some of these points are visible, others may be hidden. When a vertical edge is encountered, points may change their status from visible to hidden (at a left edge) or from hidden to visible (at a right edge). Also, points representing horizontal edges have to be inserted into or deleted from the visibility tree. The visibility tree allows for the insertion or deletion of a point representing a horizontal edge in time $O(\log n)$, where n is the number of points to be stored in the tree. For a given interval representing a vertical edge of a rectangle, the visibility tree enables the visibility status of all affected edges to be updated in time $O(\log n)$, and it enables these edges to be reported in constant time per edge. Furthermore, it uses only linear space, because each active point is stored only once.

The central data structure in our implementation of the hole detection algorithm is the visibility tree. During the line sweep, we maintain a visibility tree, together with information on the holes to which the visible points belong. To this end, we store the set of active visible edges belonging to a tentative hole in a doubly linked circular list, with one list item for each edge. This is done for each hole, and also for the outer contour. From each active visible point in the visibility tree there is a pointer to the corresponding list item. All updates of the lists are performed by following pointers from the visibility tree to list items. Actions need only be taken when edges become visible or become hidden. A starting hole or the starting contour lead to a new list containing two items,

one for each horizontal edge bounding the hole or contour. When a branch of a hole ends, the two corresponding items are removed from their list. Splitting a hole into two implies that two edges are added to the corresponding list; forming the union of two holes is realized by linking two circular lists (giving one circular list). Note that access to the circular lists is provided through the visibility tree by pointers to list items. As soon as a list becomes empty for a tentative hole and it was not the only one in existence, a hole has been detected. With this implementation, we get the following theorem.

THEOREM 3.2. *Given a set of n rectangles in the plane, it can be determined, in $O(n \log n)$ time and with $O(n)$ space, whether their union has a hole.*

Proof. The presented algorithm correctly determines whether there is a hole by sweeping a line from left to right until the right boundary of a hole is first encountered. Therefore, no hole lies entirely to the left of the current sweep-line position. With Lemma 2.7, the number of edges of all started holes and the contour is therefore $O(n)$. Hence, $O(n)$ operations have been performed on the visibility tree, each at a cost of $O(\log n)$ time. Altogether, $O(n)$ edge pieces have changed their visibility status, with each of the involved updates of the associated lists performed in constant time. This establishes the claimed time bound. The space requirement follows from that of the visibility tree. $\square$

4. Computing all holes. A lower bound for computing some constant size information about each hole, say, a point within each hole, follows immediately from the lower bound for hole detection.

COROLLARY 4.1. *Given a set of n rectangles in the plane, $\Omega(n \log n + h)$ time is required to compute a point within each hole in their union, where h is the number of holes.*

Proof. The proof follows from Lemma 3.1 and the fact that h is the number of points to be reported. $\square$

An algorithm to report a point inside each hole can be derived simply from the algorithm presented in § 3 to detect the existence of a hole. Instead of halting after the end of a hole has been detected, we report a point just to the left of the edge terminating a hole, and continue doing so for each hole until all edges have been processed. Thus, we obtain the following theorem.

THEOREM 4.2. *Given a set of n rectangles in the plane, a point within each hole in their union can be computed in time $O(n \log n + h)$ and space $O(n)$, where h is the number of holes.*

Proof. The number of operations to be carried out on the visibility tree is $O(n)$, for the $2n$ left and right vertical edges of rectangles, at a cost of $O(\log n)$ time each. For every change in the visibility status of (a piece of) an edge, constant additional cost to update the lists suffices. From Lemma 2.4 we know that there are $O(n + h)$ of these changes, establishing the claimed time bound. The space bound follows from that of the visibility tree. $\square$

5. Concluding remarks. In this paper we have proved some basic combinatorial results about the number of holes in the union of a set of rectangles. We have also provided time- and space-optimal algorithms to decide whether there is a hole and to compute a point within each hole.

Clearly, these questions can be posed in other frameworks, for example, what happens in d -dimensions, what happens with other shapes (for example, squares and circles), and what about the searching problem—given n rectangles and a query point that lies in a hole, provide the hole's contour. These questions and related ones are currently under investigation; some initial results are reported in [5].

Initially, the reason we investigated holes was to provide an alternative algorithm to solve the contour problem for a set of restricted-oriented polygons (see [6] for an introduction to restricted-oriented geometry). Of course, using the general algorithm of [4], we can solve this problem in $O(n \log n + k \log n)$ time, where the polygons have a total of n edges and there are k edge intersections. Because restricted-oriented polygons appear to be closer in their behavior to orthogonal polygons, we expected to obtain an $O(n \log n + p)$ time algorithm, where p is the number of edges in their union. Unfortunately, so far we have been unable to find such an algorithm. For this reason, we considered a different approach stemming from the following observation.

Given a point in a hole, we can extend it vertically upwards until it first hits an edge. This must contain an edge of the contour of the hole. Follow the edge to the right, say, until a second edge is met. This must also belong to the contour. We continue walking around the edges bordering the hole, in this manner, until we return to the starting edge.

To implement this algorithm for orthogonal polygons we can use a simplified form of the layered segment tree of [7]. Each search for an edge requires $O(\log n)$ time; therefore, this leads to an $O(n \log n + p \log n)$ time algorithm for the contour. This is worse than the best algorithms known for this problem (see [2], [9]) since they only require $O(n \log n + p)$ time. However, this technique can be applied to the restricted-oriented case, with the same time bounds, and, in this case, it is better than the general algorithm. At present, however, we are unable to find points in the holes within the same time bound so a contour algorithm for restricted-oriented polygons with this time complexity is still elusive.

REFERENCES

- [1] H. EDELSBRUNNER, J. VAN LEEUWEN, TH. OTTMANN, AND D. WOOD, *Computing the connected components of simple rectilinear geometrical objects in d -space*, RAIRO Inform. Théor., 18 (1984), pp. 171–183.
- [2] R. H. GÜTING, *An optimal contour algorithm for iso-oriented rectangles*, J. Algorithms, 5 (1984), pp. 303–326.
- [3] W. LIPSKI AND F. P. PREPARATA, *Finding the contour of the union of a set of iso-oriented rectangles*, J. Algorithms, 1 (1980), pp. 235–246.
- [4] TH. OTTMANN, P. WIDMAYER, AND D. WOOD, *A fast algorithm for the Boolean masking problem*, Comput. Vision, Graphics, and Image Process., 30 (1985), pp. 249–268.
- [5] G. J. E. RAWLINS, P. WIDMAYER, AND D. WOOD, *Circles, squares, and holes*, Technical Report, Department of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 1987.
- [6] G. J. E. RAWLINS AND D. WOOD, *Restricted-orientation convexity*, Technical Report, Department of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 1986.
- [7] V. K. VAISHANAVI AND D. WOOD, *Rectilinear line segment intersection, layered segment trees, and dynamization*, J. Algorithms, 3 (1982), pp. 160–176.
- [8] P. WIDMAYER AND D. WOOD, *Time- and space-optimal contour computation for a set of rectangles*, Inform. Process. Lett., 24 (1987), pp. 335–338.
- [9] D. WOOD, *The contour problem for rectilinear polygons*, Inform. Process. Lett., 19 (1984), pp. 229–236.

ON EXISTENCE OF t -DESIGNS WITH LARGE v AND λ^*

D. K. RAY-CHAUDHURI† AND N. M. SINGHI‡

Abstract. It is shown that for v sufficiently large and $k \geq 2t + 1$, for any feasible quadruple $t - (v, k, \lambda)$ there exists a $t - (v, k, \lambda)$ -design in which multiplicity of every block is 0 or ± 1 and the number of blocks with nonzero multiplicity is not too large compared to $\lambda \binom{v-t}{k-t}$. As a consequence it is shown that the usual $t - (v, k, \lambda)$ -designs in which no block is repeated more than twice exist if

$$\binom{v-t}{k-t} + c_1(t, k)v^{k-2t} \geq \lambda \geq \binom{v-t}{k-t} - c_1(t, k)v^{k-2t}$$

where $c_1(t, k)$ is some function of t and k only. This implies that in Wilson's result on the existence of a $t - (v, k, \lambda)$ -design for

$$\lambda = m \binom{v-t}{k-t} + \mu, \quad 0 \leq \mu < \binom{v-t}{k-t},$$

and m sufficiently large, the condition sufficiently large m can be replaced by $m \geq 0$ when $\mu \geq \binom{v-t}{k-t} - c_1(t, k)v^{k-2t}$ and by $m \geq 1$ when $\mu \leq c_1(t, k)v^{k-2t}$.

Key word. t -designs

AMS(MOS) subject classification. 05

1. Introduction. Throughout this paper we will assume that X is a finite set of size v . $P_k(X)$ denotes the set of all k -subsets of X , and $\mathbb{Z}$ denotes the ring of integers. Let $V_k = V_k(X)$ be the free $\mathbb{Z}$ -module of all functions $f: P_k(X) \rightarrow \mathbb{Z}$. For $f \in V_k$ we define

$$|f| = \sum_{B \in P_k(X)} |f(B)| \quad \text{and} \quad \text{supp } f = \{B \in P_k(X) | f(B) \neq 0\}.$$

An $f \in V_k$ is said to be a *signed* $t - (v, k, \lambda)$ -design if for every t -subset T of X

$$\sum_{T \subseteq B} f(B) = \lambda.$$

For $B \in P_k(X)$, $f(B)$ is called *multiplicity* of B in the design. A $t - (v, k, \lambda)$ -design is a signed $t - (v, k, \lambda)$ -design in which $f(B) \geq 0$ for all $B \in P_k(X)$. A signed $t - (v, k, \lambda)$ -design with $\lambda = 0$ is said to be a *null t -design*. A signed $t - (v, k, \lambda)$ -design is said to be *elementary* if all multiplicities are 0 or ± 1 . Clearly elementary $t - (v, k, \lambda)$ -designs are precisely $t - (v, k, \lambda)$ -designs with no repeated blocks. The set $N_{t,k} = N_{t,k}(X)$ of all null t -designs is a $\mathbb{Z}$ -submodule of V_k . It is well known that a necessary condition for existence of a $t - (v, k, \lambda)$ -design is that

$$(1.1) \quad \lambda \binom{v-i}{t-i} \equiv 0 \pmod{\binom{k-i}{t-i}}, \quad 0 \leq i \leq t.$$

Any quadruple $t - (v, k, \lambda)$ of integers is said to be *feasible* if it satisfies the above conditions (1.1) and $\lambda \geq 0$.

The aim of this note is to prove the following theorems.

* Received by the editors December 22, 1986; accepted for publication August 31, 1987.

† Department of Mathematics, Ohio State University, Columbus, Ohio 43210.

‡ School of Mathematics, Tata Institute of Fundamental Research, Colaba, Bombay 400 005 India. This work was done while the author was a visitor at Ohio State University, Columbus, Ohio 43210.

THEOREM 1.1. *There exist functions $c(t, k)$ and $s(t, k)$ such that for all feasible quadruples $t - (v, k, \lambda)$ with*

$$v^{k-2t} \geq c(t, k)\lambda$$

there exists an elementary signed $t - (v, k, \lambda)$ -design f with $|\text{supp } f| = |f| \leq s(t, k)\lambda \binom{v}{t}$.

THEOREM 1.2. *There exists a function $c_1(t, k)$ such that for any feasible quadruple $t - (v, k, \lambda)$ with*

$$\binom{v-t}{k-t} + c_1(t, k)v^{k-2t} \geq \lambda \geq \binom{v-t}{k-t} - c_1(t, k)v^{k-2t}$$

there exists a $t - (v, k, \lambda)$ -design in which every block has multiplicity ≤ 2 .

THEOREM 1.3. *There exists a function $c_1(t, k)$ such that for any feasible quadruple $t - (v, k, \mu)$,*

$$0 \leq \mu < \binom{v-t}{k-t} \quad \text{and} \quad \lambda = m \binom{v-t}{k-t} + \mu,$$

(i) *$t - (v, k, \lambda)$ -design exists for all integers $m \geq 0$, if*

$$\mu \geq \binom{v-t}{k-t} - c_1(t, k)v^{k-2t},$$

(ii) *$t - (v, k, \lambda)$ -design exists for all integers $m \geq 1$, if*

$$\mu \leq c_1(t, k)v^{k-2t}.$$

We note here that Theorems 1.1, 1.2, or 1.3 essentially give designs only for large v and $k \geq 2t + 1$. It may also be noted that the bound on $|\text{supp } f|$ in Theorem 1.1 shows that the number of blocks with multiplicity -1 in these designs is not too large.

A well-known result of Wilson [W] states that for any feasible quadruple $t - (v, k, \mu)$, $0 \leq \mu < \binom{v-t}{k-t}$, and $\lambda = m \binom{v-t}{k-t} + \mu$, a $t - (v, k, \lambda)$ -design exists for all integers m sufficiently large.

Theorem 1.3 above shows that for $\mu \geq \binom{v-t}{k-t} - c_1(t, k)v^{k-2t}$, “ m sufficiently large” in Wilson’s Theorem can be replaced by $m \geq 0$, and similarly for $\mu \leq c_1(t, k)v^{k-2t}$, “ m sufficiently large” can be replaced by “ $m \geq 1$.” We also note here that Teirlinck [T] has proved recently the existence of $t - (v, k, \lambda)$ -designs with no repeated blocks for $k = t + 1$.

Theorems 1.2 and 1.3 are easy consequences of Theorem 1.1. The inequalities we obtain in these theorems are not the best possible. There is room for improvement. In particular, perhaps v^{k-2t} in these theorems can be replaced by v^{k-t} to obtain better estimates. Perhaps even better estimates may be obtained by the method used by Wilson [W].

The proof of Theorem 1.1 is essentially based on studying various expressions of a null t -design in terms of a set of generators for the module $N_{t,k}$. This set of generators was first given by Graver and Jukart [GJ]. We will use a convenient form of this set of generators, first given by Graham, Li, and Li [GLL], which we now describe.

For each $A \subset X$, define a monomial $P(A) = \prod_{x \in A} x$, $P(\emptyset) = 1$. We identify A with $P(A)$. Thus we also think of $P_k(X)$ as the set of all square free monomials of degree k in variables from X and each $f \in V_k$ as a polynomial $f = \sum_{A \in P_k(X)} f(A)P(A)$.

Thus V_k is also the free $\mathbb{Z}$ -module of all those homogeneous polynomials of degree k with variables from X in which each monomial with nonzero coefficient is square free.

Now let $\mathbf{B} = (x_1, x_2, \dots, x_m)$ be a sequence of distinct elements of X . We will denote by B the set $\{x_1, x_2, \dots, x_m\}$. For each sequence $\mathbf{B} = (x_1, x_2, \dots, x_{2t+2})$, $|B| = 2t + 2$, we denote by $q_{\mathbf{B}}$ the polynomial

$$q_{\mathbf{B}} = (x_1 - x_2)(x_3 - x_4) \cdots (x_{2t+1} - x_{2t+2}).$$

We can now state Graver and Jukart's Theorem [GJ] (see also [GLL]).

THEOREM 1.4. *Let $v \geq k + t + 1$. The $\mathbb{Z}$ -module $N_{t,k}(X)$ is generated by polynomials $q_{\mathbf{B}}C$ where $C \in P_{k-t-1}(X)$, and $\mathbf{B}$ varies over all sequences of length $2t + 2$ such that $B \cap C = \emptyset$.*

2. Proofs of theorems. It is clear from Theorem 1.4 that for each $f \in N_{t,k}$ there exist $\mathbf{B}_1, \mathbf{B}_2, \dots, \mathbf{B}_l, C_1, C_2, \dots, C_l$ such that $f = \sum_{i=1}^l q_{\mathbf{B}_i} C_i$. Define $\|f\|$ to be minimum l for which such an expression exists for f . Clearly, $\|f + g\| \leq \|f\| + \|g\|$. For any $f \in V_k$, define $f_+ : P_k(X) \rightarrow \mathbb{Z}$ and $f_- : P_k(X) \rightarrow \mathbb{Z}$ by setting

$$f_+(B) = \begin{cases} f(B) & \text{if } f(B) > 0, \\ 0 & \text{otherwise,} \end{cases}$$

$$f_- = -f + f_+.$$

Define a function $e_1(t, k)$, $t \geq 0, k \geq 0$ inductively by

$$e_1(k, 0) = k/2 \quad \text{for } k \geq 0,$$

$$e_1(k, t) = 0 \quad \text{for } k \leq t,$$

and

$$e_1(k, t) = e_1(k-1, t) + e_1(k, t-1) + 2^t e_1(k-1, t) e_1(k, t-1) \quad \text{for } k > t.$$

LEMMA 2.1. *For each $f \in N_{t,k}(X)$, $\|f\| \leq e_1(t, k)|f|$.*

Proof. The proof is by double induction on t, k .

For monomials $x_1, x_2, \dots, x_k$ and $y_1, y_2, \dots, y_k$ in $P_k(X)$

$$x_1 x_2 \cdots x_k - y_1 y_2 \cdots y_k = (x_1 - y_1)x_2 x_3 \cdots x_k \\ + (x_2 - y_2)y_1 x_3 \cdots x_k + \cdots + (x_k - y_k)y_1 y_2 \cdots y_{k-1} \quad (k \text{ terms}).$$

Now suppose that $f \in N_{t,k}$, $t = 0$, and $k > 0$. Then, clearly, $|f_+| = |f_-| = \frac{1}{2}|f|$. Using this and above equation it is clear that

$$\|f\| \leq \frac{1}{2}k|f|.$$

The lemma now follows for $t = 0$. Also the lemma is trivially true for $k \leq t$, since $N_{t,k} = (0)$. Now suppose that $f \in N_{t,k}$, $t > 0, k > t$. Let $x \in X$. Define $f_x : P_{k-1}(X \setminus \{x\}) \rightarrow \mathbb{Z}$ and $f_{\bar{x}} : P_k(X \setminus \{x\}) \rightarrow \mathbb{Z}$ by setting

$$f_x(B) = f(BU\{x\}) \quad \text{for all } B \in P_{k-1}(X \setminus \{x\}),$$

$$f_{\bar{x}}(B) = f(C) \quad \text{for all } C \in P_k(X \setminus \{x\}).$$

Clearly $f_x \in N_{t-1,k-1}$, $f_{\bar{x}} \in N_{t-1,k}$, and $f = f_{\bar{x}} + x f_x$.

Now for $f_{\bar{x}}$, there exist $\mathbf{A}_i, C_i$, $1 \leq i \leq \|f_{\bar{x}}\| = a$ (say) $|A_i| = 2t, |C_i| = k - t > 0$ such that

$$f_{\bar{x}} = \sum_{i=1}^a q_{\mathbf{A}_i} C_i.$$

For each i choose $y_i \in C_i$. Define

$$g = \sum_{i=1}^a q_{A_i}(y_i - x)(C_i/y_i)$$

and

$$h = \frac{1}{x}(f - g) = f_x + \sum_{i=1}^a q_{A_i}(C_i/y_i).$$

Note that $g \in N_{t,k}$, $h \in N_{t,k-1}$.

Now by induction hypothesis

$$\|f_{\bar{x}}\| \leq e_1(t-1, k)|f_{\bar{x}}|$$

and

$$\|h\| \leq e_1(t, k-1)|h|.$$

Also using definitions of g and h , since $|q_{A_i}C_i| = 2^t$, we have

$$|h| \leq |f_x| + 2^t\|f_{\bar{x}}\|,$$

$$\|g\| \leq \|f_{\bar{x}}\|,$$

$$|f_x| + |f_{\bar{x}}| = |f|.$$

Hence as $f = xh + g$, $|f_x| \leq |f|$ and $|f_{\bar{x}}| \leq |f|$.

$$\begin{aligned} \|f\| &\leq \|g\| + \|h\| \leq \|f_{\bar{x}}\| + \|h\| \leq e_1(t-1, k)|f_{\bar{x}}| + e_1(t, k-1)(|f_x| + 2^te_1(t-1, k)|f_{\bar{x}}| \\ &\leq (e_1(t-1, k) + e_1(t, k-1) + 2^te_1(t-1, k)e_1(t, k-1))|f|. \end{aligned}$$

Thus, $\|f\| \leq e_1(t, k)|f|$.

This proves Lemma 2.1.

Define a homomorphism $A_{t,k}: V_k \rightarrow V_t$ by

$$A_{t,k}(B) = \sum_{T \subseteq B} T \quad \text{for all } B \in P_k(X).$$

It can be easily checked that $A_{t,k}(N_{t-1,k}) = N_{t-1,t}$. Also for $\mathbf{A}$, $|A| = 2t$, $A \subseteq X$ and $C \subseteq X$, $|C| = k - t - 1$, $A_{t,k}(q_{\mathbf{A}}C) = q_{\mathbf{A}}$.

LEMMA 2.2. Let $F \subseteq P_k(X)$ and $f \in N_{t-1,t}$. If

$$\binom{v-2t}{k-t} > \binom{k}{k-t}|F| + \binom{k+t}{k-t}(\|f\| - 1),$$

then there exists $g \in N_{t-1,k}$ such that

- (i) g is an elementary signed $(t-1) - (v, k, \lambda)$ -design with $\lambda = 0$;
- (ii) $\text{supp } g \cap F = \emptyset$;
- (iii) $A_{t,k}g = f$;
- (iv) $|g| = 2^t\|f\|$.

Proof. We can choose $\mathbf{A}_i$, $1 \leq i \leq \|f\|$, $|A_i| = 2t$ such that $f = \sum_{i=1}^{\|f\|} q_{\mathbf{A}_i}$.

We first choose by induction for each A_i a set $C_i \subseteq X$, $A_i \cap C_i = \emptyset$, $|C_i| = k - t$, such that

- (a) $C_i \not\subseteq A_j \cup C_j$ for $j < i$;
- (b) $C_i \not\subseteq B$ for all $B \in F$.

For each i , $1 \leq i \leq \|f\|$, the total number of $(k-t)$ subsets of $A_j \cup C_j$ $j < i$ or B , $B \in F$ is clearly at most

$$\binom{k}{k-t}|F| + \binom{k+t}{k-t}(\|f\| - 1).$$

Now since the total number of $(k-t)$ subsets of X disjoint from A_i is $\binom{v-2t}{k-t}$, from the given hypothesis it follows that we can choose C_i , $i \leq 1 \leq \|f\|$ satisfying the above conditions. Define $g \in N_{t-1,k}$ by

$$g = \sum_{i=1}^{\|f\|} q_{A_i} C_i.$$

From the choice of C_i 's it is clear that g is elementary and $\text{supp } g \cap F = \emptyset$.

Also, clearly, $A_{t,k}g = f$ and since $|q_{A_i} C_i| = 2^t$ it follows easily that $|g| = 2^t \|f\|$. This completes the proof.

Define a function $e_2(t, k)$ by

$$(2.1) \quad e_2(0, k) = 1, \quad e_2(t, k) = 0 \quad \text{if } t > k.$$

$$e_2(t, k) = \frac{t}{k-t+1} e_2(t-1, k) + 2^t \frac{t}{k-t+1} \binom{k}{t} e_1(t, k) e_2(t-1, k) + 2^t e_1(t, k) \quad \text{if } t \leq k.$$

For any feasible quadruple $t - (v, k, \lambda)$ the quadruples $i - (v, k, \lambda_i)$, $0 \leq i \leq t$, where λ_i is defined by $\lambda_t = \lambda$ and $\lambda_i = ((v-i)/(k-i))\lambda_{i+1}$ for $0 \leq i < t$ are also feasible. In fact, they correspond to considering a given t -design as an i -design.

LEMMA 2.3. *Let $t - (v, k, \lambda)$ be a feasible quadruple such that for $0 \leq i < t$,*

$$\binom{v-2i-2}{k-i-1} > \lambda_{i+1} \binom{v}{i+1} \left[\frac{i+1}{k-i} \binom{k}{k-i-1} e_2(i, k) \left(1 + \binom{k+i+1}{k-i-1} e_1(i+1, k) \right) + \binom{k+i+1}{k-i-1} e_1(i+1, k) \right] - \binom{k+i+1}{k-i-1}.$$

Then there exists an elementary signed $t - (v, k, \lambda)$ -design f such that

$$|f| \leq e_2(t, k) \lambda \binom{v}{t}.$$

Proof. For $0 \leq i \leq t$, by induction on i we will construct an elementary signed $i - (v, k, \lambda_i)$ -design f_i with $|f_i| \leq e_2(i, k) \lambda_i \binom{v}{i}$. For $i = 0$, clearly, f_0 can be taken to be any elementary element of V_k with $|f_0| = \lambda_0$.

Now suppose we have constructed an $i - (v, k, \lambda_i)$ signed design f_i , which is elementary, and $|f_i| \leq e_2(i, k) \lambda_i \binom{v}{i}$.

We now show how to construct f_{i+1} . Define $f'_{i+1} = A_{i+1,k} f_i - \lambda_{i+1} j_{i+1}$ where $j_{i+1} \in V_{i+1}$ is defined by $j_{i+1}(B) = 1$ for all $B \in P_{i+1}(X)$.

It can be easily seen that $f'_{i+1} \in N_{i,i+1}$. Also

$$(2.2) \quad |f'_{i+1}| \leq \binom{k}{i+1} |f_i| + \lambda_{i+1} \binom{v}{i+1} \cdots,$$

$$(2.3) \quad |f_i| \leq e_2(i, k) \lambda_i \binom{v}{i} = e_2(i, k) \left(\frac{i+1}{k-i} \right) \lambda_{i+1} \binom{v}{i+1} \cdots,$$

and using Lemma 2.1

$$\|f'_{i+1}\| \leq e_1(i+1, k)|f'_{i+1}| \leq e_1(i+1, k) \left(\binom{k}{k-i-1} |f_i| + \lambda_{i+1} \binom{v}{i+1} \right).$$

Hence from the above inequalities

$$\begin{aligned} \binom{k}{k-i-1} |f_i| + \binom{k+i+1}{k-i-1} (\|f'_{i+1}\| - 1) \\ \leq \binom{k}{k-i-1} |f_i| \left(1 + \binom{k+i+1}{k-i-1} e_1(i+1, k) \right) \\ + \binom{k+i+1}{k-i-1} \left(\lambda_{i+1} \binom{v}{i+1} e_1(i+1, k) - 1 \right) \\ \leq \text{the right-hand side of the inequality of the lemma.} \end{aligned}$$

From the last inequality and the given hypothesis, it is clear that the conditions of Lemma 2.2 are satisfied for $F = \text{supp } f_i = |f_i|$, $t = i+1$, $k = k$ and $f = f'_{i+1}$. Hence using Lemma 2.2 we can construct an elementary $g \in N_{i,k}$ such that

$$\begin{aligned} (i) \quad & A_{i+1,k} g = f'_{i+1}; \\ (2.4) \quad (ii) \quad & \text{supp } g \cap \text{supp } f_i = \emptyset; \\ (iii) \quad & |g| = 2^{i+1} \|f'_{i+1}\| \leq 2^{i+1} e_1(i+1, k) |f'_{i+1}| \cdots. \end{aligned}$$

Now we take $f_{i+1} = f_i - g$. Clearly, f_{i+1} is elementary and

$$A_{i+1,k} f_{i+1} = A_{i+1,k} f_i - A_{i+1,k} g = \lambda_{i+1} j_{i+1},$$

$$\begin{aligned} |f_{i+1}| &\leq |f_i| + |g| \\ &\leq \left[\frac{i+1}{k-i} e_2(i, k) + 2^{i+1} e_1(i+1, k) \left[e_2(i, k) \binom{i+1}{k-i} \binom{k}{i+1} + 1 \right] \right] \lambda_{i+1} \binom{v}{i+1} \\ &= e_2(i+1, k) \lambda_{i+1} \binom{v}{i+1} \quad \text{using (2.1), (2.2), (2.3), and (2.4).} \end{aligned}$$

This completes the proof.

Proof of Theorem 1.1. Using the definitions of $e_1(i, k)$, $e_2(i, k)$, and λ_i , we can easily write the right-hand and the left-hand side, respectively, of the inequality in Lemma 2.3 as

$$\lambda(c_i^i v^t + c_{i-1}^i v^{t-1} + \cdots + c_0^i), \quad \text{and}$$

$$d_{k-i-1}^i v^{k-i-1} + d_{k-i-2}^i v^{k-i-2} + \cdots + d_0^i$$

where c_l^i , d_m^i , $0 \leq l \leq t$, $0 \leq m \leq k-t$, $0 \leq i \leq t$ are bounded by some functions of k and t . Therefore we can define a function $c(t, k)$ of t and k such that if $v^{k-2t} \geq \lambda c(t, k)$ and $t - (v, k, \lambda)$ is feasible, all the inequalities of Lemma 2.3 hold. Hence Theorem 1.1 follows with $s(t, k) = e_2(t, k)$.

Proof of Theorem 1.2. Let $t - (v, k, \lambda)$ be a feasible quadruple satisfying conditions of Theorem 1.2 where $c_1(t, k) = 1/c(t, k)$ and $c(t, k)$ is the function given by Theorem 1.1. Let $\lambda' = |\lambda - \binom{v-i}{k-i}|$. Now it can be easily seen that $t - (v, k, \lambda')$ is a feasible quadruple satisfying conditions of Theorem 1.1. Let f_1 be an elementary signed $t - (v, k, \lambda')$ design obtained using Theorem 1.1. Let $j \in V_k$ be defined by $j(B) = 1$ for all $B \in P_k(X)$. j is a

$t - (v, k, \lambda_1)$ -design with $\lambda_1 = \binom{v-t}{k-t}$. Now, since f_1 is elementary, it can be easily seen that $f = f_1 + j$ or $f = j - f_1$ is a $t - (v, k, \lambda)$ -design accordingly as $\lambda \geq \binom{v-t}{k-t}$ or $\lambda \leq \binom{v-t}{k-t}$ and the multiplicity of every block in f is 0, 1, or 2. This proves the theorem.

Proof of Theorem 1.3 is also clear. In fact, the design $mj + f$ is a $t - (v, k, \lambda)$ -design, where f is a $t - (v, k, \mu)$ -design given by Theorem 1.2.

REFERENCES

- [GJ] J. E. GRAVER AND W. B. JUKART, *The module structure of integral designs*, J. Combin. Theory Ser. A, 15 (1973), pp. 75–90.
- [GLL] R. L. GRAHAM, S-Y. R. LI, AND W. W. W. LI, *On the structure of t -designs*, SIAM J. Algebraic Discrete Methods, 1 (1980), pp. 8–14.
- [W] R. M. WILSON, *The necessary conditions for t -designs are sufficient for something*, Utilitas Math., 4 (1973), pp. 207–215.
- [T] L. TEIRLINCK, *Nontrivial t -designs without repeated blocks exist for all t* , Auburn University, Alabama, preprint.

A NOTE ON INDEPENDENT SETS IN TREES*

BRUCE E. SAGAN†

Abstract. We give a simple graph-theoretical proof that the largest number of maximal independent vertex sets in a tree with n vertices is given by

$$m(T) = \begin{cases} 2^{k-1} + 1 & \text{if } n = 2k, \\ 2^k & \text{if } n = 2k + 1, \end{cases}$$

a result first proved by Wilf [*SIAM J. Algebraic Discrete Methods*, 7 (1986), pp. 125–130]. We also characterize those trees achieving this maximum value. Finally we investigate some related problems.

Key words. independent vertices, trees, extremal graphs

AMS(MOS) subject classifications. 05C35, 05C30, 05C70

1. Introduction. Herbert Wilf [5] was the first to answer the following question: What is the largest number of maximal independent vertex sets in a tree with n vertices? His proof had an algebraic flavor and was somewhat complicated. Subsequently Daniel Cohen [1] was able to provide a graph-theoretical proof, but one which was still fairly complex in view of the simplicity of the bound (see Theorem 3 below). The purpose of this note is to give a simple graph-theoretical demonstration of this result which, in addition, completely characterizes all trees achieving the maximum value. J. Griggs and C. Grinstead [2] independently found a straightforward proof which is similar to ours in some respects but differs in others.

2. Maximizing independent sets. We begin with some preliminary definitions and lemmas. For any concepts that are not defined, the reader can consult Harary's book [4].

Given a graph, G , let $V(G)$ be the vertex set of G and let $v(G) = |V(G)|$ where $| \cdot |$ denotes cardinality. Recall that a vertex $u \in V(G)$ is called an *endpoint* if $\deg u = 1$. We will say that a vertex $v \in V(G)$ is *penultimate* if v is not an endpoint and v is adjacent to (at least) $\deg v - 1$ endpoints. Note that v is adjacent to $\deg v$ endpoints if and only if v is the center of the star K_{1n} .

LEMMA 1. *Every finite tree T with $v(T) \geq 3$ has a penultimate vertex.*

Proof. The next-to-last vertex on any diameter must be penultimate. $\square$

If v is penultimate in T , then $T - v$ consists of $\deg v - 1$ isolated vertices and one other component called the *penultimate component* P (if v is the center of a star, choose any fixed component as the penultimate one). Now let

$$\text{End } v = \{w \notin P \mid w \text{ is adjacent to } v\}$$

so that $V(T) = V(P) \cup \{v\} \cup \text{End } v$ where $\cup$ denotes disjoint union.

Call a set $I \subseteq V(G)$ *independent* if no two vertices of I are adjacent in G . Now let $M(G) = \{I \subseteq V(G) \mid I \text{ is independent and maximal}\}$, i.e., if $I \in M(G)$ then there is no independent set J with $I \subsetneq J$. Also set $m(G) = |M(G)|$. We wish to find the maximum value of $m(T)$ over all trees T with $v(T) = n$. First, however, we need an upper bound.

* Received by the editors April 28, 1986; accepted for publication (in revised form) March 24, 1987. This research was supported in part by a NATO post-doctoral grant administered by the National Science Foundation.

† Department of Mathematics, University of Pennsylvania, Philadelphia, Pennsylvania 19104-6395. Present address, Department of Mathematics, Michigan State University, East Lansing, Michigan 48824.

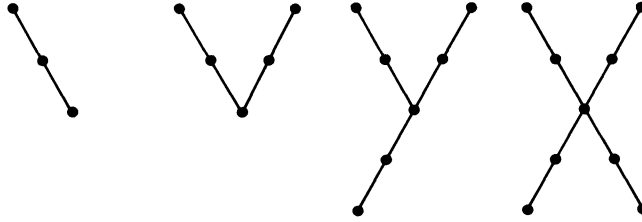


FIG. 1. Batons of length 0.

LEMMA 2. Let T be a tree and $v \in V(T)$ be penultimate with corresponding component P . Then

$$m(T) \leq 2m(P).$$

Proof. Let I be a maximal independent set in T . Then either $\text{End } v \subseteq I$ or $v \in I$ (exclusive or). In the first case, $I = \text{End } v \cup I_p$ where I_p is a maximal independent set of P . In the second, $I = \{v\} \cup (I_p - \{w\})$ where w is the unique vertex of P adjacent to v (w need not be in I_p). $\square$

Define a *baton of length l* as follows. Start with a path L of length l and attach any number of paths of length two to L 's endpoints. Hence the batons of length 0 are just "extended" stars and the first few are displayed in Fig. 1.

Similarly, the batons of length 1 form a family some of whose members are shown in Fig. 2.

THEOREM 3. Among all labeled trees T with n vertices, the maximum value of $m(T)$ is

$$m(T) = \begin{cases} 2^{k-1} + 1 & \text{if } n = 2k, \\ 2^k & \text{if } n = 2k + 1. \end{cases}$$

Furthermore this maximum is attained only by the batons of length 0 (when n is odd) or by the batons of lengths 1 and 3 (when n is even).

Proof. Induct on n . The theorem can be checked by hand for $v(T) \leq 10$ using Lemma 2 and Harary's tables [4, pp. 233–234]. (The author has made this calculation and does *not* recommend it to the reader.) Now let T be a tree with $m(T)$ maximum among all trees with $v(T) = n > 10$. By Lemma 1 there exists a penultimate vertex $v \in V(T)$ with corresponding component P . There are two cases depending upon the parity of n .

If n is odd, $n = 2k + 1$, then consider the unique baton of length 0 with n vertices, denoted T_n . Since T_{2k+1} contains k paths of length 2, a simple calculation shows that $m(T_{2k+1}) = 2^k$. Hence

$$(1) \quad m(T) \geq 2^k.$$

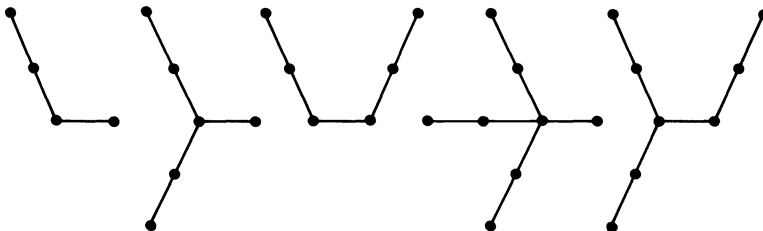
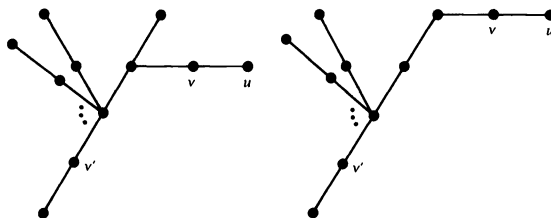


FIG. 2. Batons of length 1.

FIG. 3. G_1 and G_2 .

Now $v(P) \leq n - 2 = 2k - 1$. However, if $v(P) < 2k - 1$, then by Lemma 2 and induction (for n at least 7) we have

$$(2) \quad m(T) \leq 2m(P) < 2 \cdot 2^{k-1} = 2^k,$$

which contradicts (1). Thus $v(P) = 2k - 1$ which implies that $\deg v = 2$, and $\text{End } v = \{u\}$ for a single vertex u . Furthermore $P = T_{2k-1}$ since if it does not, induction applies which yields $m(P) < 2^{k-1}$ (this baton is the unique tree attaining the maximum value). But then (2) holds as before, a contradiction unless $P = T_{2k-1}$.

Putting all these facts together, we see that T consists of a tree T_{2k-1} with a path of length two $w-v-u$ attached to some $w \in V(T_{2k-1})$. This leaves only three possibilities for T : T_{2k+1} , G_1 or G_2 , where G_1 and G_2 are given in Fig. 3.

To eliminate G_1 and G_2 as possibilities, consider a second penultimate vertex v' as shown. If p' is the corresponding component, then for $n \geq 9$ we have $P' \neq T_{2k-1}$. Invoking Lemma 2 again we see that

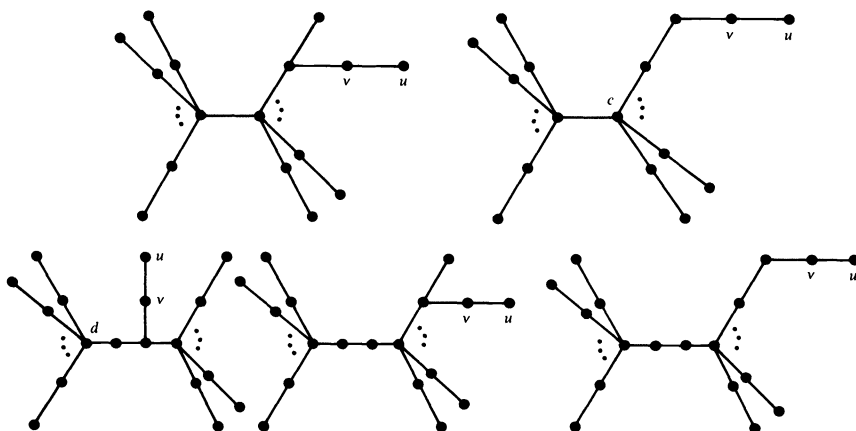
$$m(G_i) \leq 2m(P') < 2 \cdot 2^{k-1} = 2^k$$

for $i = 1, 2$ and so neither G_i maximizes $m(T)$.

For n even, $n = 2k$, exactly the same line of reasoning as in the odd case can be used. It follows that the only possibilities for T are those obtained by attaching a path of length 2 to a baton of length 1 or 3 on $2k - 2$ vertices. Hence T is either a baton of length 1 or 3 and $m(T) = 2^{k-1} + 1$ or T is one of the five graphs $H_1, \dots, H_5$ shown in Fig. 4.

Note that in H_2 (respectively H_3) we require that $\deg c \geq 3$ (respectively $\deg d \geq 2$) so that the graph does not degenerate into a baton of length 3 (respectively 1).

It is easy to verify that because $n > 10$ we can find in each of these five graphs a second penultimate vertex v' such that P' is not a baton of length 1 or 3. It follows from

FIG. 4. H_1 through H_5 .

induction and from $v(P') = n - 2 = 2k - 2$ that we have $m(P') \leq (2^{k-2} + 1) - 1 = 2^{k-2}$. Hence by Lemma 2

$$m(H_i) \leq 2m(P') \leq 2 \cdot 2^{k-2} = 2^{k-1}$$

for $i = 1, \dots, 5$ which is less than the value obtained for the batons. This finishes the proof of the theorem. $\square$

Call a tree T *extremal* if $m(T)$ is a maximum as compared to all other trees with the same number of vertices. Let $e(n)$ be the number of extremal trees, up to the labeling, on n vertices.

COROLLARY 4. *The number of extremal trees on n vertices is given by*

$$e(n) = \begin{cases} 1 & \text{if } n = 2k + 1, \\ k - 1 & \text{if } n = 2k. \end{cases}$$

Proof. It is a simple matter to count the number of batons of the appropriate sizes. $\square$

3. Remarks. Finding the minimum value of $m(T)$ is quite easy.

THEOREM 5. *The minimum value of $m(T)$ over all trees with n vertices, $n \geq 2$, is $m(T) = 2$. Furthermore the unique tree (up to relabeling) achieving this minimum is the star $K_{1,n-1}$.*

Proof. If $v(T) \geq 2$, then for any edge vw there is a maximal independent set containing v and a different one containing w . Hence $m(T) \geq 2$ and clearly $m(K_{1,n-1}) = 2$.

If $T \neq K_{1,n-1}$, then T contains a path $u-v-w-x$. This forces $m(T) \geq 3$ since a third maximal independent set containing u and x also exists. $\square$

Once one has determined the lower and upper bounds, b and B , respectively, for a graphical invariant $\beta(G)$ one looks for an *interpolation theorem*. Such a result has the following form: For all integers z satisfying $b \leq z \leq B$ there is a graph G with $\beta(G) = z$. Unfortunately there is no interpolation theorem for $m(T)$ since when $v(T) = 9$ we have $2 \leq m(T) \leq 16$ but there is no tree with $m(T) = 15$.

We should compare our proof of Theorem 3 with that of Griggs and Grinstead mentioned in the Introduction. They also begin by proving Lemmas 1 and 2. Then, however, they use the lemmas to show that the maximum value of $m(F)$ over all forests F with $v(F) = n$ is achieved precisely when F is a one-factor (possibly with an isolated vertex). By carefully amalgamating the components of the one-factor, they finally obtain the bound and extremal graphs for trees.

Following the dictum that once something is proved for trees it should be extended to all connected graphs, one is led to pose the following question: What is the maximum value of $m(G)$ over all connected graphs G with $v(G) = n$? Griggs, Grinstead and Guichard [3] have answered this query. Another proof has been found independently by Füredi.

Acknowledgments. I would like to thank Mihály Hujter for pointing out the proof of Lemma 1. I also thank the referee for suggestions that considerably improved the exposition.

REFERENCES

- [1] D. COHEN, *Counting stable sets in trees*, in Seminaire Lotharingien de combinatoire, 10^{ème} session, R. König, ed., Institute de Recherche Mathématique Avancée pub., Strasbourg, France, 1984, pp. 48–52.
- [2] J. GRIGGS AND C. GRINSTEAD, private communication.
- [3] J. GRIGGS, C. GRINSTEAD AND D. GUICHARD, *The number of maximal independent sets in a connected graph*, preprint.
- [4] F. HARARY, *Graph Theory*, Addison-Wesley, Reading, MA, 1969.
- [5] H. WILF, *The number of maximal-independent sets in a tree*, SIAM J. Algebraic Discrete Methods, 7 (1986), pp. 125–130.

QUICK GOSSIPING BY CONFERENCE CALLS*

ÁKOS SERESS†

Abstract. There are n persons. Each knows a different item of information. They communicate by k -conference calls, i.e., k persons participate in each conversation. Our goal is to give a sequence of calls subject to the fact that everyone hears each gossip exactly once; moreover, the number of calls is as small as possible. Appropriate sequences of conversations do not exist for all n ; we give calling schemes for all feasible n 's, with finitely many exceptions, subject to the fact that the number of calls is a linear function of n .

Key words. gossip problem, labeled hypergraph

AMS(MOS) subject classification. 05C65

1. Introduction. The original version of the gossip problem was proposed by A. Boyd and popularized by P. Erdős. There are n persons. Each knows a different item of information. They communicate by phone calls, and whenever two persons talk they tell each gossip they know at that time. We seek the minimal number $g(n)$ of calls required to transmit all the information to everyone. For $n \geq 4$, $g(n) = 2n - 4$, as was proved independently by B. Baker and R. Shostak [1], R. Bumby [3], A. Hajnal, E. Milner and E. Szemerédi [4], J. Spencer (unpublished) and R. Tijdeman [12].

In another version of the problem, we require that everyone has to hear each gossip exactly once. (We call such a sequence of calls to a *good calling scheme*.) Good calling sequences do not exist for all n . The set F of feasible n 's was determined in [8]:

$$F = \{1, 2, 4, 8, 12, 16\} \cup \{n: n \geq 20 \text{ and } n \text{ is even}\}.$$

For $n \in F$, let $f(n)$ denote the minimum number of calls in a good calling scheme. D. West [13] proved that $f(n) \leq 2.25n - 6$ for $4|n$. In [10] and [11], the author proved

$$f(n) = 2.25n - 6 \quad \text{if } n \geq 8, 4|n, \text{ and}$$

$$2.25n - 4.5 \leq f(n) \leq 2.25n - 3.5 \quad \text{if } n \geq 22, n \equiv 2 \pmod{4}.$$

The problems above can be generalized for k -conference calls, i.e., when k persons participate in each conversation. Denoting the minimum number of necessary calls by $g_k(n)$, J.-C. Bermond [2], K. Lebensold [6], D. Kleitman and J. Shearer [5] proved that

$$g_k(n) = 2 \left\lceil \frac{n-k}{k-1} \right\rceil + 1 \quad \text{if } k \leq n < k^2,$$

$$2 \left\lceil \frac{n-k}{k-1} \right\rceil \quad \text{if } n \geq k^2.$$

Again, requiring that everyone hears each gossip exactly once, appropriate calling sequences do not exist for all n . Denoting the set of feasible n 's by F_k , in [9] we proved the following theorem.

- THEOREM 1.1.** (i) If $n \in F_k$, $n > 1$, then $n = k + xk(k-1)$ for some integer x .
(ii) For each $k \geq 3$, there exists $x_0(k)$ s.t. for all $x \geq x_0(k)$, $k + xk(k-1) \in F_k$.
(iii) $x_0(k) \leq 4k^2 + 2k$.

* Received by the editors January 6, 1986; accepted for publication (in revised form) March 20, 1987. This research was partially supported by the Hungarian National Foundation for Scientific Research grant 1812.

† Mathematical Institute of the Hungarian Academy of Sciences, Budapest, H-1053 Hungary.

For $n \in F_k$, denote by $f_k(n)$ the minimum number of calls in good calling schemes. The recursive constructions in [9] use $c \cdot n \cdot \log n$ calls, so they do not give good estimations for $f_k(n)$. (In fact, $c \cdot n \cdot \log n$ is the greatest possible number of calls in good schemes; see D. Miklós, M. Newman, Á. Seress and D. West [7].)

The purpose of this paper is to give a linear upper bound for $f_k(n)$. We shall prove the following theorem.

THEOREM 1.2. *Let $c_k = (3k^2 - k - 1)/(k^3 - k^2)$. Then $f_k(n) \leq c_k \cdot n + d_k$, where d_k depends only on k .*

Moreover, we obtain a better estimation for $x_0(k)$, namely $x_0(k) \leq k^2 + 5k + 2$. Concerning the exact value of $x_0(k)$, we repeat the conjecture stated in [9]: if $x \geq 2k + 1$, $x \neq 3k$ then $k + xk(k - 1) \in F_k$. (The conjecture is true in the cases $k = 3, 4$.)

Although we cannot give a good estimation for d_k (since we know the elements of F_k only with finitely many exceptions), we shall see that $n = k + xk(k - 1)$, $x \geq k^2 + 5k + 2$, $k \neq 4$ imply $f_k(n) < c_k \cdot n - k$.

By Theorem 1.1(i), the feasible n 's can be written in the form

$$n = k^2 - m(k^2 - k) + x(k^3 - k^2)$$

for some integer x , and $0 \leq m \leq k - 1$. We shall use four constructions, according to the following cases:

- (i) $m = 0$;
- (ii) $m = 1$;
- (iii) $2 \leq m \leq k - 1$, except the case $k = 4$, $m = 3$;
- (iv) $k = 4$, $m = 3$.

The first three constructions can be considered as special cases of one, more general construction. However, in order to increase readability, we prefer to separate these cases.

2. The constructions.

CONSTRUCTION 2.1. For $n = k^2 + x(k^3 - k^2)$, $x \geq 3$, there exists a good calling scheme of cardinality

$$(2.1.1) \quad c_k \cdot n - 2k - [k/(k - 1)].$$

Step 1. We divide the persons into $y = 1 + x(k - 1)$ groups, $A_1, A_2, \dots, A_y$; $|A_i| = k^2$ for each $1 \leq i \leq y$. We perform $2k$ calls in each group s.t. everyone hears each gossip in his own group (first, we form k disjoint k -tuples; second, we pick one person from each original k -tuple, obtaining another set of disjoint k -tuples). After these calls, persons in the same group know the same set of gossips; we refer to the information known in A_i (it is the union of k^2 original gossips) as the i th joint gossip (JG).

Step 2. We form a set $Y = \{a_1, a_2, \dots, a_{y-2(k-1)}, b_{2k-1}, b_{2k}, \dots, b_y\}$, where $a_i, b_i \in A_i$ for each i . Clearly, $|Y| = 2y - 4(k - 1)$. In Y , we perform the calls

$$(a_1, a_2, \dots, a_k),$$

$$(a_1, a_{k+1}, a_{k+2}, \dots, a_{2k-1}), \dots, (a_1, a_{y-3(k-1)+1}, \dots, a_{y-2(k-1)})$$

and

$$(b_y, b_{y-1}, \dots, b_{y-(k-1)}),$$

$$(b_y, b_{y-k}, b_{y-k-1}, \dots, b_{y-2(k-1)}), \dots, (b_y, b_{3k-3}, \dots, b_{2k-1}),$$

in that order. After these calls, a_1 knows the JG's 1 to $y - 2(k - 1)$, a_i ($2 \leq i \leq y - 2(k - 1)$) knows the JG's 1 to $1 + (k - 1) \cdot \lceil (i - 1)/(k - 1) \rceil$, b_y knows the JG's $2k - 1$ to y , and b_i ($2k - 1 \leq i \leq y - 1$) knows the JG's $y - (k - 1) \cdot \lceil (y - i)/(k - 1) \rceil$ to y .

Step 3. We form groups $B_1, B_2, \dots, B_y$; $|B_i| = k^2$ for each $1 \leq i \leq y$. From Y , we send a_i ($2 \leq i \leq y - 2(k - 1)$) into B_{i+2k-3} , a_1 into B_y , b_i ($2k - 1 \leq i \leq y - 1$) into $B_{i-(2k-3)}$, and b_y into B_1 . Persons from the groups $A_i \setminus Y$ are assigned to the new groups by a $y \times y$ matrix Z_x . The rows of Z_x are labeled by the groups B_i , the columns by the groups A_j , and the entry $z(i, j)$ gives the number of those persons from $A_j \setminus Y$ who are sent into B_i . Z_x must satisfy the following properties:

(α) $z(i, 1) + z(i, 2) + \dots + z(i, y) = |B_i \setminus Y|$ for each $1 \leq i \leq y$;

(β) $z(1, j) + z(2, j) + \dots + z(y, j) = |A_j \setminus Y|$ for each $1 \leq j \leq y$.

In each B_i , we should like to give a sequence of calls s.t., eventually, everyone hears each gossip exactly once. To this end, we require Z_x to satisfy the following properties:

(γ) $z(i, j) = 0$ if one of the persons in $B_i \cap Y$ knows the j th JG after Step 2;

(δ) $z(i, j) \in \{1, k\}$ if none of the persons in $B_i \cap Y$ knows the j th JG after Step 2.

We claim that the conditions (α)–(δ) imply that we can finish the algorithm in each new group. In each B_i , we can partition the persons into equivalence classes. Two persons are in the same class if they know the same set of JG's after Step 2. Each JG is known in exactly one equivalence class, and the size of each class is 1 or k . If, say, there are p classes of size k , then we can form $k - p$ k -tuples from the classes of size 1. Performing $k - p$ calls in these k -tuples, we obtain k groups of size k , and k additional calls finish the algorithm.

A possible matrix Z_x satisfying (α)–(δ) is the following (we list only the nonzero entries):

(a) If $1 \leq i \leq k$, then $z(i, i) = 1$, $z(i, j) = 1$ for $k + 1 \leq j \leq 2k - 2$, and $z(i, j) = k$ for $1 \leq j \leq k, j \neq i$.

If $y - (k - 1) \leq i \leq y$, then $z(i, i) = 1$, $z(i, j) = 1$ for $y - 2k + 3 \leq j \leq y - k$, and $z(i, j) = k$ for all $y - k + 1 \leq j \leq y, j \neq i$.

(b) $z(i, 1) = 1$ for $k + 1 \leq i \leq 2k - 2$.

If $y - 2k + 3 \leq i \leq y - k$, then $z(i, j) = k$ for $y - 2k + 3 \leq j \leq y - k$, and $z(i, j) = 1$ for $y - 3k + 4 \leq j \leq y - 2k + 2, y - k + 1 \leq j \leq y$.

(c) If $k + 1 \leq i \leq y - 2k + 2$ and $k - 1 \nmid i - 1$, then $z(i, j) = k$ for $i - k + 2 \leq j \leq i$, and $z(i, j) = 1$ for $i + 1 \leq j \leq i + k - 2$.

If $k + 1 \leq i \leq y - 2k + 2$ and $k - 1 \nmid i - 1$, then let

$$i' = (k - 1) \cdot \lceil (i - 1) / (k - 1) \rceil + 1.$$

Let $z(i, i) = 1$, $z(i, j) = 1$ for $i' - 2k + 3 \leq j \leq i' - k + 1, i' + 1 \leq j \leq i' + k - 2$, and $z(i, j) = k$ for $i' - k + 2 \leq j \leq i', j \neq i$. (See Fig. 1.) Clearly, conditions (α), (γ), (δ) are satisfied, and it is easy to check that (β) holds.

Step 1 requires $2k(1 + x(k - 1))$ calls, Step 2 requires $2(x - 2)$ calls, while Step 3 requires $(2k + x - 3)(k + 1) + [1 + x(k - 1) - (2k + x - 3)](k + 2)$ calls; adding, we obtain $(3k + 2)(1 + x(k - 1)) + x - 2k - 1 = (3k + 2) \cdot n/k^2 + x - 2k - 1 = c_k \cdot n - 2k - [k/(k - 1)]$. (In Step 3, we need $k + 1$ calls in the B_i 's for $1 \leq i \leq k, y - k + 1 \leq i \leq y$, and for $\{i: k + 1 \leq i \leq y - k \text{ and } k - 1 \nmid i - 1\}$. In the other groups, we use $k + 2$ calls.) $\square$

Remark 2.2. Instead of looking at the formal description, perhaps it is easier to visualize what is happening. In the case $x = 3$ (i.e., $y = 3k - 2$), only the entries described in (a), (b) are present. As x increases, we can obtain Z_x from the matrix Z_{x-1} by inserting the pattern of Fig. 2 over the lower right $(2k - 2) \times (2k - 2)$ principal submatrix of Z_{x-1} . (Again, see Fig. 1.)

This observation offers another way of counting the number of necessary calls. Suppose that the algorithm for n required $c_k \cdot n + d$ calls, for some constant d . The increment of n by $(k^3 - k^2)$ requires $2k(k - 1)$ additional calls in Step 1, 2 calls in Step

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	1	1	4	4	4	1	1												
2		4	1	4	4	1	1												
3			4	4	1	4	1	1											
4				4	4	4	1	1	1										
5					1	1	1	1		1	4	4	1	1					
6						1	1	1	1		4	1	4	1	1				
7											4	4	4	1	1				
8												1	1	1		1	4	4	1
9													4	1	4	1	1	1	
10														4	4	4	1	1	
11															1	1	1	1	4
12																4	1	4	1
13																	4	4	4
14																		1	1
15																			1
16																			
17																			
18																			
19																			

FIG. 1. Construction 2.1 in the case $k = 4$, $x = 6$.

2, and $1 \cdot (k + 1) + (k - 2)(k + 2)$ calls in Step 3. Altogether we use $c_k \cdot n + d + 3k^2 - k - 1 = c_k \cdot [n + (k^3 - k^2)] + d$ calls, hence d can be determined from the case $x = 3$. The same observation applies in the next three constructions. $\square$

CONSTRUCTION 2.3. For $n = k + x(k^3 - k^2)$, $x \geq 6$, there exists a good calling scheme using

$$(2.3.1) \quad c_k \cdot n - 2k - [(2k - 1)/(k^2 - k)]$$

calls.

Step 1. We divide the persons into $y = 1 + x(k - 1)$ groups, $A_1, A_2, \dots, A_y$; $|A_{2k-1}| = k$, and $|A_i| = k^2$ for each $1 \leq i \leq y$, $i \neq 2k - 1$. We perform one call in A_{2k-1} and $2k$ calls in the other groups s.t. everyone hears each gossip in his own group. Again, we refer to the information known in the i th group as the i th JG.

1	k	k	...	k	k	1	1	...	1	
k	1	k	...	k	k	1	1	...	1	
k	k	1	...	k	k	1	1	...	1	
...	...	...	...	...	...	...	...	...	...	
...	...	...	...	...	...	...	...	...	...	
...	...	...	...	...	...	...	...	...	...	
k	k	k	...	1	k	1	1	...	1	
k	k	k	...	k	k	1	1	...	1	
1	1	1	...	1	1					
1	1	1	...	1	1					
...	...	...	...	...	...					
...	...	...	...	...	...					
...	...	...	...	...	...					
1	1	1	...	1	1					

$\underbrace{\hspace{10em}}_{k-1}$
 $\underbrace{\hspace{10em}}_{k-2}$

FIG. 2

Step 2. We form the set Y as in Construction 2.1, but the sequence of calls is slightly different: among the a_i 's, we perform the calls $(a_k, a_{k+1}, a_{k+2}, \dots, a_{2k-1})$, $(a_1, a_2, \dots, a_k)$, $(a_1, a_{2k}, a_{2k+1}, \dots, a_{3k-2})$, $\dots$, $(a_1, a_{y-3(k-1)+1}, \dots, a_{y-2(k-1)})$; among the b_i 's, we perform the same sequence of calls as before:

$$(b_y, b_{y-1}, \dots, b_{y-(k-1)}),$$

$$(b_y, b_{y-k}, b_{y-k-1}, \dots, b_{y-2(k-1)}), \dots, (b_y, b_{3k-3}, \dots, b_{2k-1}),$$

in that order. After these calls, a_1 knows the JG's 1 to $y-2(k-1)$, a_i ($2 \leq i \leq k$) knows the JG's 1 to $2k-1$, a_i ($k+1 \leq i \leq 2k-1$) knows the JG's k to $2k-1$, and a_i ($2k \leq i \leq y-2(k-1)$) knows the JG's 1 to $1+(k-1) \cdot \lceil (i-1)/(k-1) \rceil$; b_y knows the JG's $2k-1$ to y , and b_i ($2k-1 \leq i \leq y-1$) knows the JG's $y-(k-1) \cdot \lceil (y-i)/(k-1) \rceil$ to y .

Step 3. We form groups $B_1, B_2, \dots, B_y$; $|B_{4k-4}| = k$, and $|B_i| = k^2$ for each $1 \leq i \leq y$, $i \neq 4k-4$. From Y , we send a_i ($2 \leq i \leq k$) into B_{i+3k-4} , a_i ($k+1 \leq i \leq 2k-1$) into B_{i+k-2} , a_i ($2k \leq i \leq y-2(k-1)$) into B_{i+2k-3} , and a_1 into B_y . On the other hand, we send b_i ($2k-1 \leq i \leq 4k-5$ or $6k-6 \leq i \leq y-1$) into $B_{i-(2k-3)}$, b_{4k-4} into B_{4k-4} , b_i ($4k-3 \leq i \leq 6k-7$) into $B_{i-(2k-2)}$, and b_y into B_1 . (The assignment of the b_i 's differs from that in Construction 2.1 in that b_{4k-4} was taken out of the original ordering, and everyone in between was shifted up by one group.)

The matrix Z_x can be defined on the following way:

- (a) $z(1, j) = k$ for $1 \leq j \leq k-1$, and $z(1, j) = 1$ for $k \leq j \leq 2k-2$.
If $2 \leq i \leq k$, then $z(i, j) = 1$ for $1 \leq j \leq k-1$, and $z(i, j) = k$ for $k \leq j \leq 2k-2$.
If $k+1 \leq i \leq 2k-2$, then $z(i, j) = 1$ for $1 \leq j \leq 2k-1$, and $z(i, j) = k$ for $2k \leq j \leq 3k-3$.
- (b) If $2k-1 \leq i \leq 3k-3$, then $z(i, 3k-2) = k$ and $z(i, j) = 1$, for all $2k \leq j \leq 4k-4$, $j \neq 3k-2$.
If $2k-1 \leq i \leq 3k-4$, then $z(i, i-2k+2) = z(i, i-2k+3) = 1$, and $z(i, j) = k$ for all $1 \leq j \leq k-1$, $j \neq i-2k+2$, $j \neq i-2k+3$.
 $z(3k-3, 1) = z(3k-3, k-1) = 1$, and $z(3k-3, j) = k$ for all $2 \leq j \leq k-2$.
- (c) If $3k-2 \leq i \leq 4k-5$, then $z(i, j) = 1$ for all $2k \leq j \leq 3k-2$, $4k-2 \leq j \leq 5k-5$, and $j = i+1$; moreover, $z(i, j) = k$ for all other values of j , satisfying $2k \leq j \leq 5k-5$.
 $z(4k-4, j) = 1$ for $2k \leq j \leq 3k-3$.
If $i \geq 4k-3$ then the i th row is the same as the i th row in Construction 2.1. (See Fig. 3.)

Again, it is easy to check that Z_x satisfies (α) – (δ) .

In the case $x = 6$, Step 1 requires $1 + 6(k-1)2k$ calls, Step 2 requires 8 calls, while Step 3 requires $(2k+1)(k+1) + 1 + (4k-7)(k+2)$ calls. Adding these numbers, we can determine the constant in (2.3.1). (See Remark 2.2.) $\square$

We shall need the following two lemmas in the third construction.

LEMMA 2.4. Suppose that two sequences of positive integers, $(p_1, p_2, \dots, p_k)$, and $(q_1, q_2, \dots, q_m)$ satisfy the following properties:

- (i) $p_1 + p_2 + \dots + p_k = q_1 + q_2 + \dots + q_m$;
- (ii) for some s , $1 \leq s \leq k$, $p_1 = p_2 = \dots = p_s$, $p_{s+1} = p_{s+2} = \dots = p_k$, and $p_{s+1} = p_s + 1$;
- (iii) for all $1 \leq i \leq m$, $q_i \leq k$.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	4	4	4	1	1	1													
2	1	1	1	4	4	4													
3	1	1	1	4	4	4													
4	1	1	1	4	4	4													
5	1	1	1	1	1	1	1	4	4										
6	1	1	1	1	1	1	1	4	4										
7	1	1	4					1	1	4	1	1							
8	4	1	1					1	1	4	1	1							
9	1	4	1					1	1	4	1	1							
10								1	1	1	1	4	4	1	1				
11								1	1	1	4	1	4	1	1				
12								1	1										
13											4	4	4	1	1				
14											1	1	1	4	4	1	1	1	1
15											1	1	1	4	4	1	1	1	1
16														1	1	1	4	4	4
17														1	1	4	1	4	4
18														1	1	4	4	1	4
19														1	1	4	4	4	1

FIG. 3. Construction 2.3 in the case $k = 4$, $x = 6$.

Then there exists a simple bipartite graph $G = (A \cup B, E)$ s.t. $|A| = k$, $|B| = m$, and $(p_1, p_2, \dots, p_k), (q_1, q_2, \dots, q_m)$ are the degree sequences of the vertices in A, B , respectively.

Proof. Let $A = \{a_1, a_2, \dots, a_k\}$, and $B = \{b_1, b_2, \dots, b_m\}$. We can construct G by the following algorithm: connect b_1 with $a_{s+1}, a_{s+2}, \dots, a_{s+q_1}$, b_2 with $a_{s+q_1+1}, a_{s+q_1+2}, \dots, a_{s+q_1+q_2}$, b_3 with $a_{s+q_1+q_2+1}, \dots, a_{s+q_1+q_2+q_3}$, etc. (The indices from A are taken mod k .) Properties (i), (ii) ensure that the algorithm produces the required degree sequences, while (iii) ensures that the graph is simple. $\square$

LEMMA 2.5. For each $k \geq 5$, $2 \leq m \leq k - 3$, there exists a $(2k - 3) \times (2k - 3)$ matrix M satisfying the following properties:

- (i) each entry of M is 0 or 1;
- (ii) the row-sums and column-sums in M are as follows:

$k - 4$	in $m - 1$ rows	$m - 1$	in $m - 2$ columns
$k - 3$	in $k - 2$ rows	m	in $k - m$ columns
$k - 2$	in 2 rows	$k - 1$	in 1 column
$m + 1$	in 1 row	$k - 3$	in $k - 2$ columns.
0	in $k - m - 3$ rows		

Proof. We construct a simple bipartite graph $G = (A \cup B, E)$ s.t. $|A| = |B| = 2k - 3$, and the degree sequences in A and B are the given row-sums and column-sums, respectively. M will be the adjacency matrix of G . Let $A = \{a_1, a_2, \dots, a_{2k-3}\}$ and $B = \{b_1, b_2, \dots, b_{2k-3}\}$. By Lemma 2.4, there exists a simple bipartite graph on $\{a_1, a_2, \dots, a_{k+m-3}\} \cup B$ with degree sequences $p_1 = p_2 = \dots = p_{m-1} = k - 4$, $p_m = p_{m+1} = \dots = p_{k+m-3} = k - 3$ and $q_1 = q_2 = \dots = q_{m-2} = m - 2$, $q_{m-1} = q_m = \dots = q_{k-2} = m - 1$, $q_{k-1} = k - 1$, $q_k = q_{k+1} = \dots = q_{k+m} = k - 5$, $q_{k+m+1} = q_{k+m+2} = \dots = q_{2k-3} = k - 4$, respectively. Connecting a_{k+m-2} with $b_1, b_2, \dots, b_{k-2}$, a_{k+m-1} with $b_k, b_{k+1}, \dots, b_{2k-3}$, and a_{k+m} with $b_k, b_{k+1}, \dots, b_{k+m}$, we obtain a bipartite graph with the desired properties. $\square$

CONSTRUCTION 2.6. For all $k \geq 3$, $2 \leq m \leq k - 1$, except the case $k = 4$, $m = 3$, if $n = k^2 - m(k^2 - k) + x(k^3 - k^2)$, $x \geq m + 7$, then there exists a good calling scheme by k -conference calls among n persons, using

$$c_k \cdot n - 2k - 1 + m - [(k + mk - m)/(k^2 - k)]$$

calls.

Step 1. We write m in the form $m = m_1 + m_2$, where $1 \leq m_i \leq \max \{1, k - 3\}$. (This decomposition of m is not possible if $k = 4$, $m = 3$; that is the reason we have to consider this case separately.) We divide the persons into $y = 1 + x(k - 1)$ groups, $A_1, A_2, \dots, A_y$. Let $|A_i| = k$, if $i = k + j(k - 1)$ for some $1 \leq j \leq m_1$, or $i = y - (k - 1) - j(k - 1)$ for some $1 \leq j \leq m_2$, and let $|A_i| = k^2$ for the remaining A_i 's. (Heuristically, we try to handle the groups of size k as we did the group A_{2k-1} in Construction 2.3. Since our method works with at most $\max \{1, k - 3\}$ groups of size k , we have to split m and apply the method at both ends.) We perform 1, or $2k$ calls in each group s.t. everyone hears each gossip in his own group. Again, we refer to the information known in the i th group as the i th JG.

Step 2. We form the set Y as in Constructions 2.1 and 2.3. Among the a_i 's, we perform the calls

$$\begin{aligned} &(a_k, a_{k+1}, a_{k+2}, \dots, a_{2k-1}), (a_k, a_{2k}, a_{2k+1}, \dots, a_{3k-2}), \dots, \\ &(a_k, a_{2+m_1(k-1)}, a_{3+m_1(k-1)}, \dots, a_{k+m_1(k-1)}), (a_1, a_2, \dots, a_k), \\ &(a_1, a_{2+(m_1+1)(k-1)}, a_{3+(m_1+1)(k-1)}, \dots, a_{k+(m_1+1)(k-1)}), \dots, \\ &(a_1, a_{y-3(k-1)+1}, \dots, a_{y-2(k-1)}); \end{aligned}$$

among the b_i 's, we perform the calls

$$\begin{aligned} &(b_{y-(k-1)}, b_{y-k}, b_{y-k-1}, \dots, b_{y-2(k-1)}), \\ &(b_{y-(k-1)}, b_{y-2(k-1)-1}, b_{y-2(k-1)-2}, \dots, b_{y-3(k-1)}), \dots, \\ &(b_{y-(k-1)}, b_{y-m_2(k-1)-1}, b_{y-m_2(k-1)-2}, \dots, b_{y-(m_2+1)(k-1)}), \\ &(b_y, b_{y-1}, \dots, b_{y-(k-1)}), \\ &(b_y, b_{y-(m_2+1)(k-1)-1}, b_{y-(m_2+1)(k-1)-2}, \dots, b_{y-(m_2+2)(k-1)}), \dots, \\ &(b_y, b_{3k-3}, \dots, b_{2k-1}), \end{aligned}$$

in that order. After these calls, a_1 knows the JG's 1 to $y - 2(k - 1)$, a_i ($2 \leq i \leq k$) knows the JG's 1 to $1 + (m_1 + 1)(k - 1)$, a_i ($k + 1 \leq i \leq 1 + (m_1 + 1)(k - 1)$) knows the JG's k to $1 + (k - 1) \cdot \lceil (i - 1)/(k - 1) \rceil$, and a_i ($2 + (m_1 + 1)(k - 1) \leq i \leq y - 2(k - 1)$) knows the JG's 1 to $1 + (k - 1) \cdot \lceil (i - 1)/(k - 1) \rceil$; b_y knows the JG's $2k - 1$ to y , b_i ($y - (k - 1) \leq i \leq y - 1$) knows the JG's $y - (m_2 + 1)(k - 1)$ to y , b_i ($y - (m_2 + 1)(k - 1) \leq i \leq y - k$) knows the JG's $y - (k - 1) \cdot \lceil (y - i)/(k - 1) \rceil$ to $y - (k - 1)$, and b_i ($2k - 1 \leq i \leq y - (m_2 + 1)(k - 1) - 1$) knows the JG's $y - (k - 1) \cdot \lceil (y - i)/(k - 1) \rceil$ to y .

Step 3. We form groups $B_1, B_2, \dots, B_y$;

$$\begin{aligned} &|B_{(m_1+3)(k-1)-m_1+1}| = |B_{(m_1+3)(k-1)-m_1+2}| = \dots = |B_{(m_1+3)(k-1)}| = k, \\ &|B_{y-(m_2+3)(k-1)+1}| = |B_{y-(m_2+3)(k-1)+2}| = \dots = |B_{y-(m_2+3)(k-1)+m_2}| = k, \end{aligned}$$

and $|B_i| = k^2$ for all other i 's. From Y , we send

$$\begin{aligned}
 a_1 & \text{ into } B_y, \\
 a_i \ (2 \leq i \leq k) & \text{ into } B_{i+(m_1+2)(k-1)-1}, \\
 a_i \ (k+1 \leq i \leq 1+(m_1+1)(k-1)) & \text{ into } B_{i+k-2}, \\
 a_i \ (2+(m_1+1)(k-1) \leq i \leq y-(m_2+5)(k-1)+1) & \text{ into } B_{i+2k-3}, \\
 a_i \ (y-(m_2+5)(k-1)+2 \leq i \leq y-(m_2+3)(k-1)) & \text{ into } B_{i+2k-3+m_2}, \\
 a_i \ (y-(m_2+3)(k-1)+1 \leq i \leq y-(m_2+3)(k-1)+m_2) & \text{ into } B_i, \\
 a_i \ (y-(m_2+3)(k-1)+m_2+1 \leq i \leq y-2(k-1)) & \text{ into } B_{i+2k-3}.
 \end{aligned}$$

On the other hand, we send

$$\begin{aligned}
 b_y & \text{ into } B_1, \\
 b_i \ (y-(k-1) \leq i \leq y-1) & \text{ into } B_{i-(m_2+2)(k-1)+1}, \\
 b_i \ (y-(m_2+1)(k-1) \leq i \leq y-(k-1)-1) & \text{ into } B_{i-(k-2)}, \\
 b_i \ ((m_1+5)(k-1) \leq i \leq y-(m_2+1)(k-1)-1) & \text{ into } B_{i-(2k-3)}, \\
 b_i \ ((m_1+3)(k-1)+1 \leq i \leq (m_1+5)(k-1)-1) & \text{ into } B_{i-(2k-3)-m_1}, \\
 b_i \ ((m_1+3)(k-1)-m_1+1 \leq i \leq (m_1+3)(k-1)) & \text{ into } B_i, \\
 b_i \ (2k-1 \leq i \leq (m_1+3)(k-1)-m_1) & \text{ into } B_{i-(2k-3)}.
 \end{aligned}$$

We define Z_x in the case $x = m + 7$. For greater values of x , Z_x can be obtained by inserting the pattern of Fig. 2. over the lower right $(m_2 + 4)(k - 1) \times (m_2 + 4)(k - 1)$ principal submatrix of Z_{x-1} .

If $m_1 = 1$, then the first $(m_1 + 3)(k - 1) = 4k - 4$ rows of Z_{m+7} are the same as in Construction 2.3. If $2 \leq m_1 \leq k - 3$, then we define the first $(m_1 + 3)(k - 1)$ rows of Z_{m+7} the following way:

(a) The first $2k - 2$ rows are the same as in Construction 2.3.

(b) If $2k - 1 \leq i \leq 1 + m_1(k - 1)$, and $i = k + s(k - 1)$ for some $1 \leq s \leq m_1 - 1$, then $z(i, s) = k$, and $z(i, j) = 1$ for all $1 \leq j \leq k - 1, j \neq s$. Moreover, $z(i, j) = k$ for all $i + 1 \leq j \leq i + k - 2$.

If $2k - 1 \leq i \leq 1 + m_1(k - 1)$, and $k - 1 \nmid i - 1$, then let

$$i' = (k - 1) \cdot \lceil (i - 1) / (k - 1) \rceil + 1.$$

$z(i, j) = 1$ for $1 \leq j \leq k - 1, i' - k + 2 \leq j \leq i'$, and $z(i, j) = k$ for $i' + 1 \leq j \leq i' + k - 2$.

(c) $z(2 + m_1(k - 1), j) = 1$ for $1 \leq j \leq m_1 - 1, 2 + m_1(k - 1) \leq j \leq m_1(k - 1) + 2k - m_1$; $z(2 + m_1(k - 1), j) = k$ for $m_1 \leq j \leq k - 1, m_1(k - 1) + 2k - m_1 + 1 \leq j \leq (m_1 + 2)(k - 1)$.

(d) If $3 + m_1(k - 1) \leq i \leq (m_1 + 1)(k - 1)$, then let $z(i, j) = 1$ for $1 \leq j \leq k - 1, 2 + m_1(k - 1) \leq j \leq 1 + (m_1 + 1)(k - 1)$; moreover, if $3 + m_1(k - 1) \leq i \leq m_1(k - 1) + k - m_1$, then let $z(i, j) = k$ for $2 + (m_1 + 1)(k - 1) \leq j \leq (m_1 + 2)(k - 1)$.

(e) If $1 + (m_1 + 1)(k - 1) \leq i \leq (m_1 + 1)(k - 1) + k - m_1 - 3$, then let $z(i, j) = 1$ for $j = i - (m_1 + 1)(k - 1)$, and $z(i, j) = k$ for all other values $1 \leq j \leq k - 1$.

Let $z((m_1 + 1)(k - 1) + k - m_1 - 2, j) = k$ for $1 \leq j \leq k - m_1 - 3$, and

$$z((m_1 + 1)(k - 1) + k - m_1 - 2, j) = 1$$

for $k - m_1 - 2 \leq j \leq k - 1$.

If $(m_1 + 1)(k - 1) + k - m_1 - 1 \leq i \leq (m_1 + 2)(k - 1)$, then $z(i, j) = 1$ for all $1 \leq j \leq k - 1$.

(f) If $(m_1 + 2)(k - 1) - m_1 + 2 \leq i \leq (m_1 + 3)(k - 1) - m_1$, then let

$$z(i, (m_1 + 3)(k - 1) + 1) = k,$$

and $z(i, j) = 1$ for all $(m_1 + 3)(k - 1) + 2 \leq j \leq (m_1 + 4)(k - 1)$.

If $(m_1 + 3)(k - 1) - m_1 + 1 \leq i \leq (m_1 + 3)(k - 1)$, then let $z(i, j) = 1$ for all $2 + (m_1 + 1)(k - 1) \leq j \leq (m_1 + 2)(k - 1)$.

We define Z_{m+7} in the $(2k - 3) \times (2k - 3)$ submatrix Z^* given by the rows

$$(m_1 + 1)(k - 1) - m_1 + 2 \leq i \leq (m_1 + 3)(k - 1) - m_1,$$

and columns $2 + (m_1 + 1)(k - 1) \leq j \leq (m_1 + 3)(k - 1)$ later.

The next k rows of Z_{m+7} are defined as follows:

$$z((m_1 + 3)(k - 1) + 1, j) = k \quad \text{for } (m_1 + 2)(k - 1) + 2 \leq j \leq (m_1 + 3)(k - 1) + 1, \quad \text{and}$$

$$z((m_1 + 3)(k - 1) + 1, j) = 1 \quad \text{for } (m_1 + 3)(k - 1) + 2 \leq j \leq (m_1 + 4)(k - 1).$$

If $(m_1 + 3)(k - 1) + 2 \leq i \leq (m_1 + 4)(k - 1)$, then let $z(i, j) = 1$ for

$$(m_1 + 2)(k - 1) + 2 \leq j \leq (m_1 + 3)(k - 1) + 1,$$

$(m_1 + 4)(k - 1) + 1 \leq j \leq (m_1 + 5)(k - 1)$; let $z(i, j) = k$ for $(m_1 + 3)(k - 1) + 2 \leq j \leq (m_1 + 4)(k - 1)$,

$$z((m_1 + 4)(k - 1) + 1, j) = 1 \quad \text{for } (m_1 + 3)(k - 1) + 2 \leq j \leq (m_1 + 4)(k - 1), \quad \text{and}$$

$$z((m_1 + 4)(k - 1) + 1, j) = k \quad \text{for } (m_1 + 4)(k - 1) + 1 \leq j \leq (m_1 + 5)(k - 1).$$

Now we are ready to define the entries of Z^* . If we choose each entry in Z^* to be equal to 1, then the row-sums and column-sums in these rows and columns will be less than $|B_i \setminus Y|$, $|A_j \setminus Y|$ by the following values:

$(k - 3)(k - 1)$	in $(m - 1)$ rows	$m(k - 1)$	in $(k - m)$ columns
0	in $(k - m - 3)$ rows	$(m - 1)(k - 1)$	in $(m - 2)$ columns
$(m + 1)(k - 1)$	in one row	$(k - 1)(k - 1)$	in one column
$(k - 2)(k - 1)$	in 2 rows	$(k - 3)(k - 1)$	in $(k - 2)$ columns.
$(k - 4)(k - 1)$	in $(m - 1)$ rows		
$(k - 3)(k - 1)$	in $(k - 1 - m)$ rows		

Hence, adding $(k - 1)M^*$ to Z^* , where M^* is obtained by an appropriate permutation of rows and columns in the matrix M constructed in Lemma 2.5, we can achieve that (α) , (β) hold for these rows and columns.

Finally, we have to define the last $(m_2 + 3)(k - 1)$ rows of Z_{m+7} . We define the first $(m_2 + 3)(k - 1)$ rows of a matrix Z' by the exact rules used when we defined the first $(m_1 + 3)(k - 1)$ rows of Z_{m+7} . The only change is that we replace each occurrence of m_1 by m_2 . After that, let $z(y + 1 - i, y + 1 - j) = z'(i, j)$, where $1 \leq i \leq (m_2 + 3)(k - 1)$, and $z'(i, j)$ denotes the entries of Z' . (See Fig. 4.) It is possible to check that Z_{m+7} satisfies (α) – (δ) . $\square$

A_i 's. We perform 48 resp. 1 resp. 8 calls in each group s.t. everyone hears each gossip in his own group.

Step 2. We form the set Y as before. Among the a_i 's, we perform the calls (a_4, a_5, a_6, a_7) , (a_4, a_8, a_9, a_{10}) , $(a_4, a_{11}, a_{12}, a_{13})$, (a_1, a_2, a_3, a_4) , $(a_1, a_{14}, a_{15}, a_{16})$, $(a_1, a_{17}, a_{18}, a_{19})$, $\dots$, $(a_1, a_{y-8}, a_{y-7}, a_{y-6})$; among the b_i 's, we perform the calls $(b_y, b_{y-1}, b_{y-2}, b_{y-3})$, $(b_y, b_{y-4}, b_{y-5}, b_{y-6})$, $\dots$, (b_y, b_9, b_8, b_7) , in that order. After these calls, a_1 knows the JG's 1 to $y-6$, a_i ($2 \leq i \leq 4$) knows the JG's 1 to 13, a_i ($5 \leq i \leq 13$) knows the JG's 4 to $1 + 3 \cdot \lceil (i-1)/3 \rceil$, and a_i ($14 \leq i \leq y-6$) knows the JG's 1 to $1 + 3 \cdot \lceil (i-1)/3 \rceil$; b_y knows the JG's 7 to y , and b_i ($7 \leq i \leq y-1$) knows the JG's $y - 3 \cdot \lceil (y-i)/3 \rceil$ to y .

Step 3. We form groups $B_1, B_2, \dots, B_y$; $|B_{11}| = |B_{12}| = |B_{13}| = 64$, $|B_{16}| = |B_{17}| = |B_{18}| = 4$, and $|B_i| = 16$ for all of the remaining B_i 's. We send a_i ($2 \leq i \leq 4$) into B_{i+14} , a_i ($5 \leq i \leq 13$) into B_{i+2} , a_i ($14 \leq i \leq y-6$) into B_{i+5} , and a_1 into B_y . On the other hand, we send b_i ($7 \leq i \leq 15$) into B_{i-5} , b_i ($16 \leq i \leq 23$) into B_{34-i} , b_i ($24 \leq i \leq y-1$) into B_{i-5} , and b_y into B_1 .

At the definition of Z_x , we replace (δ) by the following condition:

(δ') $z(i, j) \in \{1, k, k^2\}$ if none of the persons in $B_i \cap Y$ knows the j th JG after Step 2.

Clearly, (δ') also ensures that we can finish the algorithm in each new group. In the case $x = 11$, Z_x is defined by Fig. 5. For greater values of x , Z_x can be obtained by inserting the pattern of Fig. 2. over the lower right 6×6 principal submatrix of Z_{x-1} . $\square$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	4	4	4	1	1	1																			
2	1	1	1	4	4	4																			
3	1	1	1	4	4	4																			
4	1	1	1	4	4	4																			
5	1	1	1	1	1	1	1	4	4																
6	1	1	1	1	1	1	1	4	4																
7	4	1	1					4	4																
8	1	1	1					1	1	1	4	4													
9	1	1	1					1	1	1	4	4													
10	4	1	1								4	4													
11	4	16	16								1	1	1	4	4	1	4	4	4	1	1				
12	16	16	16								1	1	1	1	1	1	1	1	4	1	1				
13	16	16	16											4	4	4	1	1							
14	4	1	1											1	1	4	1	1							
15	4	1	1											1	1	4	1	1							
16														1	1										
17														1	1										
18														1	1										
19																4	4	4	1	1					
20																1	1	1	4	4	1	1	1	1	
21																1	1	1	4	4	1	1	1	1	
22																		1	1	1	4	4	4	4	
23																		1	1	4	1	4	4		
24																		1	1	4	4	1	4		
25																		1	1	4	4	4	1		

FIG. 5. Construction 2.7 in the case $x = 11$.

REFERENCES

- [1] B. BAKER AND R. SHOSTAK, *Gossips and telephones*, Discrete Math., 2 (1972), pp. 191–193.
- [2] J.-C. BERMOND, *Le problème des "ouvoirs", (hypergraph gossip problem)*, in Problèmes Comb. et Theory des Graphs, J.-C. Bermond, ed., C.N.R.S. Colloq., Paris, 1976.
- [3] R. BUMBY, *A problem with telephones*, SIAM J. Algebraic Discrete Methods, 2 (1981), pp. 13–18.
- [4] A. HAJNAL, E. MILNER AND E. SZEMERÉDI, *A cure for the telephone disease*, Canad. Math. Bull., 15 (1972), pp. 447–450.
- [5] D. KLEITMAN AND J. SHEARER, *Further gossip problems*, Discrete Math., 30 (1980), pp. 151–156.
- [6] K. LEBENSOLD, *Efficient communication by phone calls*, Stud. Appl. Math., 52 (1973), pp. 345–358.
- [7] D. MIKLÓS, M. NEWMAN, Á. SERESS AND D. WEST, *The addition game: An abstraction of a communication problem*, Discrete Math., to appear.
- [8] Á. SERESS, *Gossiping old ladies*, Discrete Math., 46 (1983), pp. 75–81.
- [9] ———, *Gossips by conference calls*, Studia Sci. Math. Hungar, to appear.
- [10] ———, *Quick gossiping without duplicate transmissions*, Graphs and Combinatorics, 2 (1986), pp. 363–381.
- [11] ———, *Quick gossiping without duplicate transmissions*, extended abstract, Proc. Third International Conference on Combinatorial Mathematics, Annals of the New York Academy of Sciences, submitted.
- [12] R. TIJDEMAN, *On a telephone problem*, Nieuw Arch. Wisk., 19 (1971), pp. 188–192.
- [13] D. WEST, *Gossiping without duplicate transmissions*, SIAM J. Algebraic Discrete Methods, 3 (1982), pp. 418–419.

SOME APPLICATIONS OF AFFINE GALE DIAGRAMS TO POLYTOPES WITH FEW VERTICES*

BERND STURMFELS†

Abstract. Affine Gale diagrams are of one dimension lower than the well-known Gale transforms, and thus k -polytopes with $k + 4$ vertices can be represented by planar point configurations. The underlying algebraic reduction is due to Bokowski [6], while similar geometric arguments were used before by Perles [12]. In this paper we consider affine Gale diagrams as a special case of oriented matroid duality, and we apply this technique to several convex geometrical problems. As the main result we establish a new negative Steinitz-type theorem in the spirit of [25]; the face lattices of simplicial k -polytopes with $k + 4$ vertices cannot be characterized locally. We answer two questions posed in [11] concerning Kleinschmidt's 4-polytope Q with a facet of nonarbitrary shape [15], and we describe another such 4-polytope P with minimal number of facets. We characterize the affine Gale diagrams corresponding to a given simplicial complex, and we discuss as an example Möbius' torus with 7 vertices. Finally, we prove a partial result on Perles' problem whether all simplicial polytopes are quotients of neighborly polytopes.

Key words. affine Gale diagram, Gale transform, polytope, oriented matroid, chirotope, duality

AMS(MOS) subject classifications. 52A25, 05B35

1. Introduction. Duality and polarity are among the most fundamental tools in combinatorial convexity. A (convex) k -polytope P with n vertices and a k -polytope P^* with n facets are called *dual* (or *polar*) to each other if the face lattice of P is antiisomorphic to the face lattice of P^* . The corresponding duality operation $*$ can be realized geometrically by a *polarity* of the ambient vector space $\mathbf{R}^k$.

In this paper, however, we shall be dealing with a different type of duality which is less obvious to visualize, namely the duality of *Gale transforms* or *oriented matroids*. In that sense the object dual to a k -polytope P with n vertices is a configuration of n vectors in $(n - k - 1)$ -dimensional space which contains the entire information about P . Thus Gale transforms or *Gale diagrams*, a normalized variant, can be used to study the combinatorial properties of polytopes with few vertices in low-dimensional diagrams; for example, a k -polytope with $k + 3$ vertices admits a Gale diagram in the plane.

Many important results on k -polytopes with up to $k + 3$ vertices have been obtained by means of Gale transforms. Let us mention in particular the achievements of Perles [12, Chap. 6], Mani [18], and Kleinschmidt [14], which yield a complete classification of all (combinatorial types of) k -polytopes with up to $k + 3$ vertices. The exhaustive enumeration of k -polytopes with n vertices for $n \geq k + 4 \geq 8$, called the *Steinitz problem*, seems to be a rather hopeless enterprise, as is also indicated by our results in § 4. Let us also mention the study of toric varieties in algebraic geometry as one interesting recent application of Gale transforms (see Ewald [10]).

The classical axiomatization of oriented matroids in terms of their *circuits* and *cocircuits*, given by Las Vergnas in 1975 [16] (see also [2], [4]) was motivated mainly by questions arising in the context of optimization and network theory. For applications in convexity, however, we prefer an alternative but equivalent approach to oriented matroids using oriented bases or *chirotopes* [8]. To convince ourselves that the theory of oriented matroids provides a natural framework for duality in combinatorial geometry, let us recall the definition of Gale transforms.

* Received by the editors March 23, 1987; accepted for publication (in revised form) October 2, 1987. This research was supported in part by an Alfred P. Sloane Doctoral Dissertation Fellowship.

† Institute for Mathematics and its Applications, University of Minnesota, Minneapolis, Minnesota 55455.

Given a k -polytope P with set V of n vertices, consider the $(n - k - 1)$ -dimensional vector subspace $L(V) \subset \mathbf{R}^n$ of affine dependencies on V . A Gale transform $\bar{V}$ of V is a set of n vectors in $\mathbf{R}^{n-k-1}$ such that its space of linear dependencies $L(\bar{V})$ is the orthogonal complement to $L(V)$ in $\mathbf{R}^n$. The affine $(k + 1)$ -chirotope χ of V reflects the situation as follows. The circuits of χ are the signed elementary vectors of $L(V)$, while the cocircuits of χ are the signed elementary vectors of $L(V)^\perp$. This means that any linear realization of the dual chirotope χ^* is (isomorphic to) a Gale transform of the vertex set V .

In a survey article from 1978 McMullen [19] mentions this close relationship between Gale transforms and oriented matroids. However, we know of no publication where this connection is worked out in detail, and so we begin with a translation of the basic properties of Gale transforms into the language of chirotopes which then leads to the definition of affine Gale diagrams.

The underlying geometric idea has been used already by Perles in his construction of a nonrational 8-polytope [12, Thm. 5.5.3]. Bokowski developed the affine Gale diagram technique algebraically in connection with his work on neighborly 6-polytopes with 10 vertices [6], and he proved that the question whether an arbitrary $(k - 1)$ -sphere with $k + 4$ vertices is polytopal reduces to a planar problem (Thm. 3.2i, see also [21]). (The author wishes to thank J. Bokowski for making these unpublished results available to him as well as for many helpful discussions.)

The case of $k + 4$ vertices is of particular interest for the theory of convex polytopes because several well-known three-dimensional results fail for polytopes of dimension $k > 3$ with $n \geq k + 4$ vertices. In particular, there are nonpolytopal spheres and diagrams [11], [12], nonrational polytopes [12, Thm. 5.5.3], [25], and polytopes with nonarbitrary facets [15]. We hope that our presentation of affine Gale diagrams will provide some intuitive geometrical evidence for why “ $k + 4$ ” is a “threshold of counterexamples.”

Naturally, it is our major goal to apply our combinatorial methods to gain new insights and derive new results in geometry. In § 4 we construct an infinite family of minor-minimal nonpolytopal triangulated $(k - 1)$ -spheres with $k + 4$ vertices (k not fixed). This result is an analogue to a recent negative Steinitz-type theorem of the author on 6-polytopes [25]: the face lattices of simplicial polytopes with few vertices cannot be characterized locally.

In [15] Kleinschmidt proved that a particular property of 3-polytopes does not carry over to higher dimensions. There exists a 4-polytope Q with a facet of nonarbitrary shape. Using affine Gale diagrams we give a new proof for this property of Q , and we describe another such 4-polytope P with minimal number of facets. Moreover, we shall answer two questions posed in Ewald et al. [11] concerning the polytope Q .

In § 6 we derive a characterization for affine Gale diagrams corresponding to imbedded (simplicial) complexes, showing that affine Gale diagrams are useful not only for convex polytopes but also for various related questions in combinatorial convex geometry. As an application we discuss the well-known fact that the *Möbius torus* or *Császár torus* with 7 vertices admits an imbedding without self-intersections into Euclidean 3-space. For new results on that subject see Bokowski and Eggert [5].

In the last section we turn to a problem of M. A. Perles, communicated to the author by G. Kalai, asking whether every simplicial polytope can be represented as the quotient of some neighborly polytope. It is known that for $\beta \leq 3$ every simplicial d -polytope with $d + \beta$ vertices is the quotient of a cyclic (hence neighborly) polytope while this property fails to hold for all $\beta \geq 4$ [1]. Using sign-invariant pairs in oriented matroids, we prove that every simplicial d -polytope with n vertices is the quotient of a $(d + 1)$ -neighborly $(n + d)$ -polytope with $2n$ vertices.

2. Oriented matroid duality and affine Gale diagrams. The basic definitions and fundamental results on oriented matroids can be found in Bachem [2] and the papers quoted there. See Las Vergnas [17] for convexity in oriented matroids and the books of Grünbaum [12] and McMullen and Shephard [20] for introductions to Gale transforms and convex polytopes in general. Let us recall some basic definitions and properties on the chirotope representation of oriented matroids.

$\Lambda(n, d) := \{(\lambda_1, \dots, \lambda_d) \in \mathbb{N}^d \mid 1 \leq \lambda_1 < \dots < \lambda_d \leq n\}$ denotes the set of ordered d -tuples of n elements. Sometimes $\lambda \in \Lambda(n, d)$ will be identified with the set $\lambda = \{\lambda_1, \dots, \lambda_d\}$ as well. A mapping $\chi: \Lambda(n, d) \rightarrow \{-1, 0, +1\}$, alternatingly extended to all of $\{1, 2, \dots, n\}^d$, is a d -chirotope on $E := \{1, 2, \dots, n\}$ if for all $\mu \in \Lambda(n, d+2)$ there exist coordinate vectors $x_{\mu_1}, \dots, x_{\mu_{d+2}} \in \mathbb{R}^d$ such that

$$\chi(\lambda) = \text{sign}(\det(x_{\lambda_1}, \dots, x_{\lambda_d})) \quad \text{for all } \lambda \in \Lambda(n, d) \text{ with } \lambda \subset \mu.$$

A chirotope χ is *realizable* if there exist (global coordinates) $x_1, \dots, x_n \in \mathbb{R}^d$ such that

$$\chi(\lambda) = \text{sign}(\det(x_{\lambda_1}, \dots, x_{\lambda_d})) \quad \text{for all } \lambda \in \Lambda(n, d).$$

In this case χ is called the *linear chirotope* of the point configuration $\{x_1, \dots, x_n\}$. The *affine chirotope* of $\{y_1, \dots, y_n\} \subset \mathbb{R}^k$ is the linear chirotope of $\{(1, y_1), \dots, (1, y_n)\} \subset \mathbb{R}^{k+1}$. χ is *simplicial* if $\text{Im } \chi \subset \{-1, +1\}$. The *cocircuits* of χ are the signed vectors

$$\pm C_\mu(\chi) := \pm \begin{pmatrix} \chi(\mu_1, \dots, \mu_{d-1}, 1) \\ \vdots \\ \chi(\mu_1, \dots, \mu_{d-1}, i) \\ \vdots \\ \chi(\mu_1, \dots, \mu_{d-1}, n) \end{pmatrix} \in \{-1, 0, +1\}^n \quad \text{for } \mu \in \Lambda(n, d-1).$$

The *dual chirotope* χ^* of a d -chirotope with n vertices is defined as

$$\begin{aligned} \chi^*: \Lambda(n, n-d) &\rightarrow \{-1, 0, +1\} \\ \lambda &\mapsto \chi(\{1, \dots, n\} \setminus \lambda) * (-1)^{\sum_{i=1}^{n-d} \lambda_i}. \end{aligned}$$

A *circuit* of a chirotope χ is a cocircuit of its dual.

For any signed vector $C \in \{-1, 0, +1\}^n$ we define $C^+ := \{i \mid C(i) = +1\}$, and similarly, C^0 and C^- . $A \subset E$ is a *halfspace* of χ if there exists a cocircuit C with $C^+ = A$. With $\mathcal{H}(\chi)$ we denote the *signed circuit span* of χ . That is the set of all signed vectors which are conformal unions of circuits of χ (see [2]). Likewise $\mathcal{H}(\chi^*)$ is the *signed cocircuit span* of χ . For a subset $A \subset E$ the *reorientation* on A of χ is denoted by $\bar{A}\chi$ [4].

Given a k -polytope $P = \text{conv}\{y_1, \dots, y_n\} \subset \mathbb{R}^k$, we assign to P the affine $(k+1)$ -chirotope χ of $\{y_1, \dots, y_n\}$. If $F = \text{conv}\{y_{i_1}, \dots, y_{i_l}\}$ is a facet of P , then all vertices y_j of P that are not in F lie on the same side of the hyperplane affinely spanned by F . This is equivalent to the existence of a positive cocircuit $C = C^+$ with $C^0 = \{i_1, \dots, i_l\}$.

Hence the following definition generalizes the concept of face lattices of convex polytopes: $F \subset E$ is a *facet* of the d -chirotope χ if there is a positive cocircuit of χ with $C^0 = F$. Finite intersections of facets are called *faces* of χ , and the set of all faces of χ , ordered by inclusion, forms the *face lattice* $\text{FL}(\chi)$ of χ . Clearly, a combinatorial sphere S is polytopal if and only if there exists a *realizable* chirotope χ with $\text{FL}(\chi) = S$.

Remark 2.1. If two polytopes P and $\tilde{P}$ have the same chirotope, then they are combinatorially equivalent.

An octahedron with vertices in general position and a regular octahedron show that the converse is false. Given a $(d-1)$ -polytope P with set of vertices

$$V = \{x_1, \dots, x_n\} \subset \mathbf{R}^{d-1},$$

consider the $n \times d$ -matrix

$$M_p = \begin{pmatrix} 1 & x_1^t \\ \vdots & \vdots \\ 1 & x_n^t \end{pmatrix}.$$

A configuration $\bar{V} = \{\bar{x}_1, \dots, \bar{x}_n\} \subset \mathbf{R}^{n-d}$ is called a *Gale transform* of V if the column space of the $n \times (n-d)$ -matrix $(\bar{x}_1, \dots, \bar{x}_n)^t$ is the orthogonal complement in $\mathbf{R}^n$ of the column space of M_p .

A *Gale diagram* is a standardized version of a Gale transform. Let $\bar{V} = \{\bar{x}_1, \dots, \bar{x}_n\}$ be a Gale transform of a set V of points in $\mathbf{R}^{d-1}$. We assign to $\bar{V}$ the corresponding Gale diagram $\hat{V} := \{\hat{x}_1, \dots, \hat{x}_n\}$ by normalizing all vectors:

$$\hat{x}_i := \begin{cases} 0 & \text{if } \bar{x}_i = 0, \\ \bar{x}_i / |\bar{x}_i| & \text{else.} \end{cases}$$

Since the linear chirotopes of $\bar{V}$ and $\hat{V}$ are equal, we will not further discuss Gale diagrams. Note only that each of the following statements remains true if we substitute “Gale transform” with “Gale diagram.”

Remark 2.2. Let V be a spanning set of points in $\mathbf{R}^{d-1}$ and $\bar{V} \subset \mathbf{R}^{n-d-1}$ a Gale transform of V . Then the affine d -chirotope of V and the linear $(n-d)$ -chirotope of $\bar{V}$ are dual to each other.

Example 2.3. ($n = 6, d = 4$). Let $\mathcal{O} = \text{conv}(V) = \text{conv}\{x_1, \dots, x_6\} \subset \mathbf{R}^3$ be the regular octahedron with vertices

$$\begin{aligned} x_1 &= \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, & x_2 &= \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix}, & x_3 &= \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \\ x_4 &= \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix}, & x_5 &= \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, & x_6 &= \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}. \end{aligned}$$

A Gale transform of V is given by $\bar{V} = \{\bar{x}_1, \dots, \bar{x}_6\} \subset \mathbf{R}^2$

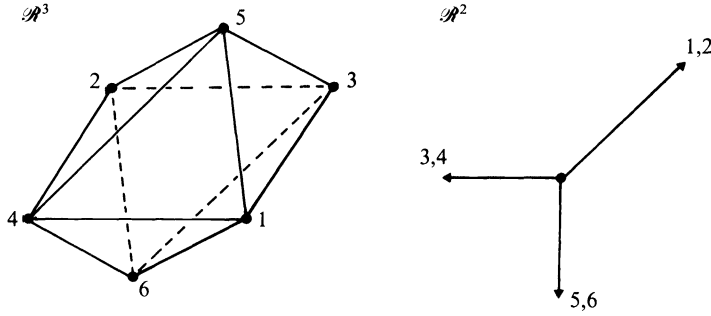
$$\begin{aligned} \bar{x}_1 &= \begin{pmatrix} 1 \\ 1 \end{pmatrix}, & \bar{x}_2 &= \begin{pmatrix} 1 \\ 1 \end{pmatrix}, & \bar{x}_3 &= \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \\ \bar{x}_4 &= \begin{pmatrix} -1 \\ 0 \end{pmatrix}, & \bar{x}_5 &= \begin{pmatrix} 0 \\ -1 \end{pmatrix}, & \bar{x}_6 &= \begin{pmatrix} 0 \\ -1 \end{pmatrix}. \end{aligned}$$

See Fig. 1.

We recall the fundamental properties of Gale transforms as stated in [12, Chap. 5.4] or [20, Chap. 3]. Given a set $X = \{x_1, \dots, x_n\} \subset \mathbf{R}^{d-1}$ and a subset $I \subset \{1, \dots, n\}$ of indices, we write $X(I) := \{x_i \in X \mid i \in I\}$ for the corresponding set of vectors.

THEOREM 2.4.

- (i) $X(I)$ is a coface of X if and only if $I = \emptyset$ or $0 \in \text{rel int}(\text{conv } \bar{X}(I))$.
- (ii) Assume that a finite set X affinely spans $\mathbf{R}^{d-1}$. Then X is the set of vertices of a $(d-1)$ -polytope P if and only if either


 FIG. 1. A Gale transform of a regular octahedron $\mathcal{O}$.

- (1) $\bar{X} = \{0\}$ in which case P is a $(d-1)$ -simplex or
- (2) every open linear halfspace H^+ of $\mathbb{R}^{n-d}$ satisfies $|\{i | \bar{x}_i \in H^+\}| \geq 2$.
- (iii) $P = \text{conv } X$ is a simplicial polytope if and only if $\dim \bar{X}(I) = n-d$ for every cofacet $X(I)$ of P .
- (iv) Let $\hat{X} = \{\hat{x}_1, \dots, \hat{x}_{n-1}, \hat{x}_n\} \subset \mathbb{R}^{n-d}$ be a Gale diagram of $X \subset \mathbb{R}^{d-1}$. Let H be a hyperplane which strictly separates x_n from $\{x_1, \dots, x_{n-1}\}$, and define $y_i := H \cap [x_i, x_n]$ for $1 \leq i \leq n-1$. Then $\hat{Y} := \{\hat{x}_1, \dots, \hat{x}_{n-1}\}$ is a Gale diagram of $Y := \{y_1, \dots, y_{n-1}\} \subset H$. In particular, if X is the set of vertices of a polytope P , and if all $[x_i, x_n]$, $1 \leq i \leq n-1$, are edges of P , then P is a Gale diagram of the vertex figure $H \cap P$ of P at x_n .

Let us translate Theorem 2.4 into the combinatorial language of chirotopes.

PROPOSITION 2.5. Let χ be an affine d -chirotope with n vertices.

- (i) $I \subset \{1, \dots, n\}$ is a coface of $\text{FL}(\chi)$ if and only if either $I = \emptyset$ or there is an element C in the signed circuit span $\mathcal{K}(\chi^*)$ of χ^* with $C^+ = I$ and $C^- = \emptyset$.
- (ii) Every vertex of χ is extreme, i.e., χ/i is affine for all $i \in \{1, \dots, n\}$, if and only if

- (1) $\text{rk } \chi^* = 0$ or
- (2) for every halfspace A of χ^* : $|A| \geq 2$.
- (iii) $\text{FL}(\chi)$ is simplicial if and only if $\text{rk}_{\chi^*}(I) = n-d$ for every cofacet I of χ .
- (iv) We have the duality relation $\chi/n = (\chi^* \setminus n)^*$ for contractions and deletions. If χ is the affine chirotope of a polytope $P = \text{conv } \{x_1, \dots, x_n\}$, then $\text{FL}(\chi/n)$ is the face lattice of the vertex figure of P at x_n .

Proof.

(i) An element $C \in \mathcal{K}(\chi^*)$ satisfies $C^- = \emptyset$ if and only if C is the union of positive cocircuits of χ . By definition, faces of χ are intersections of facets which are complements of positive cocircuits of χ . Hence the cofaces of χ are exactly the unions of positive circuits of χ^* .

(ii) If $\text{rk } \chi^* = 0$, then χ is clearly a simplex. Assume that $\text{rk } \chi^* \geq 1$ and every halfspace of χ^* contains at least two elements. In particular, for any i there is no cocircuit C of χ^* with $C^+ = \{i\}$. This means that there is no circuit C of χ with $C^+ = \{i\}$. By Minty's lemma [2] we conclude the existence of a cocircuit D of χ with $D^+ = \{i\}$, and therefore i is an extreme vertex by [17, Prop. 1.6]. Conversely, if every vertex of χ is extreme, then no circuit C of χ with $|C^+| \leq 1$ can exist. Hence every cocircuit C of χ^* satisfies $|C^+| \geq 2$.

(iii) A facet F of χ is a simplex if and only if it is contained in a basis of the underlying matroid of χ . This means that the corresponding cofacet F^c contains a basis of the dual χ^* , i.e., $\text{rk}_{\chi^*}(F^c) = n-d$.

- (iv) See Las Vergnas [17, p. 233]. $\square$

It follows from the discussion above that for a given polytope P with affine chirotope χ any realization of χ^* can be used to read off the face lattice of P . This motivates the following.

DEFINITION 2.6. Let $V = \{x_1, x_2, \dots, x_n\} \subset \mathbb{R}^{d-1}$ and χ the corresponding affine chirotope on $E = \{1, 2, \dots, n\}$. An *affine Gale diagram* of V (or χ) is a pair (Π, A) , where Π is an affine $(n - d)$ -chirotope on E , $A \subset E$, and $_{\bar{A}}\Pi = \chi^*$.

Example 2.7. (See Fig. 2(b).) Let Π be the affine 2-chirotope of $\{x_1, \dots, x_6\} \subset \mathbb{R}_1$ where $x_1 = x_2 < x_3 = x_4 < x_5 = x_6$. Then the pair $(\Pi, \{3, 4\})$ forms an affine Gale diagram for the regular octahedron in Example 2.3.

3. On k -polytopes with $k + 4$ vertices. Gale transforms and Gale diagrams have been very helpful in the study of k -polytopes with up to $k + 3$ vertices (see [19]). For k -polytopes with $k + 4$ vertices, however, a Gale diagram is a three-dimensional configuration that loses its essential properties (whether or not $0 \in \text{rel int conv } \bar{X}(I)$ for some $I \subset \{1, \dots, n\}$) under projections into the plane. In this case affine Gale diagrams are genuinely two-dimensional objects, which explains their usefulness to the study of k -polytopes with $k + 4$ vertices.

To begin with, let us consider a three-dimensional example. We represent an affine Gale diagram (Π, A) graphically by labeling points $i \in A$ with a “−” and the points $i \notin A$ with a “+”.

Example 3.1. Add a point x_7 right above the center of the facet $\text{conv } \{x_1, x_4, x_5\}$ of the regular octahedron $\mathcal{O}$ from Example 2.3. The convex hull of $\mathcal{O} \cup \{x_7\}$ is a 3-polytope $\mathcal{O}'$ with 7 vertices whose face lattice is derived from the face lattice of $\mathcal{O}$ by *stellar subdivision* of the facet 145. The affine chirotope in Fig. 2(c) with $A = \{3, 4\}$ forms an affine Gale diagram of $\mathcal{O}'$.

In the plane there are only four possible types of Radon partitions which in a rank 3 affine Gale diagram correspond to three different types of facets, indicated in Fig. 3. The types (a) and (b) describe the cofacet of a simplicial facet while (c) and (d) correspond to nonsimplicial facets with $k + 1$ and $k + 2$ vertices, respectively. In contrast to the simplicial Example 3.1 where we had only cofacets of types (a) and (b), the following discussion will draw our attention to facets of type (d).

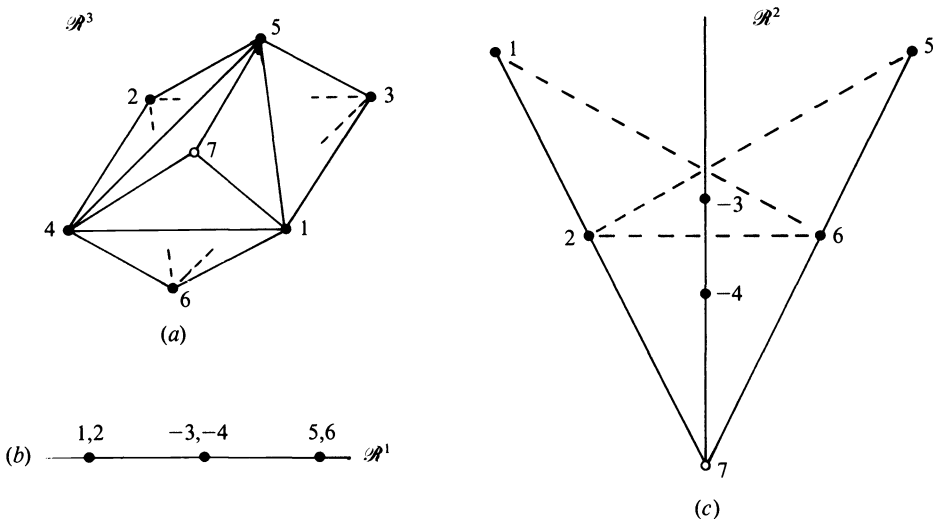


FIG. 2. (a) A polytope $\mathcal{O}'$ obtained from a regular octahedron $\mathcal{O}$ by stellar subdivision of the facet 145. (b) Affine Gale diagram of $\mathcal{O}$. (c) Affine Gale diagram of $\mathcal{O}'$.

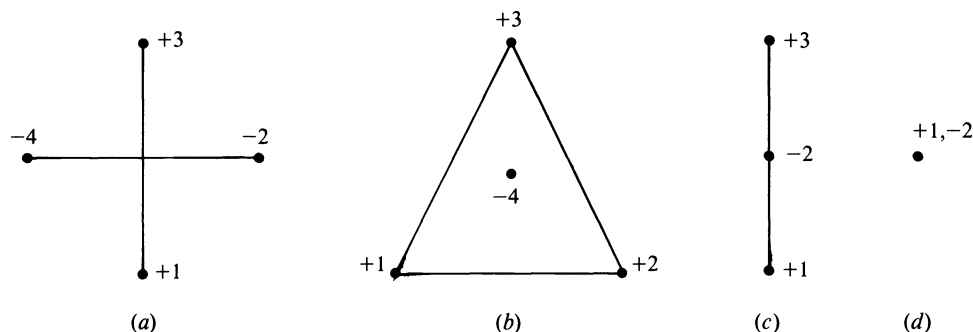


FIG. 3. The four possible types of cofacets in a plane affine Gale diagram.

In 1980 Lawrence found a construction that assigns to every chirotope χ a higher dimensional chirotope $\Lambda(\chi)$ such that the face lattice of $\Lambda(\chi)$ is polytopal if and only if the original chirotope χ is realizable. His result has been first published in [3, Thm. 2.2]; see also [9, Thm. 6.1] for an alternative description and proof.

Lawrence's construction is very easily described and understood using affine Gale diagrams. Given an affine Gale diagram (Π, A) for a d -chirotope χ on $E = \{1, 2, \dots, n\}$, we "double" every point in Π to obtain a chirotope Π' on $E \cup E'$ where $E' := \{n+1, \dots, 2n\}$. Setting $A' := A \cup \{a' \in E' \mid a' - n \notin A\}$, i.e., negating every new point, we obtain an affine Gale diagram (Π', A') for $\Lambda(\chi)$.

Proposition 2.5 shows that (Π', A') is the affine Gale diagram of a $(2n - d - 1)$ -polytope with $2n$ vertices. It follows from the fact that every set of the form $\{i, i + n\}$, $i \in E$, is a cofacet and that the chirotopes corresponding to *Lawrence polytopes* are uniquely determined by their face lattice.

Figure 4 shows the affine Gale diagram of a Lawrence polytope, the prism over a tetrahedron. It is not difficult to see that the underlying idea generalizes and that every prism over a simplex is a Lawrence polytope.

The Λ -technique and also the σ -construction described in [25] provide the possibility to construct infinitely many nonisomorphic nonpolytopal spheres. Note that all these spheres have two-element cofacets which means that they are highly nonsimplicial. These degeneracies are in a certain sense unsatisfactory, and Grünbaum [private communication] raised the question whether our approach can be modified to obtain "less flat" nonpolytopal spheres.

Before turning to this question in the next section let us formulate as a general result on affine Gale diagrams the equivalence of the Steinitz problem for k -polytopes with $k + 4$ vertices with the realizability problem for affine 3-chirotopes. Part (i), the reduction

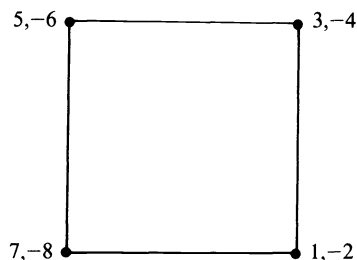


FIG. 4. Affine Gale diagram of a prism over a tetrahedron.

of the Steinitz problem for k -polytopes with $k + 4$ vertices into the plane, is due to Bokowski. For a proof see [7].

THEOREM 3.2.

(i) Given a combinatorial $(k - 1)$ -sphere S with $k + 4$ vertices, there exists a finite set $\Xi(S)$ of affine 3-chirotopes with $k + 4$ vertices such that S is polytopal if and only if at least one $\chi \in \Xi(S)$ is realizable.

(ii) Conversely, given any affine 3-chirotope χ with at least 7 vertices, there exists a combinatorial sphere S with $\chi \in \Xi(S)$.

4. A negative Steinitz-type theorem for simplicial k -polytopes with $k + 4$ vertices. A recent negative Steinitz-type theorem of the author [25] says that there exists no local characterization for face lattices of k -polytopes for $k \geq 6$. In this section we prove an analogous result for simplicial polytopes with few vertices by describing a construction method for an infinite family of minorminimal nonpolytopal simplicial $(k - 1)$ -spheres with $k + 4$ vertices, where $k \in \mathbb{N}$ is not fixed.

Let χ be a nonrealizable affine simplicial 3-chirotope with n vertices. We can represent χ by an arrangement A of points $x_1, \dots, x_n$ and $\binom{n}{2}$ pseudolines in the plane. The arrangement A divides the plane into disjoint regions, k of which are contained in the (generalized) convex hull of the n vertices of χ . Insert new points $x_{n+1}, \dots, x_{n+k}$ such that each interior region contains one of the new points and that the 3-chirotope χ' with $n + k$ vertices is simplicial, too. Then $(\chi', \{n + 1, \dots, n + k\})$ is the affine Gale diagram of an affine $(n + k - 3)$ -chirotope $\tau(\chi)$ with $n + k$ extreme vertices.

Example 4.1. If χ is the affine chirotope of a pentagon, then $\tau(\chi)$ is the affine chirotope of a 12-polytope with 16 vertices (see Fig. 5).

LEMMA 4.2. If χ is nonrealizable then $\tau(\chi)$ is nonpolytopal.

Proof. Let $m := (n + 1, \dots, n + k)$. Since $\chi^* = \tau(\chi)/m$ we have to show that the orientations of all simplices $\lambda \in \Lambda(n + k, n + k - 3)$ with $m \subset \lambda$ are determined by the face lattice of $\tau(\chi)$.

Clearly, the face lattice of a simplicial chirotope determines the orientation of all outer simplices. Here $\lambda \in \Lambda(n + k, n + k - 3)$ is called an *outer simplex* if it contains a facet of $\tau(\chi)$.

Thus Lemma 4.2 follows if we can prove that all λ with $m \subset \lambda$ are outer simplices of $\tau(\chi)$. We will do so by showing that for every such λ there exists an $m_i \in m$ such that $\lambda \setminus m_i$ is a facet of $\tau(\chi)$.

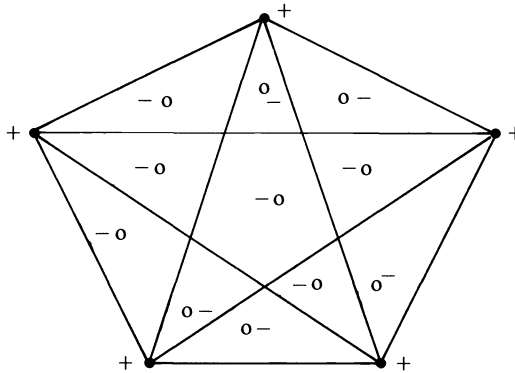


FIG. 5. Affine Gale diagram of a simplicial 12-polytope with 16 vertices obtained from a pentagon by τ -construction.

Let $\{a, b, c\} = \{1, \dots, n+k\} \setminus \lambda$. It follows from the above definition of the τ -construction that in the extended plane pseudoline-arrangement χ' some (new) point m_i lies in the convex hull of $\{a, b, c\}$. Hence $\{a, b, c, m_i\}$ is a cofacet of $\tau(\chi)$ of type Fig. 3(a). Consequently $\lambda \setminus m_i$ is a facet of $\tau(\chi)$, and λ is an outer simplex which proves the lemma. $\square$

In general it will not be necessary to insert all k new points in the original plane diagram. In other words, one can look for a suitable minor $\tilde{\tau}(\chi)$ of $\tau(\chi)$ which still fulfills the assertion of the above theorem but no proper minor of $\tilde{\tau}(\chi)$ does. With this argument we can prove the following theorem.

THEOREM 4.3. *There exists a sequence $\{k_n\}_{n \in \mathbb{N}}$ of integers and a family $\{S_n\}_{n \in \mathbb{N}}$ of triangulated $(k_n - 1)$ -spheres with $k_n + 4$ vertices, such that for all $n \in \mathbb{N}$*

- (i) S_n is not polytopal.
- (ii) Every proper subcomplex of S_n extends to a polytopal sphere with no additional vertices.

Proof. Apply the above τ -construction to the family $\{\chi_n\}$ of minorminimal non-realizable simplicial 3-chirotopes in [7]. The face lattice S_n of $\tilde{\tau}(\chi_n)$ is a triangulated PL-sphere by a result of Mandel (see [2, Thm. 20]), and hence S_n has the desired properties by Lemma 4.2. $\square$

In order to give a satisfactory answer to Grünbaum's question, which we mentioned in the previous section, it would be desirable to show the existence of a family of minorminimal nonpolytopal *simplicial* spheres of some fixed dimension. In the view of Kalai's results on the asymptotic number of (nonpolytopal) triangulated spheres [13], it is likely that such a family exists. However, this interesting problem is still open.

CONJECTURE 4.4. *There exists a (fixed) $d \in \mathbb{N}$ and an infinite family $\{T_n\}_{n \in \mathbb{N}}$ of triangulated $(d - 1)$ -spheres such that for all $n \in \mathbb{N}$*

- (i) T_n is not polytopal.
- (ii) Every proper subcomplex of T_n extends to a polytopal sphere with no additional vertices.

5. Facets of 4-polytopes cannot be prescribed arbitrarily. As a second application of the affine Gale diagram technique, let us discuss facets of nonarbitrary shape for 4-polytopes. We say that a facet F of a polytope P is *arbitrarily prescribable* if for all polytopes F' combinatorially equivalent to F , there is a polytope P' combinatorially equivalent to P such that F' is the image of F under the isomorphism between P and P' [11], [15]. It is well known as a consequence of the proof of Steinitz theorem that facets of 3-polytopes are arbitrarily prescribable. The same result holds for k -polytopes with up to $k + 3$ vertices.

In [15] Kleinschmidt shows that a certain 4-polytope Q with 8 vertices has a facet of nonarbitrary shape. Figure 6(a) shows an affine Gale diagram of the polytope Q . The labeling of the vertices agrees with the labeling in [11, Table 1] and in [15].

The positions of the points 2 and 5 forces the lines $\overline{17}$ and $\overline{18}$ to be distinct because 5678 and 2378 are cofacets of Q . This means that in the affine Gale diagram of the facet 235678 the points 7 and 8 have to be distinct, i.e., in no polytope combinatorially equivalent to Q the octahedron 235678 can be regular (see Fig. 2(b)).

The affine Gale diagram also provides an immediate solution to the problems [11, p. 139, (b) and (c)]. Since the underlying chirotope of our affine Gale diagram is not projectively unique, also the polytope Q is not projectively unique. It is furthermore clear that the affine Gale diagram can be coordinatized with rational coordinates which implies the same property for Q .

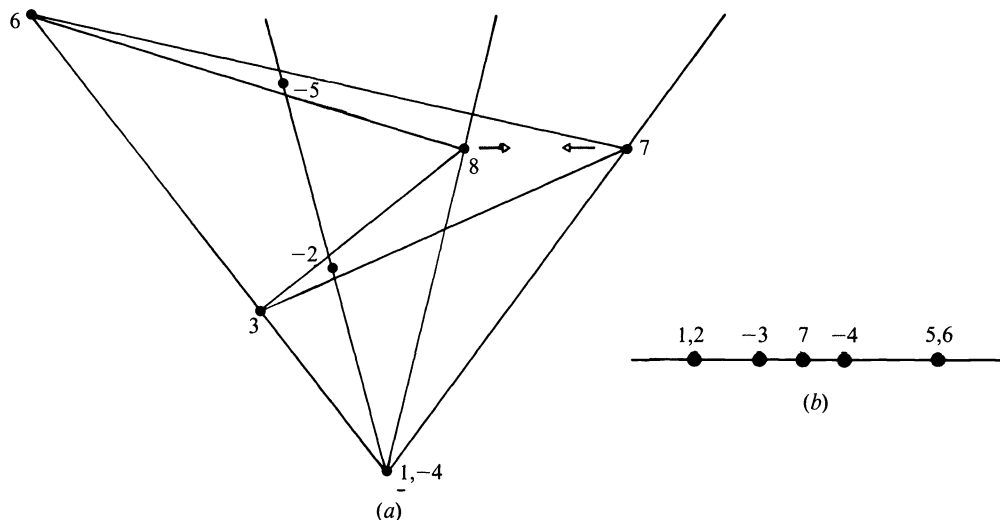


FIG. 6. (a) Affine Gale diagram of Kleinschmidt's polytope Q —the facet 235678, an octahedron, cannot be prescribed. (b) Affine Gale diagram of a polytope P^* . The vertex figure at vertex 7, an octahedron, cannot be prescribed.

The polytope Q has 14 facets, and we can ask whether this number of facets is minimal for polytopes with nonarbitrarily shaped facets. By checking all possible combinatorial types, it is easy to see that no 4-polytope with less than 7 facets has this property.

PROPOSITION 5.1. *There is a 4-polytope P with 7 facets, one of which is not arbitrarily prescribable.*

Proof. Let P be the polar polytope of the 4-polytope P^* with 7 vertices whose affine Gale diagram is given in Fig. 6(b). The vertex figure of point 7 in P^* is an octahedron. The affine Gale diagram of the regular octahedron in Fig. 2(b), however, cannot be extended by a point 7 such that $\{3, 4, 7\}$ becomes a cofacet. So, the vertex figure of P^* at point 7 is not arbitrarily prescribable, and therefore the facet 7 of the polar polytope P , a cube, is not arbitrarily prescribable. $\square$

6. Affine Gale diagrams for simplicial complexes. Given a simplicial complex $\mathcal{S}$ of dimension k it is well known that $\mathcal{S}$ can be imbedded without self-intersections in $(2k + 1)$ -dimensional Euclidean space. Yet, determining the smallest number $l(\mathcal{S}) \leq 2k + 1$ for which such an imbedding is possible is in general a hard geometrical problem. Naturally, it is of particular interest whether or not $l(\mathcal{S}) = k + 1$.

In this section we derive necessary and sufficient conditions for an affine Gale diagram to represent an imbedding of a given simplicial complex $\mathcal{S}$. This result allows us to give a simple explanation for the imbeddability of the *Möbius torus*, the unique triangulation of the torus with only 7 vertices. For a complete classification of all imbeddings of the Möbius torus as a *chirotope manifold* and a discussion of possible affine symmetries see Bokowski and Eggert [5].

An affine d -chirotope χ with n vertices is said to *realize* a simplicial complex $\mathcal{S} \subset \mathcal{P}(\{1, \dots, n\})$ if for any two faces $A, B \in \mathcal{S}$ the pair (A, B) is not a Radon partition in χ . Two subsets $A, B \in \{1, \dots, n\}$ are *complementary halfspaces* of a chirotope χ if there exists a cocircuit C of χ such that $A = C^+$ and $B = C^-$.

PROPOSITION 6.1. *An $(n - d)$ -chirotope Π with n vertices is dual to a realization of a simplicial complex $\mathcal{S} \subset \mathcal{P}(\{1, \dots, n\})$ if and only if no two faces $A, B \in \mathcal{S}$ are complementary halfspaces of Π .*

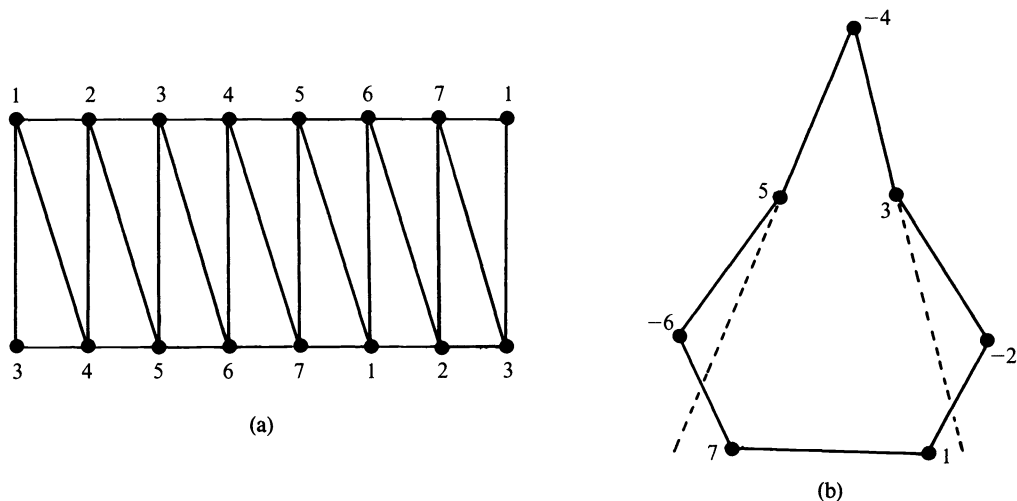


Fig. 7. (a) The Möbius torus $\mathcal{M}$. (b) Affine Gale diagram of an imbedding of $\mathcal{M}$ in 3-space.

Proof. If (A, B) is a Radon partition of an affine chirotope χ , then there is a minimal Radon partition (A', B') such that $A' \subset A$ and $B' \subset B$. The minimal Radon partitions of an affine chirotope are exactly the circuits, i.e., there is a circuit C of χ such that $C^+ = A'$ and $C^- = B'$. By definition, C is a cocircuit of the dual chirotope χ^* , i.e., A' and B' are complementary halfspaces of χ^* . But, if A and B are faces of the simplicial complex $\mathcal{S}$, so are A' and B' . Hence there exists a Radon partition (A, B) of faces of $\mathcal{S}$ in χ if and only if there exist faces A' and B' of $\mathcal{S}$ which are complementary halfspaces of the dual χ^* . $\square$

Example 6.2. The Möbius torus. We claim that the configuration in Fig. 7(b) is the affine Gale diagram $(\chi, \{2, 4, 6\})$ of an imbedding of the Möbius torus $\mathcal{M}$ (see Fig. 7(a)). By Proposition 6.1 we have to show that there is no pair (A, B) of complementary halfspaces in χ such that both A and B are facets. Since we have $|A| + |B| = 5$ for all pairs (A, B) of complementary halfspaces, it is sufficient to check that no 2-face of $\mathcal{M}$ is a halfspace in χ . This can be done by complete enumeration of all halfspaces of χ . Two examples: $\{4, 6\}$ and $\{3, 5, 7\}$ are halfspaces of χ , separated by the hyperplane $\{1, 2\}$; $\{6\}$ and $\{1, 2, 5, 7\}$ are halfspaces of χ , separated by the hyperplane $\{3, 4\}$. We see that none of the facets of $\mathcal{M}$ as indicated in Fig. 7(a), e.g., 124, 134, forms a halfspace of χ . This proves that any realization of the dual χ^* yields an imbedding of the Möbius torus in three-dimensional Euclidian space.

7. On the universality of neighborly polytopes. A convex polytope $P \subset \mathbb{R}^d$ is called *m-neighborly* if the convex hull of every m -element set of vertices of P is a face of P . As usual in the theory of neighborly polytopes, see [23], [26], we restrict ourselves to the even-dimensional case. So, by a *neighborly* polytope we mean an *m-neighborly* $2m$ -polytope $P \subset \mathbb{R}^{2m}$ for some m . Given any convex polytope P and a face F of P , the quotient P/F is a polytope whose face lattice is isomorphic to the upper segment $[F, P]$ in the face lattice of P . This definition generalizes in the obvious way to triangulated spheres. Clearly, the face lattice of a contraction χ/F of a chirotope χ by a face F of χ is the quotient $\text{FL}(\chi)/F$.

In this final section we prove a partial result in connection with an open problem due to Perles. It is known [1] (and easy to prove with chirotopes) that for $\beta \leq 3$ every simplicial d -polytope with $d + \beta$ vertices is the quotient of a cyclic (hence neighborly)

polytope while this property fails to hold for all $\beta \geq 4$. The natural generalization of this property to simplicial polytopes with more vertices reads as follows.

CONJECTURE 7.1 (Perles). *Every simplicial polytope is a quotient of an (even-dimensional) neighborly polytope.*

As G. Kalai [private communication] remarked, Conjecture 7.1 is in a certain sense dual to the longstanding open conjecture that every simplicial polytope is combinatorially equivalent to a subpolytope of a stacked polytope (see Shephard [24]). For example, subpolytopes of neighborly polytopes are neighborly and quotients of stacked polytopes are stacked. Here we prove the following partial result.

THEOREM 7.2. *Every simplicial d -polytope with n vertices is the quotient of a $(d + 1)$ -neighborly $(n + d)$ -polytope with $2n$ vertices.*

Theorem 7.2 solves the problem in a certain asymptotical sense for simplicial d -polytopes with few vertices: setting $n = d + \beta$ for any fixed β we have $(d + 1)/(n + d) \rightarrow 1/2$ as $d \rightarrow \infty$. For example, every simplicial 6-polytope with 10 vertices is the quotient of a 7-neighborly 16-polytope with 20 vertices. Affine Gale diagrams seem to be a very suitable method to tackle Conjecture 7.1 in the case of d -polytopes with $d + 4$ vertices.

Recall the following definition from the theory of oriented matroids [4], [22]. Two distinct elements e and e' of E are called *covariant* (respectively, *contravariant*) in a chirotope χ on $E = \{1, 2, \dots, n\}$ if e and e' have always the same sign (respectively, opposite sign) in all circuits of χ containing them. Observe that a pair $\{e, e'\}$ is covariant in χ if and only if it is contravariant in χ^* .

Proof of Theorem 7.2. Let P be a simplicial d -polytope with n vertices and χ the affine $(d + 1)$ -chirotope associated with P . We need to show that there exists a simplicial $(d + 1)$ -neighborly $(n + d + 1)$ -chirotope $\tilde{\chi}$ on $E \cup E'$, E' a disjoint copy of $E = \{1, 2, \dots, n\}$, such that $\tilde{\chi}/E' = \chi$.

Let $\Pi = \chi^*$ be the dual of M , which is a realizable $(n - d - 1)$ -chirotope on E . By adding n points “nearly antipodal” to E , we can construct a realizable extension $\tilde{\Pi}$ of Π on $E \cup E'$ such that $\tilde{\Pi}$ is simplicial and that $\{e, e'\}$ is *covariant* for all $e \in E$. For example, such an $\tilde{\Pi}$ can be obtained by inductive *principle extensions* [8, § 5] of the form $e' = [e^-, e_1^-, \dots, e_{i_n-d-2}^-]$.

The dual $\tilde{\chi} := \tilde{\Pi}^*$ of $\tilde{\Pi}$ is a realizable simplicial $(n + d + 1)$ -chirotope on $E \cup E'$ with $\tilde{\chi}/E' = \chi$ such that all pairs $\{e, e'\}$, $e \in E$, are contravariant in $\tilde{\chi}$. Let C be a signed circuit of $\tilde{\chi}$. Since χ is simplicial, we have $|C| = n + d + 2$ and $|(E \cup E') \setminus C| = n - d - 2$. Hence the set $A := \{e \in E \mid \{e, e'\} \subset C\}$ has at least $d + 2$ elements. Since the pairs $\{e, e'\}$ with $e \in E$ are contravariant in $\tilde{\chi}$, we have $e \in C^+ \Leftrightarrow e' \in C^-$ and $e \in C^- \Leftrightarrow e' \in C^+$ for all $e \in A$. This implies $|C^+| \geq d + 2$ and $|C^-| \geq d + 2$. Hence $\tilde{\chi}$ is $(d + 1)$ -neighborly by [26, Lemma 2.2]. $\square$

It is natural to ask how far this result generalizes to arbitrary (not necessarily simplicial) convex polytopes. If the assumption “uniform” is dropped in the hypotheses of Theorem 7.2, the counting argument in the end of the above proof has to be modified. Still, we can derive the following weaker statement.

COROLLARY 7.3 *Every d -polytope with n vertices is the quotient of a 3-neighborly $(n + d)$ -polytope with $2n$ vertices.*

As P. Gritzmann pointed out to the author, Corollary 7.3 can also be obtained directly by n -fold application of the dual wedge construction which corresponds to doubling all points in an affine Gale diagram. Let us mention the following related result which has been obtained very recently by J.-P. Roudneff [private communication].

THEOREM 7.4 (Roudneff). *Every simplicial d -polytope with n vertices is the quotient of a $(t + 1)$ -neighborly $(d + 2t)$ -polytope with $n + 2t$ vertices for all $t \geq 0$.*

Note added in proof. An alternative proof for Proposition 5.1 can be found in [27].

REFERENCES

- [1] A. ALTSHULER AND M. PERLES, *Quotient polytopes of cyclic polytopes. Part I: Structure and characterization*, Israel J. Math., 36 (1980), pp. 97–125.
- [2] A. BACHEM, *Convexity and optimization in discrete structures*, in Convexity and Applications, P. M. Gruber and J. Wills, eds., Birkhäuser, Basel, 1983.
- [3] L. J. BILLERA AND B. S. MUNSON, *Polarity and inner products in oriented matroids*, European J. Combin., 5 (1984), pp. 293–308.
- [4] R. G. BLAND AND M. LAS VERGNAS, *Orientability of matroids*, J. Combin. Theory Ser. B, 24 (1978), pp. 94–123.
- [5] J. BOKOWSKI AND A. EGGERT, *All realizations of Möbius' torus with seven vertices*, manuscript, Technische Hochschule Darmstadt, 1986.
- [6] J. BOKOWSKI AND I. SHERER, *Neighborly 6-polytopes with 10 vertices*, Israel J. Math., 58 (1987), pp. 103–124.
- [7] J. BOKOWSKI AND B. STURMFELS, *An infinite family of minorminimal nonrealizable simplicial 3-chirotopes*, submitted.
- [8] ———, *On the coordinatization of oriented matroids*, Discrete Comput. Geom., 1 (1986), pp. 293–306.
- [9] ———, *Polytopal and non-polytopal spheres—An algorithmic approach*, Israel J. Math., 57 (1987), pp. 257–271.
- [10] G. EWALD, *Spherical complexes and nonprojective toric varieties*, Discrete Comput. Geom., 1 (1986), pp. 115–122.
- [11] G. EWALD, P. KLEINSCHMIDT, U. PACHNER, AND C. SCHULZ, *Neuere Entwicklungen in der kombinatorischen Konvexgeometrie*, in Contributions to Geometry, J. Tölke and J. M. Wills, eds., Birkhäuser, Basel, 1978.
- [12] B. GRÜNBAUM, *Convex Polytopes*, Interscience, London, 1967.
- [13] G. KALAI, *Many triangulated spheres*, manuscript, The Hebrew University, Jerusalem, 1986.
- [14] P. KLEINSCHMIDT, *Sphären mit wenigen ecken*, Geom. Dedicata, 5 (1976), pp. 307–320.
- [15] ———, *On facets with non-arbitrary shapes*, Pacific J. Math., 65 (1976), pp. 511–515.
- [16] M. LAS VERGNAS, *Matroides orientables*, C.R. Acad. Sci. Paris Sér. A, 280 (1975), pp. 61–64.
- [17] ———, *Convexity in oriented matroids*, J. Combin. Theory Ser. B, 29 (1980), pp. 231–243.
- [18] P. MANI, *Spheres with few vertices*, J. Combin. Theory Ser. A, 13 (1972), pp. 346–352.
- [19] P. McMULLEN, *Transforms, diagrams and representations*, in Contributions to Geometry, J. Tölke and J. M. Wills, eds., Birkhäuser, Basel, 1978.
- [20] P. McMULLEN AND G. C. SHEPHARD, *Convex polytopes and the upper bound conjecture*, London Math. Soc. Lecture Notes, Ser. 3, 1971.
- [21] M. RIEGER, *Shemer-Sphären mit 10 Ecken im $\mathbb{R}^6$ und Rückführung auf ein ebenes Problem*, Staatsexamensarbeit, Bochum, 1983.
- [22] J. P. ROUDNEFF, *Inseparability graphs of oriented matroids*, Manuscript, 1986.
- [23] I. SHERER, *Neighborly polytopes*, Israel J. Math., 43 (1982), pp. 291–314.
- [24] G. C. SHEPHARD, *Subpolytopes of stack polytopes*, Israel J. Math., 19 (1974), pp. 292–296.
- [25] B. STURMFELS, *Boundary complexes of convex polytopes cannot be characterized locally*, J. London Math. Soc., 35 (1987), pp. 314–326.
- [26] ———, *Neighborly polytopes and oriented matroids*, European J. Combin., to appear.
- [27] D. W. BARNETTE, *Two "simple" 3-spheres*, Discrete Math., 67 (1987), pp. 97–99.

THE VORONOI REGION OF E_7^*

R. T. WORLEY†

Abstract. The Voronoi region and covering radius of the lattice E_7^* are determined, and the normalised second moment is calculated, confirming the estimate given by Conway and Sloane [*Voronoi regions of lattices, second moments of polytopes, and quantization*, IEEE Trans. Inform. Theory, 28 (1982), pp. 211–226].

Key words. quantization, Voronoi region, covering radius

AMS(MOS) subject classifications. primary 10E30; secondary 52A25, 52A45

Introduction. For an n -dimensional lattice $\Lambda \subset \mathbb{R}^n$ the Voronoi region is the convex polytope

$$V_\Lambda = \{\mathbf{x} \in \mathbb{R}^n: |\mathbf{x}| \leq |\mathbf{x} - \mathbf{l}| \text{ for all } \mathbf{l} \in \Lambda\}$$

where $|\mathbf{x}| = \sqrt{\mathbf{x} \cdot \mathbf{x}}$ denotes the length of $\mathbf{x}$. Conway and Sloane [1], [3] have investigated the Voronoi regions of certain special lattices and calculated their second moments in connection with the use of the lattices as quantizers. In particular they calculated the second moments of E_6^* and E_7^* by Monte Carlo integration as their Voronoi regions were not known.

This paper determines the Voronoi region of E_7^* , confirms the value for the second moment obtained by Conway and Sloane, and shows that E_7^* provides a slightly worse covering than A_7^* . The method used is similar to that used in [5] in which the Voronoi region of E_6^* is determined. A polytope W containing the Voronoi region V is defined, the vertices of W are calculated and shown to lie in V , so $W = V$.

1. The lattice E_7^* and its isometries. We write the points of $\mathbb{R}^7$ as 7-tuples or row vectors $(x_1, \dots, x_7)$. A generating matrix G for E_7 is given in Sloane [4]. The rows of

$$(G^{-1})^T = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 1 & 0 & 0 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 2 & 0 & -2 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

form a basis for E_7^* . A better picture of E_7^* may be obtained by observing [3] that it contains a sublattice $2\mathbb{Z}^7$ of index 16 with coset representatives

$$\begin{aligned} \mathbf{v}_0 &= (0, 0, 0, 0, 0, 0, 0), & \bar{\mathbf{v}}_0 &= (1, 1, 1, 1, 1, 1, 1), \\ \mathbf{v}_1 &= (1, 1, 1, 0, 1, 0, 0), & \bar{\mathbf{v}}_1 &= (0, 0, 0, 1, 0, 1, 1), \\ \mathbf{v}_2 &= (0, 1, 1, 1, 0, 1, 0), & \bar{\mathbf{v}}_2 &= (1, 0, 0, 0, 1, 0, 1), \\ \mathbf{v}_3 &= (0, 0, 1, 1, 1, 0, 1), & \bar{\mathbf{v}}_3 &= (1, 1, 0, 0, 0, 1, 0), \\ \mathbf{v}_4 &= (1, 0, 0, 1, 1, 1, 0), & \bar{\mathbf{v}}_4 &= (0, 1, 1, 0, 0, 0, 1), \\ \mathbf{v}_5 &= (0, 1, 0, 0, 1, 1, 1), & \bar{\mathbf{v}}_5 &= (1, 0, 1, 1, 0, 0, 0), \end{aligned}$$

* Received by the editors February 9, 1987; accepted for publication (in revised form) October 14, 1987.

† Department of Mathematics, Monash University, Clayton, Victoria, Australia 3168.

$$\begin{aligned} \mathbf{v}_6 &= (1, 0, 1, 0, 0, 1, 1), & \bar{\mathbf{v}}_6 &= (0, 1, 0, 1, 1, 0, 0), \\ \mathbf{v}_7 &= (1, 1, 0, 1, 0, 0, 1), & \bar{\mathbf{v}}_7 &= (0, 0, 1, 0, 1, 1, 0). \end{aligned}$$

Other representations of E_7^* may be found in [2]. It will be noted that $\bar{\mathbf{v}}_i$ is the binary complement of $\mathbf{v}_i$, and that $\mathbf{v}_1, \dots, \mathbf{v}_7$ are cyclic permutations of each other. Because these $\mathbf{v}_i, \bar{\mathbf{v}}_i$ are a complete set of representatives of $E_7^* \bmod 2\mathbb{Z}^7$, the sum and difference of any representatives, modulo 2, are again representatives. Any point $\mathbf{l}$ of E_7^* is, modulo 2, one of these representatives. Thus all points of E_7^* of length less than $\sqrt{2}$ can be obtained from these by replacing any 1 by a -1 , or by replacing just one 0 in $\mathbf{v}_0$ by ± 2 .

An isometry of E_7^* is a length-preserving linear transformation of $\mathbb{R}^7$ which preserves E_7^* . There are some obvious isometries, namely:

(i) The cycling permutation

$$C: (l_1, l_2, \dots, l_7) \rightarrow (l_2, l_3, \dots, l_7, l_1)$$

and its powers, which cycle $\mathbf{v}_1, \dots, \mathbf{v}_7 \rightarrow \mathbf{v}_7, \mathbf{v}_1, \dots, \mathbf{v}_6$.

(ii) Reflection in any coordinate

$$M_i: l_i \rightarrow -l_i \quad \text{with the other } l_j \text{ unchanged.}$$

(iii) The reordering

$$R: (l_1, \dots, l_7) \rightarrow (l_1, l_3, l_5, l_7, l_2, l_4, l_6),$$

which maps $\mathbf{v}_1, \dots, \mathbf{v}_7$ into $\mathbf{v}_1, \mathbf{v}_5, \mathbf{v}_2, \mathbf{v}_6, \mathbf{v}_3, \mathbf{v}_7, \mathbf{v}_4$.

(iv) The isometry

$$S: c_1 \mathbf{v}_1 + \dots + c_7 \mathbf{v}_7 \rightarrow c_1 \mathbf{v}_4 + c_2 \mathbf{v}_1 + c_3 \mathbf{v}_7 + c_4 \mathbf{v}_2 + c_5 \mathbf{v}_6 + c_6 \mathbf{v}_5 + c_7 \mathbf{v}_3$$

is a rotation through $2\pi/3$ in the plane of $\mathbf{v}_1$ and $\mathbf{v}_2$, combined with some reflections.

(v) The “quaternion” isometry

$$Q_1: (l_1, \dots, l_7) \rightarrow (l'_1, l'_2, l'_3, l_4, l'_5, l_6, l_7)$$

where

$$l'_1 = (l_1 + l_2 + l_3 + l_5)/2,$$

$$l'_2 = (l_1 + l_2 - l_3 - l_5)/2,$$

$$l'_3 = (l_1 - l_2 + l_3 - l_5)/2,$$

$$l'_5 = (l_1 - l_2 - l_3 + l_5)/2,$$

which can be found on noting that $(2, 0, 0, 0, 0, 0, 0)$ and $\mathbf{v}_1$ have the same length. Other Q_i corresponding to $\mathbf{v}_2, \dots, \mathbf{v}_7$ exist.

For later reference we consider the effect of these isometries on $\mathbf{v}_1, \dots, \mathbf{v}_7$:

(a) Any $\mathbf{v}_i$ may be transformed to $\mathbf{v}_1$ by a power of C .

(b) Any two distinct $\mathbf{v}_i, \mathbf{v}_j$ may be transformed to $\mathbf{v}_1, \mathbf{v}_2$ since applying a power of C makes $i = 1, j = 2, 3$, or 4 , and then applying R if $j = 3$ or S^{-1} if $j = 4$ gives $\mathbf{v}_1, \mathbf{v}_2$.

(c) Any three distinct $\mathbf{v}_i, \mathbf{v}_j, \mathbf{v}_k$ may be transformed to $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_4$. By (b) we may assume $i = 1, j = 2$, and we may assume $k = 3, 4, 5$, or 6 . Then

$$\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\} \xrightarrow{CS^{-1}C} \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_5\} \xrightarrow{C^3S^{-1}} \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_6\} \xrightarrow{C^3M_2M_3M_5Q_1C^6} \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_4\}.$$

2. Faces and vertices of the Voronoi region of E_7^* . The points $\mathbf{x}$ of the Voronoi region of a lattice Λ satisfy $\mathbf{x} \cdot \mathbf{l} \leq \frac{1}{2} \mathbf{l} \cdot \mathbf{l}$ for all points $\mathbf{l} \neq \mathbf{0}$ of the lattice Λ . Since $-\mathbf{l}$ is in

a lattice whenever $\mathbf{l}$ is, we may assume $\mathbf{l}$ has the first nonzero component positive ($\mathbf{l} > \mathbf{0}$) and determine the Voronoi region by

$$|\mathbf{x} \cdot \mathbf{l}| \leq \frac{1}{2} \mathbf{l} \cdot \mathbf{l} \quad \text{for all } \mathbf{l} \in \Lambda.$$

In reality only a finite number of lattice points $\mathbf{l} > \mathbf{0}$ close to the origin is needed.

For E_7^* , let W denote the polytope defined by the following faces:

$$(\pm 2, 0, 0, 0, 0, 0, 0) \cdot \mathbf{x} = 2 \quad (\text{type 1}),$$

$$(\pm 1, \pm 1, \pm 1, 0, \pm 1, 0, 0) \cdot \mathbf{x} = 2 \quad (\text{type 2}),$$

$$(0, 0, 0, \pm 1, 0, \pm 1, \pm 1) \cdot \mathbf{x} = \frac{3}{2} \quad (\text{type 3})$$

and their images under powers of C , arising from all points $\mathbf{l}$ of E_7^* with length less than $\sqrt{\frac{7}{2}}$. To find all the vertices of W we only need to look in the nonnegative orthant and apply the isometries M_i to the vertices we find there.

Let $\mathbf{x}$ be a vertex of W in the nonnegative orthant. If $\mathbf{x}$ lies on a type 2 face $\mathbf{l} \cdot \mathbf{x} = 2$, then $\mathbf{x}$ lies on the corresponding face $\mathbf{v}_j \cdot \mathbf{x} = 2$ where $\mathbf{v}_j = (|l_1|, \dots, |l_7|)$, and must have $x_i = 0$ whenever $l_i < 0$. If either two l_i are negative, or there are two distinct $\mathbf{l}$ corresponding to the same $\mathbf{v}_j$, then $\mathbf{x}$ can have at most two nonzero components. For $\mathbf{x}$ to be on the correct side of the type 1 faces, $\mathbf{x}$ must therefore have two components equal to 1. However this makes $\mathbf{x}$ on the wrong side of a type 3 face. One can argue similarly for the type 3 faces. Thus we have the following proposition:

PROPOSITION 1. *A vertex of W in the nonnegative orthant lies on at most one type 1 face. If it lies on a type 2 or type 3 face $\mathbf{l} \cdot \mathbf{x} = k$, then it lies on the corresponding face $\mathbf{v}_i \cdot \mathbf{x} = k$ (or $\bar{\mathbf{v}}_i \cdot \mathbf{x} = k$), and $\mathbf{l}$ differs from $\mathbf{v}_i$ (or $\bar{\mathbf{v}}_i$) by replacing a single "1" by "-1". Furthermore there is at most one $\mathbf{l} \neq \mathbf{v}_i$ (or $\bar{\mathbf{v}}_i$) corresponding to $\mathbf{v}_i$ (or $\bar{\mathbf{v}}_i$).*

Now suppose the nonnegative vertex $\mathbf{x}$ of W lies on both $\mathbf{v}_i \cdot \mathbf{x} = 2$ and $\bar{\mathbf{v}}_i \cdot \mathbf{x} = 1\frac{1}{2}$. Without loss of generality we may take $i = 1$, so

$$x_1 + x_2 + x_3 + x_5 = 2, \quad x_4 + x_6 + x_7 = 1\frac{1}{2}.$$

Either $x_1 + x_3 \geq 1$ or $x_2 + x_5 \geq 1$, so for $\mathbf{x}$ to satisfy $|\mathbf{x} \cdot \bar{\mathbf{v}}_5| \leq 1\frac{1}{2}$ and $|\mathbf{x} \cdot \bar{\mathbf{v}}_6| \leq 1\frac{1}{2}$ we must have $x_4 \leq \frac{1}{2}$. Similarly considering $x_1 + x_2 \geq 1$ or $x_3 + x_5 \geq 1$ we deduce $x_6 \leq \frac{1}{2}$, and we also get $x_7 \leq \frac{1}{2}$. Thus to satisfy $x_4 + x_6 + x_7 = 1\frac{1}{2}$ we have $x_4 = x_6 = x_7 = \frac{1}{2}$, and this forces

$$x_1 + x_2 = x_3 + x_5 = x_1 + x_3 = x_2 + x_5 = x_1 + x_5 = x_2 + x_3 = 1.$$

Thus $\mathbf{x} = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$, and we have the following proposition:

PROPOSITION 2. *A vertex of W in the nonnegative orthant lies on at most one of $\mathbf{v}_i \cdot \mathbf{x} = 2$, $\bar{\mathbf{v}}_i \cdot \mathbf{x} = 1\frac{1}{2}$ unless it is the vertex $\frac{1}{2} = (\frac{1}{2}, \frac{1}{2}, \dots, \frac{1}{2})$ which lies on all faces $\mathbf{v}_i \cdot \mathbf{x} = 2$, $\bar{\mathbf{v}}_i \cdot \mathbf{x} = 1\frac{1}{2}$.*

Suppose the nonnegative vertex $\mathbf{x}$ of W lies on three faces $\mathbf{v}_i \cdot \mathbf{x} = 2$. By applying isometries we may take $\mathbf{x}$ to lie on the faces $\mathbf{x} \cdot \mathbf{v}_1 = 2$, $\mathbf{x} \cdot \mathbf{v}_2 = 2$, $\mathbf{x} \cdot \mathbf{v}_4 = 2$. Then $2(x_2 + x_3) = \mathbf{x} \cdot (\mathbf{v}_1 + \mathbf{v}_2 - \mathbf{v}_4) = 2$, so $x_2 + x_3 = 1$. Similarly $x_1 + x_5 = x_4 + x_6 = 1$. Suppose $x_1 = t$. Then $|\mathbf{x} \cdot \bar{\mathbf{v}}_3| \leq 1\frac{1}{2}$, $|\mathbf{x} \cdot \bar{\mathbf{v}}_5| \leq 1\frac{1}{2}$ imply $x_2 + x_6 \leq 1\frac{1}{2} - t$, $x_3 + x_4 \leq 1\frac{1}{2} - t$, and $2 = x_2 + x_3 + x_4 + x_6 \leq 3 - 2t$, so $t \leq \frac{1}{2}$, that is, $x_1 \leq \frac{1}{2}$. A similar argument shows $x_5 \leq \frac{1}{2}$, so to satisfy $x_1 + x_5 = 1$ we must have $x_1 = x_5 = \frac{1}{2}$. A similar argument shows $x_2 = x_3 = \frac{1}{2}$, $x_4 = x_6 = \frac{1}{2}$. Plainly $x_7 \leq \frac{1}{2}$, so $\mathbf{x}$ lies on the line joining the vertices $(\frac{1}{2}, \frac{1}{2}, \dots, \frac{1}{2})$ and $(\frac{1}{2}, \dots, \frac{1}{2}, -\frac{1}{2})$. Thus we have the following proposition:

PROPOSITION 3. *A vertex of W in the nonnegative orthant lying on at least three faces $\mathbf{v}_i \cdot \mathbf{x} = 2$ is obtained from $\frac{1}{2}$ by applying an isometry of E_7^* .*

Let $\mathbf{x} \cdot \mathbf{l}_1$ be a type 1 face, and $\mathbf{x} \cdot \mathbf{v}_i = 2$, $\mathbf{x} \cdot \mathbf{l}_2 = 2$ two type 2 faces, where either $\mathbf{l}_2 = \mathbf{v}_j$ or $\mathbf{l}_2$ corresponds to $\mathbf{v}_i$. Then $\mathbf{v}_k$ can be chosen to have a 1 where $\mathbf{l}_1$ has a 2, and to be different from $\mathbf{v}_i$, $\mathbf{v}_j$ (if $\mathbf{l}_2 = \mathbf{v}_j$) or to have a 1 where $\mathbf{l}_2$ has a -1 (if $\mathbf{l}_2$ corresponds to $\mathbf{v}_i$). Then the quaternion isometry based on $\mathbf{v}_k$ with a "real" part corresponding to the component where $\mathbf{l}_1$ has a 2 is such that $Q_k(\mathbf{l}_1)$, $Q_k(\mathbf{v}_i)$, $Q_k(\mathbf{l}_2)$ are different $\mathbf{v}_j$. Thus the remaining cases to be considered are, up to an isometry of E_7^* , those where the nonnegative vertex $\mathbf{x}$ lies on no type 1 face; it either lies on just two type 2 faces $\mathbf{x} \cdot \mathbf{v}_1 = 2$, $\mathbf{x} \cdot \mathbf{v}_2 = 2$, or on just one type 2 face $\mathbf{x} \cdot \mathbf{v}_1 = 2$, or on no type 2 face. We show that no vertices $\mathbf{x}$ satisfy this condition.

Suppose the nonnegative vertex $\mathbf{x}$ of W lies on $\mathbf{x} \cdot \mathbf{v}_1 = 2$ and $\mathbf{x} \cdot \mathbf{v}_2 = 2$, and on no other type 2 or type 1 face. Then none of $x_1, \dots, x_6$ can be zero. Now $\mathbf{x}$ must lie on five type 3 faces, which must be among the six faces $\mathbf{x} \cdot \bar{\mathbf{v}}_i = 1\frac{1}{2}$ ($i = 3, \dots, 7$), $x_2 + x_3 - x_7 = 1\frac{1}{2}$. But the five faces cannot be $\mathbf{x} \cdot \bar{\mathbf{v}}_i = 1\frac{1}{2}$ ($i = 3, \dots, 7$), otherwise $\mathbf{x} = \frac{1}{2}$, which lies on all type 2 faces. Thus $x_2 + x_3 = 1\frac{1}{2}$ and $x_7 = 0$, and $\mathbf{x}$ lies on three of $\mathbf{x} \cdot \bar{\mathbf{v}}_i = 1\frac{1}{2}$ ($i = 3, 5, 6, 7$). Since $x_i \leq 1$, we have $x_2 \geq \frac{1}{2}$, $x_3 \geq \frac{1}{2}$, and since none of $x_1, \dots, x_6$ is zero we must have $\mathbf{x} \cdot \bar{\mathbf{v}}_i > \frac{1}{2}$ for $i = 3, 5, 6, 7$. Thus

$$\mathbf{x} \cdot (\bar{\mathbf{v}}_3 + \bar{\mathbf{v}}_5 + \bar{\mathbf{v}}_6 + \bar{\mathbf{v}}_7) > \frac{1}{2} + 3 \cdot (1\frac{1}{2}) = 5.$$

However,

$$\mathbf{x} \cdot (\bar{\mathbf{v}}_3 + \bar{\mathbf{v}}_5 + \bar{\mathbf{v}}_6 + \bar{\mathbf{v}}_7) = 2\mathbf{x} \cdot (\mathbf{v}_1 + \mathbf{v}_2) - 2(x_2 + x_3) = 5,$$

which is a contradiction.

Suppose the nonnegative vertex $\mathbf{x}$ of W lies on $\mathbf{x} \cdot \mathbf{v}_1 = 2$ and on no other type 2 or type 1 face. Then none of x_1, x_2, x_3, x_5 can be zero. The remaining six faces on which $\mathbf{x}$ lies must be among $\mathbf{x} \cdot \bar{\mathbf{v}}_i = 1\frac{1}{2}$ ($i = 2, \dots, 7$) and the faces obtained from these by replacing the coefficient 1 of x_4 or x_6 or x_7 by -1 . (i) If no x_i is zero, then as above we get $\mathbf{x} = \frac{1}{2}$ which lies on too many type 2 faces. (ii) If just one x_i is zero, say, for example, $x_7 = 0$, $\mathbf{x}$ must lie on just one of $\mathbf{x} \cdot \mathbf{v}_2 = 1\frac{1}{2}$, $\mathbf{x} \cdot \mathbf{v}_4 = 1\frac{1}{2}$ and on all of $\mathbf{x} \cdot \bar{\mathbf{v}}_i = 1\frac{1}{2}$, $i = 3, 5, 6, 7$. Thus $\mathbf{x} \cdot \mathbf{v}_1 = \mathbf{x} \cdot (\bar{\mathbf{v}}_5 + \bar{\mathbf{v}}_6) + 2x_4 = \mathbf{x} \cdot (\bar{\mathbf{v}}_3 + \bar{\mathbf{v}}_7) + 2x_6$, so $x_4 = x_6 = \frac{1}{2}$, and $x_1 + x_3 = x_2 + x_5 = x_3 + x_5 = x_1 + x_2 = 1$. But either $x_1 + x_5 = 1\frac{1}{2}$ or $x_2 + x_3 = 1\frac{1}{2}$, so either $\mathbf{x} \cdot \mathbf{v}_4 = 2\frac{1}{2}$ or $\mathbf{x} \cdot \mathbf{v}_2 = 2\frac{1}{2}$. Similar contradictions can be found if just $x_6 = 0$ or just $x_4 = 0$. (iii) If two x_i are zero, say for example $x_6 = x_7 = 0$, then $\mathbf{x}$ must be on just one of $\mathbf{x} \cdot \mathbf{v}_2 = 1\frac{1}{2}$, $\mathbf{x} \cdot \mathbf{v}_4 = 1\frac{1}{2}$, or just one of $\mathbf{x} \cdot \mathbf{v}_3 = 1\frac{1}{2}$, $\mathbf{x} \cdot \mathbf{v}_7 = 1\frac{1}{2}$, and both $\mathbf{x} \cdot \mathbf{v}_5 = 1\frac{1}{2}$, $\mathbf{x} \cdot \mathbf{v}_6 = 1\frac{1}{2}$. As in (ii) $x_4 = \frac{1}{2}$. Furthermore either $x_1 + x_5 = 1\frac{1}{2}$ or $x_2 + x_3 = 1\frac{1}{2}$, and either $x_1 + x_2 = 1\frac{1}{2}$ or $x_3 + x_5 = 1\frac{1}{2}$. Thus $\mathbf{x}$ is one of $(1, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, 0, 0)$, $(\frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2}, 1, 0, 0)$, $(1, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2}, 0, 0)$ or $(0, \frac{1}{2}, 1, \frac{1}{2}, \frac{1}{2}, 0, 0)$, all of which are images of $\frac{1}{2}$ under isometries of E_7^* and lie on too many type 1 and 2 faces. Similar contradictions arise if $x_4 = x_6 = 0$ or $x_4 = x_7 = 0$. (iv) If $x_4 = x_6 = x_7 = 0$ then either $x_1 + x_5 = 1\frac{1}{2}$ or $x_2 + x_3 = 1\frac{1}{2}$, either $x_1 + x_2 = 1\frac{1}{2}$ or $x_3 + x_5 = 1\frac{1}{2}$, and either $x_1 + x_3 = 1\frac{1}{2}$ or $x_2 + x_5 = 1\frac{1}{2}$. These are not compatible with $\mathbf{x} \cdot \mathbf{v}_1 = 2$.

Finally, suppose the nonnegative vertex $\mathbf{x}$ of W lies on no type 1 or type 2 face. (i) If no x_i is zero then as previously $\mathbf{x} = \frac{1}{2}$ which lies on all type 2 faces. (ii) If just one $x_i = 0$, say $x_1 = 0$, then $\mathbf{x}$ can only lie on the faces $\mathbf{x} \cdot \bar{\mathbf{v}}_i = 1\frac{1}{2}$, $i = 1, 4, 6, 7$ and just one pair of faces corresponding to one of $\mathbf{x} \cdot \bar{\mathbf{v}}_2 = 1\frac{1}{2}$, $\mathbf{x} \cdot \bar{\mathbf{v}}_3 = 1\frac{1}{2}$, or $\mathbf{x} \cdot \bar{\mathbf{v}}_5 = 1\frac{1}{2}$ without being on the wrong side of a type 2 face. Six faces do not determine a vertex. Similar arguments apply whichever $x_i = 0$. (iii) Similarly if exactly two x_i are zero, there are not enough faces. (iv) If exactly three x_i are zero, there are two distinct cases. In one case, typified by $x_1 = x_2 = x_3 = 0$, there are three faces $\mathbf{x} \cdot \bar{\mathbf{v}}_3 = 1\frac{1}{2}$, $\mathbf{x} \cdot \bar{\mathbf{v}}_4 = 1\frac{1}{2}$, $\mathbf{x} \cdot \bar{\mathbf{v}}_5 = 1\frac{1}{2}$ on which $\mathbf{x}$ cannot lie, so $\mathbf{x}$ lies on the remaining four. Adding three of these and subtracting twice

the fourth shows that the vertex is of the form $(0, 0, 0, \frac{1}{2}, 1, \frac{1}{2}, \frac{1}{2})$ which lies on a type 2 face. In the other case, typified by $x_1 = x_2 = x_6 = 0$ there are not enough faces to avoid conditions like $x_5 + x_7 = 1\frac{1}{2}$, $x_2 + x_6 = 1\frac{1}{2}$ which put $\mathbf{x}$ on the wrong side of a type 2 face.

Thus we have shown:

PROPOSITION 4. *The vertices of W are obtained from $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ by applying isometries of E^* .*

Finally, since all vertices of E^* have integer coordinates, there is plainly no point of E^* closer to $\frac{1}{2}$ than $\mathbf{0}$, so $\frac{1}{2}$ lies in the Voronoi region of E^* . Hence we have the following theorem:

THEOREM. *The Voronoi region of E^* has faces*

$$(\pm 2, 0, 0, 0, 0, 0, 0) \cdot \mathbf{x} = 2,$$

$$(\pm 1, \pm 1, \pm 1, 0, \pm 1, 0, 0) \cdot \mathbf{x} = 2,$$

$$(0, 0, 0, \pm 1, 0, \pm 1, \pm 1) \cdot \mathbf{x} = 1\frac{1}{2}$$

and their images under powers of C . The vertices are obtained from $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ by applying isometries of E^* , and are of the form $\frac{1}{2}\mathbf{l}$ where $\mathbf{l}$ is a point of E^* of length $\sqrt{7}$ arising from $\bar{\mathbf{v}}_i$ by replacing any 1 by -1 and replacing just one zero by ± 2 .

3. The covering constant and quantizing constant of E^* . The vertices of the Voronoi region have distance $\sqrt{7}/2^{10/7}$ from $\mathbf{0}$ when the lattice is normalised to have determinant 1; so for spheres, E^* provides a lattice covering of density $(\sqrt{7} \cdot 2^{-10/7})^7 J_7 = 0.88622 \dots J_7$ where J_7 is the volume of a sphere in $\mathbb{R}^7$ of radius 1. This can be compared with A^* , which has a covering constant of $(\sqrt{42} \cdot 8^{-13/14})^7 J_7 = 0.64757 \dots J_7$. Although this seems quite marked, the radii of the spheres involved are in the ratio

$$1:(0.64757 \dots / 0.88622 \dots)^{1/7} = 1:0.96 \dots$$

To determine the quantization constant it is necessary to analyze the faces of V . This is straightforward but tedious, since the above theorem gives the faces and vertices of V . There are two nonisomorphic types of face of V : the first type is obtained from $\mathbf{v}_1 \cdot \mathbf{x} = 2$ by reflection, cycling, and quaternion isometries (this covers the first two forms listed in the theorem), and the second type is obtained from $\bar{\mathbf{v}}_1 \cdot \mathbf{x} = 1\frac{1}{2}$ by reflection and cycling isometries. Since the first type of face is rather simple, the second type of face will be used as an example.

There are plainly $56 = 2^3 \cdot 7$ faces of type 2 generated by three independent reflections and the cycle of order 7. We analyse the face F with equation $\bar{\mathbf{v}}_1 \cdot \mathbf{x} = 1\frac{1}{2}$. It plainly has distance $\frac{1}{2}\sqrt{3}$ from $\mathbf{0}$. From the description of the vertices of V given in the theorem, the following vertices lie on F where a, b is a permutation of 1, 0 and c, d, e, f is a permutation of 1, 0, 0, 0:

$(\pm \frac{1}{2}, \pm \frac{1}{2}, \pm \frac{1}{2}, \frac{1}{2}, \pm \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$	from $\bar{\mathbf{v}}_0$,
$(c, d, e, \frac{1}{2}, f, \frac{1}{2}, \frac{1}{2})$	from $\bar{\mathbf{v}}_1$,
$(\pm \frac{1}{2}, 0, 0, a, \pm \frac{1}{2}, b, \frac{1}{2})$	from $\bar{\mathbf{v}}_2$,
$(\pm \frac{1}{2}, \pm \frac{1}{2}, 0, a, 0, \frac{1}{2}, b)$	from $\bar{\mathbf{v}}_3$,
$(0, \pm \frac{1}{2}, \pm \frac{1}{2}, a, 0, b, \frac{1}{2})$	from $\bar{\mathbf{v}}_4$,
$(\pm \frac{1}{2}, 0, \pm \frac{1}{2}, \frac{1}{2}, 0, a, b)$	from $\bar{\mathbf{v}}_5$,

$$(0, \pm \frac{1}{2}, 0, \frac{1}{2}, \pm \frac{1}{2}, a, b) \quad \text{from } \bar{v}_6,$$

$$(0, 0, \pm \frac{1}{2}, a, \pm \frac{1}{2}, \frac{1}{2}, b) \quad \text{from } \bar{v}_7.$$

The five-dimensional edges of F are the intersection with $\bar{v}_1 \cdot \mathbf{x} = 1\frac{1}{2}$ of the other faces of V that contain at least six of the above points. These faces are clearly:

Ti: $2x_i = 2$ ($i = 4, 6, 7$);

Tii: The four faces arising from each of $\mathbf{v}_i \cdot \mathbf{x} = 2$ ($i = 2, \dots, 7$) by independent reflections in the x_1, x_2, x_3, x_5 coordinates (only two of the reflections give new faces each case);

Tiii: The four faces arising from each of $\bar{\mathbf{v}}_i \cdot \mathbf{x} = 1\frac{1}{2}$ ($i = 2, \dots, 7$) by independent reflections in the x_1, x_2, x_3, x_5 coordinates;

Tiv: Three faces arising from $\bar{\mathbf{v}}_1 \cdot \mathbf{x} = 1\frac{1}{2}$ by reflection in exactly one of the coordinates x_4, x_6, x_7 .

There are 54 of these faces: each contains 16 of the above vertices, and the faces are distinct because they contain different sets of vertices. There are isometries of V preserving $\mathbf{v}_1 \cdot \mathbf{x} = 1\frac{1}{2}$ and transforming the types Ti into Tii, and Tiii into Tiv, and the intersections with $\bar{\mathbf{v}}_1 \cdot \mathbf{x} = 1\frac{1}{2}$ of types Ti and Tiv are isomorphic. For example, the quaternion isometry $\mathbf{x} \rightarrow \mathbf{x}'$ where

$$2x'_2 = x_2 - x_3 + x_4 - x_6,$$

$$2x'_3 = -x_2 + x_3 + x_4 - x_6,$$

$$2x'_4 = x_2 + x_3 + x_4 + x_6,$$

$$2x'_6 = -x_2 - x_3 + x_4 + x_6$$

(leaving x_1, x_5, x_7 unchanged) fixes $\bar{\mathbf{v}}_1 \cdot \mathbf{x} = 1\frac{1}{2}$, yet the Ti face $2x_4 = 2$ corresponds to the Tii face $\mathbf{v}_2 \cdot \mathbf{x}' = 2$ and the Tiii face $-x_2 - x_3 + x_7 = 1\frac{1}{2}$ corresponds to the Tiv face $-x'_4 + x'_6 + x'_7 = 1\frac{1}{2}$. To see that the Ti and Tiv faces give isomorphic five-dimensional edges of F it is convenient to project onto F . This is made more convenient if we apply the orthogonal transformation

$$2x_4 = 2Rx'_4 + 2Rx'_6 - 2Rx'_7,$$

$$2x_6 = 2Rx'_4 + (1 - R)x'_6 + (1 + R)x'_7,$$

$$2x_7 = 2Rx'_4 + (-1 - R)x'_6 + (-1 + R)x'_7,$$

where $R = 1/\sqrt{3}$ and other coordinates are unchanged. This gives F the equation $3Rx'_4 = \frac{3}{2}$, that is, $x'_4 = \frac{1}{2}\sqrt{3}$. Projection can now be achieved by setting $x'_4 = \frac{1}{2}\sqrt{3}$ and dropping the x'_4 coordinate. This transforms the Ti face $2x_4 = 2$ into

$$1 + 2R(x'_6 - x'_7) = 2, \quad \text{that is,} \quad x'_6 - x'_7 = \frac{1}{2}\sqrt{3}$$

and the Tiv face $-x_4 + x_6 - x_7 = 1\frac{1}{2}$ becomes

$$\frac{1}{2} - 2R(x'_6 - x'_7) = 1\frac{1}{2}, \quad \text{that is,} \quad x'_6 - x'_7 = -\frac{1}{2}\sqrt{3}.$$

Thus all five-dimensional edges of F are at a distance of $\frac{1}{2}\sqrt{\frac{3}{2}}$ from the centre of F . Showing that these are isomorphic is completed by applying the orthogonal transformation to all the vertices on F , and to all the faces of V containing at least five vertices of F , and projecting onto $x'_4 = \frac{1}{2}\sqrt{3}$.

The nature of the five-dimensional edges of F (and so on for decreasing dimensional and the other six-dimensional faces of V) is analysed similarly. The results are given in

the Table 1. The table entries for the Voronoi region show that E_7^* has quantization constant

$$G = \frac{1}{7} \cdot \frac{21361}{2880} \cdot \frac{1}{8^{9/7}} = 0.0731165^{(-)}$$

confirming the value obtained by Conway and Sloane.

The table also gives the results of calculations using the formulae (see [1]):

(i) For a regular n -simplex S of edge length $2l$ the volume is

$$\mu(s) = (l\sqrt{2})^n \sqrt{n+1}/n!$$

and the unnormalised second moment is

$$U(s) = \mu(s)^{1+2/n} n(n!)^{2/n} / ((n+2)(n+1)^{1+1/n}).$$

(ii) For a polytope P with faces F_i at distance h_i from a centre the volume is

$$\mu(P) = \frac{1}{n} \sum_i h_i \mu(F_i)$$

and the unnormalised second moment is

$$U(P) = \frac{1}{n+2} \sum_i h_i (h_i^2 \mu(F_i) + U(F_i)).$$

Thus, for example, since V (and therefore all its subfaces) have edge length 1, the subfaces that are simplices have

$$\begin{aligned} n = 3: \quad \mu &= \frac{1}{2\sqrt{2}} \cdot \frac{2}{6} = \frac{1}{6\sqrt{2}}, \quad U = \left(\frac{1}{6\sqrt{2}}\right)^{5/3} \cdot 3 \cdot 6^{2/3} / (5 \cdot 4^{4/3}) = \frac{1}{80\sqrt{2}}; \\ n = 4: \quad \mu &= \frac{\sqrt{5}}{96}, \quad U = \frac{\sqrt{5}}{1440}; \\ n = 5: \quad \mu &= \frac{\sqrt{3}}{480}, \quad U = \frac{\sqrt{3}}{8064}. \end{aligned}$$

TABLE 1

dim	Type	Volume	U	Subfaces at distances
3	tetrahedron	$1/6\sqrt{2}$	$1/80\sqrt{2}$	(a simplex)
4	type 8v	$\frac{1}{6}$	$\frac{1}{45}$	16 tetrahedra at $h = 1/2\sqrt{2}$
4	4-simplex	$\sqrt{5}/96$	$1/288\sqrt{5}$	(a simplex)
5	type 16v	$13/60\sqrt{2}$	$83/2016\sqrt{2}$	10 type 8v at $h = 1/2\sqrt{2}$ 16 4-simplex at $h = 3/2\sqrt{10}$
5	5-simplex	$1/160\sqrt{3}$	$1/2688\sqrt{3}$	(a simplex)
6	type 2	$43/360$	$97/3360$	12 type 16v at $h = 1/2\sqrt{2}$ 32 5-simplex at $h = 1/\sqrt{3}$
6	type 3	$39\sqrt{3}/80$	$1851\sqrt{3}/8960$	54 type 16v at $h = \sqrt{3}/2\sqrt{2}$
7	Voronoi	8	$21361/2880$	126 type 2 at $h = 1$ 56 type 3 at $h = \sqrt{3}/2$

For the other faces we build up from the values for lower dimensional faces using the formulae (ii). Thus, for the five-dimensional type 16- v face we have

$$\mu = \frac{1}{5} \left(10 \cdot \frac{1}{2\sqrt{2}} \cdot \frac{1}{6} + 16 \cdot \frac{3}{2\sqrt{10}} \cdot \frac{\sqrt{5}}{96} \right) = \frac{13}{60\sqrt{2}},$$

$$U = \frac{1}{7} \left(10 \cdot \frac{1}{2\sqrt{2}} \left(\frac{1}{8} \cdot \frac{1}{6} + \frac{1}{45} \right) + 16 \cdot \frac{3}{2\sqrt{10}} \left(\frac{9}{40} \frac{\sqrt{5}}{96} + \frac{1}{288\sqrt{5}} \right) \right) = \frac{83\sqrt{2}}{4032}.$$

REFERENCES

- [1] J. H. CONWAY AND N. J. A. SLOANE, *Voronoi regions of lattices, second moments of polytopes, and quantization*, IEEE Trans. Inform. Theory., 28 (1982), pp. 211–226.
- [2] ———, *Fast quantizing and decoding algorithms for lattice quantizers and codes*, IEEE Trans. Inform. Theory, 28 (1982), pp. 227–232.
- [3] ———, *On the Voronoi regions of certain lattices*, SIAM J. Algebraic Discrete Methods, 5 (1984), pp. 294–305.
- [4] N. J. A. SLOANE, *Tables of sphere packings and spherical codes*, IEEE Trans. Inform. Theory, 27 (1981), pp. 327–338.
- [5] R. T. WORLEY, *The Voronoi region of E_6^** , J. Austral. Math. Soc., Ser. A., 48 (1987), pp. 268–278.

QUESTIONS RELATED TO THE ERDŐS-TURAN CONJECTURE*

MARTIN DOWD†

Abstract. New relationships between questions in combinatorial number theory are shown. The Erdős-Turan conjecture is shown to be equivalent to the finite problem; the Erdős-Turan conjecture for N is shown to follow from that for Z ; and the additive number theory of the integers and that of the finite bit strings under bitwise mod 2 addition are contrasted. The dual space technique from coding theory is applied to the integer question, leading to a lower bound on the number of collisions. A subsidiary question in coding theory is raised, which is a refinement of the perfect code theorem. Finally some computations for cyclic codes are given.

Key words. Erdős-Turan conjecture, additive bases, perfect sets, postage stamp problems, linear double error correcting codes

AMS(MOS) subject classification. 11B13

1. Versions of the Erdős-Turan conjecture. If G is an Abelian semigroup and $S \subseteq G$ define $S + S = \{x + y : x, y \in S\}$. Define the cost $\chi(z)$ of an element $z \in G$ to be $|\{\{x, y\} \subseteq S : x + y = z\}|$; this may be ∞ . Define the cost $\chi(S)$ of S to be $\max\{\chi(z) : z \in S + S\}$; this may be ∞ even if all elements of $S + S$ have finite cost. S is said to be a base for G if $G = S + S$; and a perfect or B_2 set if its cost is 1. In the classic case G is the integers N . Questions which have been studied include upper and lower bounds on the asymptotic density of a base or perfect set, and the Erdős-Turan conjecture, which states that the cost of any base is ∞ (see [HR]).

Other infinite G of interest include the integers Z , and the finite bit strings under bitwise mod 2 sum, denoted Z_2^∞ . In these cases the question of whether the cost of some element must be infinite arises as a refinement of the Erdős-Turan conjecture. In the latter case to avoid triviality the element 0 is omitted in the definition of the cost of S . Indeed the k -cost could be defined to be the k th in the sorted list of costs, and more general questions suggest themselves, such as whether the k -cost is unbounded for all k .

These questions may also be asked for finite G , such as the cyclic groups or Z_2^n ; postage stamp problems provide another example. In this case we consider the asymptotic behavior of the cost, size, etc. of the best S for a given G in the family, as the size of G increases. As we will see below, there are close relationships between the finite and infinite questions.

[GS] contains a survey of the classical finite problems. The case of Z_2^n does not seem to have been considered before in this context. There are well-known questions in coding theory which are analogues of the above questions of combinatorial number theory. In this case sums and linear combinations are the same, so that the tools of linear algebra may be applied.

2. Finite and infinite questions. Define a nonempty set $S \subseteq N$ to be a prebase if $[0, m] \subseteq S + S$ where $m = \max S$. Clearly if S is a base then $S_m = \{x \in S : x \leq m\}$

*Received by the editors September 27, 1986; accepted for publication August 31, 1987.

†2532 Orange Ave., Costa Mesa, CA 92627.

is a prebase. Further, if S is of cost $\leq c$ then so is every S_m . Noting that a base is infinite, we have proved half of the following theorem.

THEOREM 2.1. *There is a base of cost $\leq c$ for N iff for each m there is a prebase S of cost $\leq c$ with $\max S \geq m$.*

Proof. Say that prebase T is a son of prebase S if $T = S \cup \{x\}$ for some $x > S$. Consider the tree whose root is the empty set, and where the sons of a node are its son prebases. Any prebase has only finitely many sons so this tree is finitely branching. Further, if S has cost $\leq c$ so does any prebase contained in it; such prebases thus form a subtree. By hypothesis there are arbitrarily large prebases of cost $\leq c$, so this tree has arbitrarily long branches. It therefore has an infinite branch; the union of the prebases on this branch comprises a base, of cost $\leq c$. The other direction was proved above.

THEOREM 2.2. *If the Erdős-Turan conjecture holds for Z it holds for N .*

Proof. Suppose there is a base for N of cost $\leq c$, so that there are arbitrarily large prebases of cost $\leq c$. Form a tree of pairs $(S, [l, u])$ where S is a finite subset of Z , $[l, u] \subseteq S + S$, and $l + u \leq 3$; and $(T, [m, v])$ is a son of $(S, [l, u])$ if $T = S \cup x$ for $x \notin S$ and $[l, u] \subset [m, v]$. This tree is finitely branching, and the pairs where the cost of S is $\leq c$ form a subtree. Further, it is easily seen to have arbitrarily long branches, by "centering" the prebases. Finally, the union of the S of the pairs on an infinite branch comprises a base for Z of cost $\leq c$.

It follows also that the Erdős-Turan conjecture for the cyclic groups implies it for N . It suffices to observe that if S is a prebase of cost $\leq c$ with maximum element m then $S - \{m\}$ is a base for Z_m of cost $\leq 2c$. It follows for Z_2^∞ from that for the Z_2^r . The tree consists of the pairs (S, r) such that $S \subseteq Z_2^r$, with (T, s) a son of (S, r) if $s = r + 1$ and T is obtained from S by appending a leading 0 and adding at least one more element. (An equivalent finite problem is obtained by allowing bases for Z_2^r to be in a larger Z_2^s .)

The finite and infinite versions of other problems may be seen to be related by similar proofs. We content ourselves with one example. Define prebase S to be an ancestor of prebase T if $S \subseteq T$ and the members of $T - S$ are greater than the members of S . Let $\kappa(S) = (|S| - 1)/\sqrt{\max S}$ (or 0 if $S = \{0\}$); and let $\lambda_k(S)$ be the maximum of $\kappa(T)$ for ancestors T of S of size $\geq \min(k, |S|)$.

THEOREM 2.3. *There is a base S with $\limsup \kappa(S_n) \leq c$ iff for some k there are arbitrarily large prebases P with $\lambda_k(P) \leq c$.*

Proof. First, if $\lambda_k(P) \leq c$ and Q is an ancestor of P then $\lambda_k(Q) \leq c$. Second, if $\lambda_k(S_n) \leq c$ for all n then $\limsup \kappa(S_n) \leq c$. One direction follows, and the other is immediate.

3. Coding theory and the case Z_2^n . In order to motivate what follows, we first consider the case Z_2^n . Here an extensive theory has been developed [MS], which we will adapt to more general problems. If $S \subseteq Z_2^n$ and $0 \in S$, consider the members of $S - \{0\}$ to be the rows of an $n \times r$ matrix H . We suppose that $n \geq r$ and H is full rank. The left nullspace of H is then a single error correcting linear code of length n and redundancy r .

If $x \in Z_2^n$ let $N_t(x) = \{y : |y - x| \leq t\}$, where $|x|$ is the weight of x , i.e., $|x|$ when x is considered as the set of positions where it is 1. If $C \subseteq Z_2^n$ is a code, let $D_t(x) = |N_t(x) \cap C| = |\{y \in C : |x - y| \leq t\}|$. Then $D_t(x) = |\{y \in x + C : |y| \leq t\}|$. Define C to be a t -cover if $D_t(x) \geq 1$ for all $x \in Z_2^n$, and a t -packing (more usually called a t error correcting code) if $D_t(x) \leq 1$ for all $x \in Z_2^n$. It is readily seen that C is a t -cover iff $\bigcup_{x \in C} N_t(x) = Z_2^n$, and C is a t -packing if $N_t(x) \cap N_t(y) = \emptyset$ for $x, y \in C$,

$x \neq y$. Also C is a t -packing iff the minimum weight of C is $\geq 2t + 1$.

THEOREM 3.1. *Let C be the null space of H , where H corresponds to a set $S \subseteq Z_2^r$. C is a t -cover iff every $w \in Z_2^r$ is a sum of t members of S . C is a t -packing iff every $w \in Z_2^r$ is a sum of t members of S in at most one way.*

Proof. Vectors $x, y \in Z_2^n$ are in the same coset of C iff $xH = yH$. Also $w \in Z_2^r$ is a sum of t members of S iff $w = xH$ for some $x \in Z_2^n$ with $|x| \leq r$. Finally, since H has rank r , for every $w \in Z_2^r$ there is an $x \in Z_2^n$ such that $w = xH$. The theorem follows easily from these facts.

THEOREM 3.2. *The cost of S equals $\max\{D_2(x) : x \in Z_2^n, |x| \leq 2\}$.*

Proof. Suppose S is cost c ; then there are distinct $w_i \in S$ such that

$$w_1 + w_2 = w_3 + w_4 = \cdots = w_{2c-1} + w_{2c},$$

where one w_i (say w_1) may be 0. Let $x \in Z_2^n$ be 1 in the w_1 and w_2 (or just w_2) positions; then $D_2(x) = c$. Conversely if there is a vector x with $|x| = 2$ and $D_2(x) = c$, the nonzero codewords within distance 2 of x are of weight 3 or 4; two of them intersect in exactly the positions of x ; only one can be weight 3; and there are $c - 1$ of them. These yield $c - 1$ additional singletons or pairs summing to xH . The proof if $|x| = 1$ is similar, and $D_2(0) = 0$.

This theorem generalizes to higher order t , provided S is assumed to be a $(t - 1)$ -packing (the cost of an element being defined as the number of sets of size $\leq t$ which sum to it). C is said to be perfect if it is both a t -packing and a t -cover; it is well known that the only nontrivial perfect 2-cover has $n = 5$. C is said to be quasi-perfect if it is both a $(t - 1)$ -packing and a t -cover. A t -packing is said to be maximal if S cannot be extended.

THEOREM 3.3. *A t -packing C is maximal iff C is a $(2t - 1)$ -cover.*

Proof. C is maximal iff adding any vector $w \in Z_2^r$ to S yields a dependency of weight $\leq 2t$. This dependency must involve w , so this is the case iff $w = xH$ for some x with $|x| \leq 2t - 1$.

Since we assume all codes are 1-packings, 2-covers are quasi-perfect. Also a 2-packing is maximal iff it is quasi-perfect. Most of the facts stated in this section have probably appeared somewhere in the enormous volume of coding theory literature; they are stated here because they clarify the relationship between the "additive number theory" of the rows of the parity check matrix, and the code.

4. A decomposition of codes. Given a code C and a set W of nonzero weights, define the W -subspace of C to be the subspace generated by $\{x : |x| \in W\}$. Say that C is a W -code if it equals its W -subspace. Define a W -set to be a set of vectors whose weights are among W . Define a W -set to be closed if it contains every vector in its span whose weight is in W ; independent if it is linearly independent; and define its closure to be the least closed W -set containing it.

A set S of vectors in Z_2^r forms an incidence structure, with points the coordinate positions and lines the vectors. A line contains a point if the vector is 1 in the position. An incidence structure may be shortened by deleting a set of points and the lines containing any of them; the result is called a residual structure. Note that a residual structure of a W -set is a W -set; if the original set was closed so is the residual structure. Also any (minimum weight ≥ 3) code is a residual structure of the Hamming code.

We easily verify that the Hamming code of redundancy r (where S is all vectors) is a $\{3\}$ -code. Considering the members of S as the points of $\text{PG}(r - 1, 2)$, the weight 3 vectors are the lines.

THEOREM 4.1. *The independent $\{3\}$ -sets form a matroid.*

Proof. Let I be a linearly independent set of weight 3 vectors, which is not full rank. Suppose $x \notin \text{span}(I)$. Let x_1, x_2 be rows of the parity check matrix corresponding to positions where x is 1. If the vector with ones in these positions and the position corresponding to $x_1 + x_2$ is not in $\text{span}(I)$ we are done; otherwise adding it to x reduces the weight of x .

THEOREM 4.2. *A maximal 2-packing is a $\{5, 6, 7\}$ code.*

Proof. By Theorem 3.3 it is a 3-cover. Given a dependency α of weight $w > 7$, say $x_1 + \dots + x_w = 0$, we show it is the sum of a dependency β of weight 5, 6, or 7; and a dependency γ of weight v with $5 \leq v < w$. For $x_1 + x_2 + x_3 + x_4$ is a sum of at most 3 members of S ; this yields β , and $\gamma = \alpha + \beta$.

Again, this theorem can be generalized to higher order t ; the code is generated by its vectors of weight $\leq 4t - 1$. The weight 5 and 6 vectors determine the redundancy of a maximal 2-packing. To see this, call the collection C_x of pairs or triples which sum to x the covering complex of x . Say that C_x is of type i if it contains i triples, and of type i^* if it contains a pair and $i - 1$ triples. If S is maximal, the covering complexes are in 1-1 correspondence with the points of $Z_2^r - \{0\} - S$; the redundancy is thus in fact determined by the number of covering complexes. But the types of the covering complexes are determined by the weight 5 or 6 vectors of the code and conversely. Indeed, with each covering complex of type i or i^* is associated $\binom{i}{2}$ weight 5 or 6 vectors; each weight 5 or 6 vector appears in 10 such collections.

THEOREM 4.3. *Let C be a code, and let W be a set of nonzero weights. Then C is the sum of its W subspace, and a code which contains no vector of weight $w \in W$.*

Proof. Choose a basis S for the W -subspace, and let T be such that $S \cup T$ is a basis for C . Then C is the sum of $\text{span}(S)$ and $\text{span}(T)$; further $\text{span}(T)$ contains no $x \in C$ with $|x| \in W$, since all such x are in $\text{span}(S)$.

THEOREM 4.4. *If $V \subseteq W$, D is the W -subspace of C ; and E is the V -subspace of D then E is the V -subspace of C .*

Proof. D contains the vectors $x \in C$ with $|x| \in V$.

For example, a minimum weight 3 code may be written as $D + E + F$ where D is the $\{3\}$ -subspace, $D + E$ is the $\{3, 4\}$ -subspace, and F is a code of minimum weight 5; also D is a $\{3\}$ -code of minimum weight 3 and E is a $\{4\}$ -code of minimum weight 4. Let α be the dimension of D , and let β be the dimension of $D + E$. Trivially $\alpha \leq \beta \leq k = n - r$.

THEOREM 4.5. *A base S contains a perfect set of size $\geq |S| - \beta$.*

Proof. Choose a basis for the $\{3, 4\}$ -subspace and expand it to a basis for C using members of the basis comprising the rows of the matrix containing the identity in columns 1 to k , and H in rows $k + 1$ to n . The members of H in the rows of this basis which are used, together with 0 and the unit vectors, comprise the required perfect set.

Unfortunately, little is known about the maximum size of a perfect set in this case, i.e., the maximum length of a linear double error correcting code of redundancy r . From the sphere packing bound it is bounded above by $\sqrt{2} \cdot 2^{r/2}$, neglecting lower order terms. The best known codes, however, have length $2^{r/2}$ [MS]. It is known from the perfect code theorem that the sphere packing bound is not attained for $r > 4$, but this seems to be the best almost everywhere correction known. We will discuss this problem further in §6.

Also, little is known about the minimum size of a base. There seems to be nothing in the literature. Well-known bases in the integer case can be adapted. For example,

taking r even, let S consist of the bit strings which are 1 only in the first $r/2$ positions, or the second $r/2$ positions. This yields an upper bound on the minimum size of $2 \cdot 2^{r/2}$; we conjecture that, as in the integer case [HH] this can be improved. By the sphere covering bound the size is bounded below by $\sqrt{2} \cdot 2^{r/2}$, neglecting lower order terms.

5. The null space of a prebase. The null space of a prebase may be defined as the lattice of dependencies with coefficients in Z . More specifically, given $S \subseteq N$ arrange the nonzero elements in a column H . The set $\{x \in Z^n : x \cdot H = 0\}$ forms an integer lattice C of dimension $n - 1$ in Z^n , where $n = |S - \{0\}|$. We assume that $\gcd(S - \{0\}) = 1$, and that H is in increasing order.

Let Δ^+ be the vectors in which the nonzero entries are a single 1, a single 2, or 2 1's; let $\Delta^- = -\Delta^+$. Let Γ be the possible dependencies obtained by adding a member of Δ^+ and a member of Δ^- . In these, the nonzero coordinates are disjoint, and two single 1's or two single 2's do not occur. Let β be the dimension of the subspace generated by Γ .

THEOREM 5.1. *Suppose C is the left nullspace of H . Then S is a prebase iff $\bigcup_{x \in C} x + \Delta^+ = Z^n$. The cost of S is $\max\{|(x + \Delta^-) \cap C| : x \in \Delta^+\}$.*

Proof. Since the gcd is 1, S is a prebase iff for all $x \in Z^n$ there is $y \in \Delta^+$ such that $xH = yH$; the first claim follows. The second is immediate from the definitions.

THEOREM 5.2. *A prebase S contains a perfect set of size $\geq |S| - \beta$.*

Proof. Let $H = \{h_1, \dots, h_n\}$. Call a $c \in C$ simple if it is of the form $c_1 = -h_{i+1}$, $c_{i+1} = 1$, $c_j = 0$ otherwise, for some i . We construct an upper triangular base for the lattice C consisting of simple vectors and Γ -vectors. Assume rows 1 to $i - 1$ have been constructed. If $i + 1$ is not the highest numbered column of a Γ vector, add the simple vector with 1 in column $i + 1$; otherwise choose any such Γ vector.

In this case there are good bounds on the size of a perfect set and of a prebase, in terms of the maximum element m . A perfect set can contain at most $(1 + o(1))\sqrt{m}$ elements, and this is tight [HR]. A prebase must contain at least $1.45\sqrt{m}$ elements [K] (the best known prebases contain $1.9\sqrt{m}$ elements [HH]). It follows that the maximum size of a perfect subset of a prebase is $\leq .7n$; and using Theorem 5.2 that $\beta \geq .45\sqrt{m}$. It is easy to see that β is a lower bound on $\sum_{x \in S+S} (\chi(z) - 1)$; we conjecture that the latter is bounded below by a constant times m . It follows from the results of [K] that $|\{z \in S + S : z > m\}|$ is bounded below by a constant times m .

Let H' be H with the row containing m removed, and let $C' = \{x \in Z^{n-1} : x \cdot H' \equiv 0 \pmod{m}\}$.

THEOREM 5.3. *The volume of C' is m .*

Proof. Construct inductively an $n - 1 \times n$ matrix rows are a basis for C' ; the first $n - 1$ columns will form an upper triangular matrix whose rows are a basis for C' , of determinant m . For $n = 2$ $C_{11} = m$ and $C_{12} = -h_1$. For $n > 2$ let $d = \gcd(\{h_2, \dots, h_n\})$; let D be the matrix derived inductively from $\{h_2/d, \dots, h_n/d\}$. Let b_i be such that $\sum_2^n b_i h_i = d$. Then $C_{11} = d$, $C_{1i} = -b_i h_1$ for $i > 1$, $C_{i1} = 0$ for $i > 1$, and the remainder of C is D .

6. Remarks on linear double error correcting codes. The best hope for an improvement to the sphere packing bound for linear double error correcting codes seems to be the Johnson bound [J]. The linear programming bound has to date [RV] yielded no improvement, and the Preparata codes show that it cannot yield much of one. For linear double error correcting codes the Johnson bound may be stated

$$1 + n + \binom{n}{2} + \frac{\binom{n}{3} - 10A_L(n)}{\lfloor n/3 \rfloor} \leq 2^r,$$

where $A_L(n)$ is the maximum size of the collection of weight 5 vectors in a linear double error correcting code of length n . Since the weight 5 and 6 vectors determine r (for maximal 2-packings), the Johnson bound may not be a gross overestimate.

The sphere packing bound corresponds to the bound

$$\frac{1}{10} \binom{n}{3}$$

on $A_L(n)$. Note that equality is achieved iff the weight 5 vectors form a $3-(n, 5, 1)$ design. There is a well-known [D] infinite family of such designs, with $n = 4^d + 1$; it has been shown [O'N] that these generate the code of redundancy 0. Indeed, it follows from the perfect code theorem that the weight 5 vectors cannot form such a design; since every triple occurs, every sum of a triple is a sum of a pair and it follows that the code is perfect.

One possible approach to obtaining a bound on $A_L(n)$ is to consider derived designs of the weight 5 vectors. Define an A configuration to be the set of weight 5 vectors of a linear double error correcting code. Define a B configuration to those members of an A configuration containing a given position x . Given a line $L \in A$ and a position $x \in L$ the C configuration determined by them is defined to be those $K \in A$ other than L containing x and some other position of L . We define $A_L(n)$ (resp., $B_L(n)$, $C_L(n)$) to be the maximum size of an A (resp., B , C) configuration in a linear double error correcting code of length n . Clearly $A_L(n) \leq nB_L(n)/5$.

LEMMA 6.1. $B_L(n) \leq (n-1)(C_L(n) + 4)/16$.

Proof. Arrange the vectors of the B configuration in an incidence matrix, and delete the all 1's column. Let p_i be the number of 1's in column i ; let b be the number of rows; and let q_j be the number of rows having a 1 in common with row j (note that two rows have at most one 1 in common). Then $\sum p_i = 4b$ and $\sum q_j = 2 \sum \binom{p_i}{2}$; $\sum p_i$ is maximized, subject to $\sum q_j \leq bC_L(n)$, or $2 \sum \binom{p_i}{2} \leq (C_L(n)/4) \sum p_i$, when all p_i are equal (to p , say), giving $2(n-1) \binom{p}{2} \leq C_L(n)(n-1)p/4$, or $p \leq C_L(n)/4 + 1$. The lemma follows immediately.

LEMMA 6.2. *Given any $x \in V$, at most one pair $\{a, b\} \subseteq S$ sums to x , and the triples $\{a, b, c\} \subseteq S$ which sum to x , together with the pair if any, are disjoint.*

Proof. If, for example, $x = a + b = c + d$ then $a + b + c + d = 0$; the other cases are similar.

A C configuration consists of 4 classes of vectors C_i , $1 \leq i \leq 4$, where if $L = \{x, y_1, y_2, y_3, y_4\}$ then C_i is those vectors containing x and y_i . Let $C'_i = \{K - \{x, y_i\} : K \in C_i\}$; by Lemma 6.2 the triples in C_i are disjoint (i.e. C_i is a covering complex). It is immediate that $C_L(n) \leq 4(n-5)/3$; this yields the trivial bound on $A_L(n)$. It is an interesting question whether this bound is attained infinitely often.

THEOREM 6.1. *Given collections C'_i , $1 \leq i \leq 4$, of disjoint weight 3 vectors, these are the classes C'_i of a C configuration iff for each subset $X \subseteq \bigcup C'_i$, with $i_j = |C_j \cap X|$,*

- (1) $\text{weight}(\sum X) \geq 5$ if all i_j are even and the sum is nonzero;
- (2) $\text{weight}(\sum X) \geq 4$ if 3 or 4 i_j are odd;
- (3) $\text{weight}(\sum X) \geq 3$ in any case.

Proof. Let Y be the set of weight 5 vectors from which X is derived; let w be the weight of $\sum Y$ in the positions of L ; and let $a = |\{j : i_j \text{ odd}\}|$. Then w is determined by a , as follows.

$$\begin{array}{l} a : 0 \ 1 \ 2 \ 3 \ 4 \\ w : 0 \ 2 \ 2 \ 4 \ 4 \end{array}$$

The theorem now follows easily.

Clearly the bound on $C_L(n)$ can be attained only if $n - 5 \equiv 0 \pmod{3}$; in fact we must have $n - 5 \equiv 0 \pmod{6}$. To see this, observe that if $n - 5 \equiv 3 \pmod{6}$, at most one class C'_i can contain $(n - 5)/3$ triples, else the conditions of Theorem 6.1 would be violated by deleting two intersecting triples.

The configuration

$$\{a, b, e\}, \{c, d, f\} \in C'_1; \{a, c, g\}, \{b, d, h\} \in C'_2$$

is not allowed. Thus the triples of a class C'_i , together with the remaining positions outside L if any, may be considered as points, with the lines of another class C'_j being lines through them. This collection of points and lines must satisfy:

- (1) each line contains 3 points;
- (2) two lines intersect in at most one point;
- (3) each point lies on at most 3 lines.

Such configurations where each point lies on 3 lines can be characterized as follows. Let a and b be permutations of $\{0, \dots, p - 1\}$, and consider the table whose rows are the sets $\{i, a(i), b(i)\}$. This is a configuration of the required type iff the following permutations are derangements: $a^{-1}b, a^2, b^2, ab^{-1}a, ba^{-1}b, ab, ba, b, a, b^{-1}a^2, a^{-1}b^2$, and $b^{-1}ab^{-1}a$. (There are redundancies in this list, since a is a derangement if a^2 is.) For $p \geq 7$ an example is provided by $a = \lambda x(x + 1) \pmod{p}$ and $b = \lambda x(x + 3) \pmod{p}$. It can be verified that when p is even this example generates no nonzero vectors of weight < 5 .

7. Covering complex size in the BCH codes. The sizes of the covering complexes of the double error correcting BCH codes can be computed, following [GPZ]. Let $q = 2^j$ where the code has redundancy $2j$. We seek the solutions in $\text{GF}(q)$ to

$$(I) \quad \begin{aligned} x_1 + x_2 + x_3 &= s_1 \\ x_1^3 + x_2^3 + x_3^3 &= s_3 \end{aligned} \quad s_3 \neq s_1^3.$$

Note that two x_i cannot be equal, so the solutions fall into sextuples under permuting the values. The sextuples for (s_1, s_3) correspond to the complex for (s_1, s_3) . Call those where $x_i = 0$ for some i type A, and the remainder type B.

Letting $s = s_1^3 + s_3$ and $y_i = x_i + s_1$ the above system becomes

$$(II) \quad \begin{aligned} y_1 + y_2 + y_3 &= 0 \\ y_1^3 + y_2^3 + y_3^3 &= s \end{aligned} \quad s \neq 0.$$

The y_i are distinct and nonzero. To each sextuple of solutions to system II there corresponds 3 type A sextuples and $q - 3$ type B sextuples of solutions to system I.

For s fixed the solutions to II are given in terms of a parameter p , namely

$$y_1^3 = s/(p^2 + p) \quad y_2 = py_1 \quad y_3 = (1 + p)y_1 \quad p \neq 0, 1,$$

where p is legal for s if $s/(p^2 + p)$ is a cube. Distinct values of p yield distinct solutions, for if $y'_1 = y_1$ and $y'_2 = p'y'_1 = py_1 = y_2$ then $p' = p$.

If j is odd every value of p is legal, and each legal p determines one solution. Note that $6|(q - 2)$. For given s there are $(q - 2)/6$ sextuples of solutions to system II. Thus there are

$$\begin{aligned} \binom{q-1}{2} & \quad (q-2)/6^* \text{ complexes, and} \\ (q^2-1)/2 & \quad (q-2)/6 \text{ complexes.} \end{aligned}$$

If j is even let ω be a primitive cube root of unity. The values p , $1/p$, $1+p$, $1/(1+p)$, $(1+p)/p$, $p/(1+p)$ are distinct unless $p = \omega$ or $p = \omega^2$, in which case $p = 1/(1+p) = (1+p)/p$ and $1/p = 1+p = p/(1+p)$. Further, if one is legal they all are, and a solution for one value is a permutation of some solution for another. For each legal p there are 3 solutions. The 3 solutions for ω , together with those for ω^2 , comprise a sextuple. For other values of p , the solutions belong to different sextuples.

Say that x is in class i when $x = \alpha^{3k+i}$, $i = 0, 1, 2$, where α is a primitive root. Note that p is legal for s iff $p^2 + p$ is in the same class as s . Let $t = |\{p : p^2 + p \text{ is class } 0\}|$; then if $p^2 + p$ is class i , $i = 1, 2$, then $p^4 + p^2$ is class $3-i$. Hence $|\{p : p^2 + p \text{ is class } i\}|$ is $(q-t-2)/2$ for $i = 1, 2$. There are $1 + (1/6)3(t-2)$, or $t/2$ sextuples for an s of class 0, and $(1/6)3(q-t-2)/2$, or $(q-t-2)/4$ for an s of class 1 or 2. There are thus

$$\begin{aligned} q-1 & \quad t/2^* \text{ complexes;} \\ (1/3)(q-3)(q-1) & \quad t/2 \text{ complexes;} \\ 2(q-1) & \quad (q-t-2)/4^* \text{ complexes; and} \\ (2/3)(q-3)(q-1) & \quad (q-t-2)/4 \text{ complexes.} \end{aligned}$$

For example, when $r = 8$ $t = 2$ (in general $2 \leq t \leq (q-1)/3$), so there are 15 1^* complexes, 65 1 complexes, 30 3^* complexes, and 130 3 complexes. It is interesting to compare this with the cyclic code of length 17; by computer one determines that there are 68 3^* complexes, 68 3 complexes, 68 4^* complexes, and 34 4 complexes. It is interesting to note that there is a 5^* complex with $r = 8$; it does not occur in either of these codes.

8. Computational results. The maximum size of a cost 2 prebase is 12, and there is a unique such prebase, namely

$$\{0, 1, 2, 4, 5, 7, 11, 16, 19, 24, 32, 40\}.$$

There are 30 values z with $\chi(z) = 1$, and 24 with $\chi(z) = 2$. For this prebase $\beta=10$, since each element > 1 is the largest of a dependency in Γ . There is a unique perfect subset of size 6, namely $\{0, 1, 4, 11, 16, 24\}$.

To the author's knowledge it is open whether there is a limit to the size of a cost 3 prebase. A computer search has been conducted through part of the search space, and the largest cost 3 prebase found so far contains 33 elements. About 50 such have been found, in different parts of the search space.

Acknowledgments. Dean Sturtevant pointed out that the set of dependencies is a linear space. Doug Eardley provided some computer time.

REFERENCES

- [D] P. Dembowski, *Finite Geometries*, Springer-Verlag, Berlin, 1968.
- [GPZ] D. C. Gorenstein, W. W. Peterson, and N. Zierler, *Two-error correcting Bose-Chaudhuri codes are quasi-perfect*, Inform. and Control, 3 (1960) pp. 291-294.
- [GS] R. L. Graham and N. J. A. Sloane, *On additive bases and harmonious graphs*, SIAM J. Alg. Disc. Meth., 1 (1980) pp. 382-404.
- [HH] N. Hammerer and G. Hofmeister, *Zu einer Vermutung von Rohrbach*, J. Reine Angew. Math., 286/287 (1976) pp. 239-247.

- [HR] H. Halberstam and K. F. Roth, *Sequences*, Springer-Verlag, Berlin, 1982.
- [J] S. M. Johnson, *A new upper bound for error-correcting codes*, IEEE Trans. Info. Theory 8, (1962) pp. 203–207.
- [K] W. Klotz, *Eine obere Schranke für die Reichweite einer Extremalbasis zweiter Ordnung*, J. Reine Angew. Math., 238 (1969) pp. 161–168.
- [MS] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error Correcting Codes*, North-Holland, Amsterdam, 1983.
- [O’N] M. O’Nan, personal communication.
- [RV] C. Roos and C. de Vroedt, *Upper bounds for $A(n,4)$ and $A(n,6)$ derived from Delsarte’s linear programming bound*, Discrete Math., 40 (1982) pp. 261–276.

A SET SYSTEM POLYNOMIAL WITH COLOURING AND RELIABILITY APPLICATIONS*

JASON I. BROWN† AND CHARLES J. COLBOURN‡

Abstract. In order to relate the chromatic and all-terminal reliability polynomials, a simple two-variable polynomial is introduced. The latter polynomial is defined on a set system; as a result, many similarities between the chromatic and all-terminal reliability polynomials can be derived. In fact, within this general framework it is often possible to generalize from one of these two graph polynomials in order to get new results for the other.

Key words. reliability polynomial, chromatic polynomial, Whitney polynomial, unimodal sequence, logarithmic concavity, matroid, broken circuit complex, shellable complex

AMS(MOS) subject classifications. 68E10, 05C30

1. Preliminaries. We follow [4] in general for graph theoretical terminology. All graphs are undirected and finite, containing no loops or multiple edges. For a graph G , n and e denote, respectively, the number of vertices and edges of G . H is a (spanning) subgraph of G (written $H < G$) if and only if $V(H) = V(G)$ and $E(H) \subseteq E(G)$.

A *set system* (or hypergraph) S is a set of vertices $V(S)$ and a collection $B(S)$ of subsets of S (called blocks); S is a *complex* if, in addition, $B(S)$ is nonempty and whenever $B \in B(S)$, $A \subseteq B$, then $A \in B(S)$. The *rank* of S , $r(S)$, is the maximum cardinality of a block in S ; S is said to be *purely* $(r - 1)$ -dimensional if every maximal block (i.e., *base*) of S has size $r = r(S)$. The size of $V(S)$ is denoted by $v = v(S)$. Three set systems play a large role in our paper. Given a graph $G = (V, E(G))$, the *graphic matroid* of G , $\text{Gr}(G)$, has vertex set $E(G)$ and $H \in B(\text{Gr}(G))$ if and only if H is an acyclic subgraph of G (see [18] for standard matroid terminology). The *cographic matroid* of G , $\text{Cog}(G)$, also has vertex set $E(G)$, but $H \in B(\text{Cog}(G))$ if and only if $E(G) - H$ contains a spanning tree of G (here we assume G is connected, so $\emptyset \in B(\text{Cog}(G))$). It is well known that the graphic and cographic matroids are purely $(n - 2)$ - and $(e - n)$ -dimensional complexes, respectively. The final set system is the *broken circuit complex* $\text{Br}(G; <)$ with vertex set $E(G)$. Let $<$ be a fixed linear order of $E(G)$, and call a $S \subseteq E(G)$ a *broken circuit* if and only if there is an element $x \in E(G) - S$ such that $\{x\} \cup S$ is minimally not in $E(\text{Gr}(G))$ and $x < y$ for all $y \in S$ (i.e., x is the least element of the cycle $\{x\} \cup S$ in G). The blocks of $\text{Br}(G, <)$ are all $S \subseteq E(G)$ that do not contain a broken circuit. It is known [19] that $\text{Br}(G, <)$ is a purely $(n - 2)$ -dimensional complex.

The last items of terminology are the graph polynomials. The chromatic polynomial of G , $\pi(G; \lambda)$, is the number of ways to colour the vertices of G with λ colours so that adjacent vertices receive different colours. Whitney's classic theorem [13] on chromatic polynomials states that the coefficient of λ^{n-i} in $\pi(G; \lambda)$ is precisely $(-1)^i$ times the number of blocks of size i in $\text{Br}(G; <)$ (for any fixed linear order $<$ on $E(G)$).

Given a nonempty subset of K of $V(G)$, the *K-terminal reliability polynomial* of G , $\text{Rel}_K(G; p)$, is equal to $\sum p^{|E(H)|} (1 - p)^{e - |E(H)|}$, where the sum is taken over all $H < G$ that contain a *Steiner K-tree* (i.e., a minimal subgraph of G that connects all vertices of K). For any $p \in [0, 1]$, $\text{Rel}_K(G; p)$ is equal to the probability that all vertices of K can communicate in G when each edge of G operates independently with probability p . For

* Received by the editors October 27, 1986; accepted for publication (in revised form) December 9, 1987. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada.

† Department of Mathematics, University of Toronto, Toronto, Ontario Canada M5S 1A1.

‡ Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario Canada N2L 3G1.

$K = V(G)$, we write $\text{Rel}_{V(G)}(G; p)$ as $\text{Rel}_A(G; p)$ and call it the *all-terminal reliability polynomial*. Given $K \subseteq V(G)$, we let F_K be the complex with vertex set $E(G)$ and $H \in B(F_K)$ if and only if $E(G) - H$ contains a Steiner K -tree; note that $F_A = F_{V(G)}$ is the same as $\text{Cog}(G)$. The essential observation is that the coefficient of $p^{e-i}(1-p)^i$ in $\text{Rel}_K(G; p)$ is the number of blocks of size i in F_K [7].

We remark here that the all-terminal reliability and chromatic polynomials can be expressed in terms of a well-known two-variable graph polynomial $R(G; x, y)$, called the *rank polynomial* (see [5], for example). It is also well known that reliability polynomials and chromatic polynomials can be written as evaluations of the Tutte polynomial of the graph [3], [8], and this yields a number of results concerning chromatic theory and network reliability. In this paper, we examine a simpler two-variable polynomial and use it to translate results from the colouring to the reliability domain, and vice versa. We examine two widely studied questions. The first asks for bounds on the coefficients of the polynomial. In the reliability case, this has immediate practical application in developing reliability bounds [7]. In the chromatic case, this has been studied by Wilf [19] in examining which polynomials are chromatic. The second question addresses conjectures that the sequence of coefficients is unimodal, or log concave. This question has been studied in the chromatic case, and our formulation enables us to derive similar results for reliability. In the process of relating colouring and reliability via the two-variable polynomial, we also describe an explicit shelling for the broken circuit complex of a graph, and we establish the equivalence of a number of forms of the reliability and chromatic polynomials.

2. The α -polynomial. Given a set system S , we set $v = v(S)$ and define a polynomial in variables p and q :

$$(*) \quad \alpha(S; p, q) = \sum_{X \in B(S)} q^{|X|} p^{v - |X|}.$$

Both the K -terminal reliability and chromatic polynomials are essentially evaluations of α -polynomials for purely dimensional complexes:

LEMMA 2.1. *For any graph G , $K \subseteq V(G)$, and $<$ a fixed ordering of $E(G)$, we have*

- (i) $\text{Rel}_K(G; p) = \alpha(F_K; p, 1 - p)$ and
- (ii) $\lambda^{e-n} \pi(G; \lambda) = \alpha(\text{Br}(G; <); \lambda, -1)$.

We first note a general reduction theorem. For a set system S and vertex $f \in V(S)$, we define two new set systems $S - f$ and $lk(S; f)$ on vertex set $V(S) - \{f\}$ as follows:

- (i) $H \in B(S - f)$ if and only if $H \in B(S)$ and $f \notin H$;
- (ii) $H \in lk(S; f)$ if and only if $f \notin H$ and $H \cup \{f\} \in B(S)$.

THEOREM 2.2. *If S is any set system, and f is a vertex of S , $\alpha(S; p, q) = p \cdot \alpha(S - f; p, q) + q \cdot \alpha(lk(S; f); p, q)$.*

Note that for a graph G and edge $f \in E(G)$, that $F_{K(G)} - f = F_{K \cdot f}(G \cdot f)$, and $lk(F_K; f) = F_K(G - f)$, where $G - f$ arises from G by deleting edge f , $G \cdot f$ by identifying the end points of f and $K \cdot f$ is the image of K under this contraction. Similarly, Brylawski [6] has shown that $\text{Br}(G; <) - f = \text{Br}(G - f; <)$ and $lk(\text{Br}(G; <); f) = \text{Br}(G \cdot f; <)$. Thus these observations together with Theorem 2.2 yield the standard reduction theorem for chromatic polynomials (see [13], for example) and the factor theorem [14] for reliability polynomials.

We shall call the form of α given by $(*)$ its *F-form*, and now introduce a new, useful form for some set systems. If S is a complex and L and U are blocks of S with $L \subseteq U$, then the set $[L, U] = \{S \mid L \subseteq S \subseteq U\}$ is called an *interval* of S . S is *partitionable* (see [1]) if it is purely dimensional and its blocks can be partitioned into intervals

$[L_1, U_1], \dots, [L_k, U_k]$ where each U_i is a base (i.e., maximal block) of S . A complex S is *vertex decomposable* (see [7], for example) if it is purely d -dimensional for some d and either (i) $B(S)$ is the set of all subsets of $V(S)$, or (ii) there is a vertex $v \in V(S)$ such that $S - f$ is purely d -dimensional and vertex decomposable and $lk(S; f)$ is purely $(d - 1)$ -dimensional and vertex decomposable. Every vertex decomposable complex is partitionable (see [7], for example), so as both $F_A = \text{Cog}(G)$ and $\text{Br}(G; <)$ are vertex decomposable, they are partitionable [12].

If S is partitionable with partition $[L_1, U_1], \dots, [L_k, U_k]$, then as any interval $[L, U]$ adds $q^{|L|}p^{v-|U|}(p+q)^{|U|-|L|}$ to $\alpha(S; p, q)$, we have

$$\alpha(S; p, q) = \sum_{i=0}^r H_i q^i p^{v-r}(p+q)^{r-i},$$

where $H_i = |\{L_j: |L_j| = i, 1 \leq j \leq k\}|$ and $r = r(s)$. This form of α is called its *H-form*. While F_K is not, in general, partitionable, F_A is; the *H-form* of $\text{Rel}_A(G; p)$ is $\alpha(F_A; p, 1-p) = p^{n-1} \sum_{i=0}^{e-n+1} H_i (1-p)^i$. Similarly, as $\text{Br}(G; <)$ is partitionable, we get the *H-form* $\alpha(\text{Br}(G; <); \lambda, -1) = \lambda^{e-n+1} \sum H_i (-1)^i (\lambda - 1)^{n-1-i}$, so $\pi(G; \lambda) = (-1)^{n-1} \lambda \sum_{i=0}^{n-1} H_i (1-\lambda)^{n-1-i}$.

Let the standard or *S-form* of $\text{Rel}_K(G; p)$ and $\pi(G; \lambda)$ be its expansion into powers of p and λ , respectively, i.e., $\text{Rel}_K(G; p) = \sum_{i=n-1}^e S_i p^i$, $\pi(G; \lambda) = \sum_{i=1}^n S_i \lambda^i$. It is clear from both *H-forms* of $\text{Rel}_A(G; p)$ and $\pi(G; \lambda)$ that the signs of the coefficients in standard form alternate; the alternating sign theorem for $\pi(G; \lambda)$ is well known.

The *H-form* of the reliability polynomial has proved very useful in obtaining bounds on the coefficients of the polynomial [2]; the same technique bounds coefficients in the chromatic polynomial. We therefore develop a method for explicitly shelling the broken circuit complex to develop an interesting form of the chromatic polynomial. Following Biggs [3], an edge f of a tree T is *internally active with respect to T* if and only if it is the least edge under $<$ of the unique cutset of G containing f and no other edge of T (here, as always, $<$ is a fixed ordering of $E(G)$). T^i is defined as the set of internally active edges of T , and $\bar{T} = T - T^i$. Let $T_1, \dots, T_t$ be the spanning trees of G with no broken circuits. For any block X of $\text{Br}(G; <)$ of size $< n - 1$, let $\lambda(X)$ be the smallest edge in $G - X$ whose addition decreases the number of components of $X < G$. As Wilf [19] noted, $X \cup \lambda(X)$ is a block of $\text{Br}(G; <)$, so X^λ , the closure of X under λ , is a spanning tree with no broken circuits. Proposition 13.7 of [3] states that for any spanning tree T , with no broken circuits and any block X of $\text{Br}(G; <)$, $X^\lambda = T$ if and only if $\bar{T}_j \subseteq X \subseteq T_j$. It follows that $\{\{\bar{T}_j, T_j\}: 1 \leq j \leq t\}$ is a partition of $\text{Br}(G; <)$. Setting $H_k = |\{\bar{T}_j: |\bar{T}_j| = k\}|$, we have

$$\begin{aligned} \alpha(\text{Br}(G; <); \lambda, -1) &= \sum_{i=0}^{n-1} H_i (-1)^i \lambda^{e-n+1} (\lambda - 1)^{n-1-i} \\ \rightarrow \pi(G; \lambda) &= (-1)^{n-1} \lambda \sum_{i=0}^{n-1} H_i (1-\lambda)^{n-1-i} \\ \rightarrow \pi(G; \lambda) &= (-1)^{n-1} \lambda \sum_{i=0}^{n-1} h_i (1-\lambda)^i \end{aligned}$$

where $h_i = H_{n-1-i}$ is the number of spanning trees without broken circuits and with i internally active edges. This final form of the chromatic polynomial can be found in [3], but our proof is somewhat shorter and is clearly motivated by the underlying set structure. In addition, it provides a useful combinatorial interpretation of the coefficients of the *H-form* of the chromatic polynomial.

We now turn to a well-known conjecture (due to Read [13]) which states that the absolute values of the coefficients of the S -form of $\pi(G; \lambda)$ are unimodal, i.e., if $\pi(G; \lambda) = \sum \alpha_i \lambda^i$, then there is a J such that $|a_i| \leq |a_{i+1}|$ for $0 \leq i < J$, and $|a_i| \geq |a_{i+1}|$ for $i \geq J$. An examination of the all-terminal reliability of small graphs has led us to extend the conjecture to the S -form of $\text{Rel}_A(G; p)$.

CONJECTURE 2.3. *If $\text{Rel}_A(G; p) = \sum a_i p^i$, the sequence $|a_{n-1}|, \dots, |a_e|$ is unimodal.*

The coefficients $\{a_i\}$ have a combinatorial interpretation as the “ i -parities” of the graph; see [15]. While not much progress has been made on the chromatic unimodal conjecture, Heron [9] has proved a partial result, based on the H -form of the chromatic polynomial. This can be extended to give a partial result for Conjecture 2.3. First, we need a lemma used by Heron (see [3], [9]).

LEMMA 2.4. [3]. *If $f(p) = \sum_{i=0}^l f_i(1-p)^i$, $f_i \geq 0$ for $0 \leq i < l$, then the S -form of f , $\sum a_i p^i$, has $|a_{i+1}| \leq |a_i|$ for $(l-1)/2 \leq i < l$.*

Using this lemma, we extend the results to encompass reliability polynomials as well.

THEOREM 2.5. *Let the S -forms of $\pi(G; \lambda)$ and $\text{Rel}_A(G; p)$ be, respectively, $\sum_{i=1}^n a_i \lambda^i$ and $\sum_{i=n-1}^e b_i p^i$. Then (i) $|a_{i+1}| \leq |a_i|$ for $i \geq n/2$, (ii) $|b_{n+i}| \leq |b_{n+i-1}|$ for $(e-n)/2 \leq i < e-n$.*

Proof. $\pi(G; \lambda) = (-1)^{n-1} \lambda \sum_{i=0}^n h_i(1-\lambda)^i$ where $h_i \geq 0$. From Lemma 2.4, $|a_{i+2}| \geq |a_{i+1}|$ for $i \geq (n-2)/2$, which is equivalent to (i). For (ii), note that the H -form of $\text{Rel}_A(G; p) = p^{n-1} \sum_{i=0}^{e-n+1} H_i(1-p)^i$, so by Lemma 2.4, $|b_{n+i}| \leq |b_{n+i-1}|$ for $i \geq (e-n)/2$, and we are done. $\square$

We state one other related conjecture.

CONJECTURE 2.6. *If S is a partitionable complex and $\alpha(S; p, q) = \sum F_i p^i q^{v-i}$, then the sequence $F_0, \dots, F_v$ is unimodal.*

We note that for S a matroid, this is a well-known conjecture (see [18], for example).

3. Alternate forms of the α -polynomial. We have already seen the F -form and the H -form of the α -polynomial (the latter only for partitionable set systems). We now introduce three other forms for general set systems, in order to further study bounds on the coefficients and unimodality conjectures.

Given a set system S , we define three new set systems $C(S)$, $N(S)$, $M(S)$ on vertex set $V(S)$.

- (i) X is a block of $C(S)$ if and only if X is not a block of S ,
- (ii) X is a block of $N(S)$ if and only if $V(S) - X$ is a block of S ,
- (iii) X is a block of $M(S)$ if and only if $V(S) - X$ is not a block of S .

We have the following simple relationships.

LEMMA 3.1. *For any set system S , we have*

- (i) $\alpha(S; p, q) + \alpha(C(S); p, q) = (p + q)^v$,
- (ii) $\alpha(S; p, q) = \alpha(N(S); q, p)$,
- (iii) $\alpha(M(S); p, q) + \alpha(N(S); p, q) = (p + q)^v$,
- (iv) $N(N(S)) = S$; $M(M(S)) = S$; $C(C(S)) = S$,
- (v) $N(C(S)) = M(S)$; $M(C(S)) = N(S)$; $C(N(S)) = M(S)$;
 $M(N(S)) = C(S)$; $C(M(S)) = N(S)$; $N(M(S)) = C(S)$.
- (vi) *If S is a complex and $V(S)$ is not a block of S , $M(S)$ is a complex.*

In particular, if $I(S) = S$, then the set $\{I, C, N, M\}$ forms the Klein four-group under composition by Lemma 3.1 (iv), (v). In light of the above, we define three new forms of the α -polynomial.

DEFINITION 3.2. For any set system S , the C -, N -, and M -forms of $\alpha(S; p, q)$ are defined by the (unexpanded) expansions $(p + q)^v - \alpha(C(S); p, q)$, $\alpha(N(S); q, p)$, and $(p + q)^v - \alpha(N(S); q, p)$, respectively.

These three forms have apparently not been utilized in chromatic theory. However, they do yield new forms of $\pi(G; \lambda)$. For example, by considering the C -form of the chromatic polynomial, we get the following theorem.

THEOREM 3.3. *If G is any graph, $<$ any fixed ordering of G , and b_i is the number of i edge subgraphs of G containing a broken circuit, then*

$$\pi(G; \lambda) = \frac{(\lambda - 1)^e}{\lambda^{e-n}} - \sum_{i=0}^e (-1)^i b_i \lambda^{n-i}.$$

Proof. From Lemma 3.1 (i), we have

$$\begin{aligned} \alpha(\text{Br}(G; <); \lambda, -1) + \alpha(C(\text{Br}(G; <)); \lambda, -1) &= (\lambda - 1)^e \\ \rightarrow \lambda^{e-n} \pi(G; \lambda) + \sum_{i=0}^e (-1)^i b_i \lambda^{e-i} &= (\lambda - 1)^e \\ \rightarrow \pi(G; \lambda) &= \frac{(\lambda - 1)^e}{\lambda^{e-n}} - \sum_{i=0}^e (-1)^i b_i \lambda^{n-i}. \quad \square \end{aligned}$$

In the reliability case, the F -, C -, N -, and H -forms have been studied (see [7]) because the underlying structure of each has its own advantages in estimating reliability. In fact, one of the best techniques involves the Kruskal–Katona bounds [7], which apply to any complex. We have established that applying the Kruskal–Katona theorem to the M -form is equivalent to applying it to the F -form [5]. Hence no new bounds on reliability arise at present by considering the M -form.

We remark that Ball and Provan [2] have proposed a method for bounding the coefficients of the H -form of the α -polynomial for partitionable systems in order to get better bounds on all-terminal reliability. As $\text{Br}(G; <)$ is also partitionable, their results can be applied to the H -form of $\pi(G; \lambda)$ as well.

In addition, Conjecture 2.6 can be extended to the C -, N -, and M -forms of the α -polynomial as well. In fact, for matroids, a stronger property than unimodality has been considered. A sequence $a_1, \dots, a_k$ of numbers is *log concave* if and only if $a_i^2 \geq a_{i-1} a_{i+1}$ for $1 < i < k$. It is easy to see that any log concave sequence of nonnegative numbers is unimodal, but the converse need not hold (e.g., 5, 3, 2). If M is a matroid with M_i sets of size i , then a well-known conjecture (see [18], for example) states that the sequence (M_i) is log concave. In light of this, we propose:

CONJECTURE 3.4. *If S is a partitionable complex, and $\alpha(S; p, q)$ has F -, C -, N -, and M -forms $\sum F_i q^i p^{v-i}$, $(p + q)^v - \sum C_i q^i p^{v-i}$, $\sum N_i p^i q^{v-i}$, and $(p + q)^v - \sum M_i p^i q^{v-i}$, respectively, then the sequences (F_i) , (C_i) , (N_i) , and (M_i) are all log concave.*

For $S = \text{Br}(G; <)$, the conjecture for the F -form is essentially that of Read [13].

We now show that if Conjecture 3.4 holds for the F_i 's, it holds for the N_i 's and C_i 's as well.

THEOREM 3.5. *If S is a complex, then log concavity for the F -form implies log concavity for the C - and N -forms as well.*

Proof. If (F_i) is log concave, then as $(N_i) = (F_{v-i})$, the sequence (N_i) is log concave. We must show (F_i) is log concave which implies that (C_i) is log concave. Note that $F_i = \binom{v}{i} - C_i$ for $0 \leq i \leq v$. We may assume $V(S) \notin B(S)$, as otherwise $C_i = 0$ for all i .

We need an old result due to Sperner [17] which states that if S is a complex, $(v - i + 1)F_{i-1} \geq iF_i$ for $0 < i \leq v$. By a trivial computation, this implies

$$(v - i + 1)C_{i-1} \leq iC_i$$

for $0 < i \leq v$. Clearly, $(i+1)/(v-i)C_i \geq i/(v-i+1)C_i$, and we get from this and Sperner's lemma that

$$(1) \quad \begin{aligned} \frac{i+1}{v-i} C_i &\geq C_{i-1} \\ \rightarrow C_i &\geq \frac{v-i}{i+1} C_{i-1}. \end{aligned}$$

Similarly, as $i/(v-i+1)F_{i+1} \leq (i+1)/(v-i)F_{i+1}$, by Sperner's lemma we get $i/(v-i+1)F_{i+1} \leq F_i$

$$(2) \quad \rightarrow -F_i \geq -\frac{i}{v-i+1} F_{i+1}.$$

Now assume that the sequence (F_i) is log concave. Then from (1) and (2) we have

$$\binom{v}{i} - 2F_i = C_i - F_i \geq \frac{v-i}{i+1} C_{i-1} - \frac{i}{v-i+1} F_{i+1}.$$

Using some algebra, we obtain the desired result: $C_i^2 \geq C_{i+1}C_{i-1}$, that is, (C_i) is log concave. $\square$

4. Concluding remarks. We conclude by noting that the α -polynomial provides a convenient and simple setting in which to consider the chromatic and reliability polynomials. Many of the known results on one of the latter two polynomials can be generalized to the α -polynomial and hence provide new theorems and proofs for chromatic and reliability polynomials. Also, our α -variable polynomial focuses attention on the underlying set systems, whose study we feel is essential to both chromatic and reliability theory.

In this paper, we have concentrated on reliability and colouring, since these are both widely studied problems. However, it is important to note that the results apply in other domains of interest. For example, the α -polynomial of a graphic matroid expresses the probability of obtaining an acyclic subgraph of a graph. In this setting, the F -form counts acyclic subgraphs, while the N -form counts "feedback edge sets" in the graph. The methods developed here for bounding coefficients and for studying unimodality conjectures apply equally well in this case.

Acknowledgment. The authors would like to thank Derek Corneil for his insightful remarks.

REFERENCES

- [1] M. O. BALL AND G. L. NEMHAUSER, *Matroids and reliability analysis problem*, Math. Oper. Res., 4 (1979), pp. 132-143.
- [2] M. O. BALL AND J. S. PROVAN, *Bounds on the reliability polynomial for shellable independence systems*, SIAM J. Algebraic Discrete Methods, 3 (1982), pp. 166-181.
- [3] N. BIGGS, *Algebraic Graph Theory*, Cambridge University Press, London, 1974.
- [4] J. A. BONDY AND U. S. R. MURTY, *Graph Theory with Applications*, North-Holland, Amsterdam, 1976.
- [5] J. I. BROWN AND C. J. COLBOURN, *A combinatorial study of the reliability polynomial*, Proc. Sixteenth Manitoba Conference Numerical Math. and Computing, Winnipeg, 1986, pp. 71-89.
- [6] T. BRYLAWSKI, *The broken-circuit complex*, Trans. Amer. Math. Soc., 234 (1977), pp. 417-433.
- [7] C. J. COLBOURN, *The Combinatorics of Network Reliability*, Oxford University Press, Oxford, 1987.
- [8] C. J. COLBOURN AND W. R. PULLEYBLANK, *Matroid Steiner problems, the Tutte polynomial and network reliability*, J. Combin. Theory Ser. B, to appear.

- [9] A. HERON, *Matroid polynomials*, in Combinatorics, Proc. Conf. Comb. Math. Inst., Oxford, 1972, pp. 164–202.
- [10] G. KATONA, *A theorem of finite sets*, in Theory of Graphs, P. Erdős and G. Katona, eds., Akademia Kiado, Budapest, 1966, pp. 187–207.
- [11] J. B. KRUSKAL, *The number of simplices in a complex*, in Mathematical Optimization Techniques, R. Bellman, ed., University of California Press, Berkeley, 1963, pp. 251–278.
- [12] J. S. PROVAN AND L. J. BILLERA, *Decompositions of simplicial complexes related to diameters of convex polyhedra*, Math. Oper. Res., 5 (1980), pp. 579–594.
- [13] R. C. READ, *An introduction to chromatic polynomials*, J. Combin. Theory, 4 (1968), pp. 52–71.
- [14] A. SATYANARAYANA AND M. K. CHANG, *Network reliability and the factoring theorem*, Networks, 13 (1983), pp. 107–120.
- [15] A. SATYANARAYANA AND Z. KHALIL, *On an invariant of graphs and the reliability polynomial*, SIAM J. Algebraic Discrete Methods, 7 (1986), pp. 399–403.
- [16] R. M. VAN SLYKE AND H. FRANK, *Network reliability analysis I*, Networks, 1 (1972), pp. 279–290.
- [17] E. SPERNER, *Ein Satz über Untermengen einer endlichen Menge*, Math. Z., 27 (1928), pp. 544–548.
- [18] D. J. A. WELSH, *Matroid Theory*, Academic Press, New York, 1979.
- [19] H. S. WILF, *Which polynomials are chromatic?*, Colloquio Internazionale sulle Teorie Combinatorie, (1976), pp. 247–256.

WIDE-SENSE NONBLOCKING NETWORKS*

PAUL FELDMAN†, JOEL FRIEDMAN‡, AND NICHOLAS PIPPENGER§

Abstract. A new method for constructing wide-sense nonblocking networks is presented. Application of this method yields (among other things) wide-sense nonblocking generalized connectors with n inputs and outputs and size $O(n \log n)$, and with depth k and size $O(n^{1+1/k}(\log n)^{1-1/k})$.

Key words. nonblocking network, connection network, concentrator, generalizer

AMS(MOS) subject classifications. 94C15, 68E10, 05C35

1. Introduction. In this introductory section, we shall give an informal account of our results. More precise definitions will be given in later sections.

Informally, a “network” is an acyclic directed graph that allows connections to be made between “inputs” and “outputs.” A “nonblocking” network is one that allows connections to be created and destroyed dynamically, with each request for creation being satisfied without disturbing other previously established connections.

Given a network, we may consider a game between two players, the “blocker” (representing the collective activity of the users interconnected by the network) and the “router” (representing the activity of the controller of the network). The positions in this game are the states of the network, and the initial position is the state in which no connections are established. The players move alternately, starting with the blocker. The blocker moves by making a request to create or destroy a connection, and the router responds by satisfying this request (if possible). The goal of the blocker is to make a request that cannot be satisfied, and the goal of the router is to avoid this by satisfying every request. Roughly speaking, a “wide-sense” nonblocking network is one in which the router can indefinitely avoid losing. (In addition to wide-sense nonblocking networks, there are “strictly” nonblocking networks and “rearrangeable” networks. We do not need these notions to formulate our results, but we shall occasionally need them to discuss other results in the literature.)

The main contribution of this paper is a new principle, according to which certain “expanding graphs” are shown to be “generalized concentrators” (a special case of which are “concentrators”). For these networks, inputs request connection to one or more outputs, without specifying to which output they are to be connected. This principle will be presented in § 2.

In § 3, we apply the results of § 2 to construct “generalized connectors” (a special case of which are “connectors”). For these networks, inputs request connection to one or more specified outputs. Our results in this section, taken together with results of Friedman [F], show that wide-sense nonblocking connectors can have smaller size than strictly nonblocking connectors with the same number of inputs and outputs and the same depth. This answers a question that has been open since the distinction between “wide-sense” and “strictly” nonblocking networks was introduced by Beneš in 1962 [B1] (see also [B2], [B3]).

In § 4, we apply the results of § 2 to construct “bi-generalized connectors.” This new type of network provides greater interconnection facility than most other types

* Received by the editors June 17, 1987; accepted for publication (in revised form) November 6, 1987.

† Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

‡ Princeton University, Princeton, New Jersey 08544.

§ IBM Almaden Research Center, San Jose, California 95120-6099.

considered in the literature, and it serves as a natural setting to display the power of the new principle.

The results of §§ 3 and 4 concern networks with “limited depth,” in which the length of the routes from inputs to outputs is constrained. In § 5 we apply the results of § 2 to construct networks of “unlimited depth,” in which only the “size” (corresponding to the number of switches in the network) is to be minimized. Our results in this section, taken together with results of Shannon [S], show that wide-sense nonblocking generalized connectors need be larger than wide-sense nonblocking connectors by only a constant factor. The corresponding result for “rearrangeable” rather than “wide-sense nonblocking” networks has been known since the work of Ofman [O] in 1965. The corresponding result for “strictly nonblocking” rather than “wide-sense nonblocking” networks does not hold (see Bassalygo and Pinsker [BP2]).

A preliminary account of some of these results appears in our earlier paper [FFP].

2. Generalized concentrators. In this section we shall define networks precisely and present the principle on which most of the results of this paper are based. We shall base our formulation of a “network” on a directed graph with “inputs” and “outputs.” There is an alternative formulation based on an undirected graph with “terminals.” Our choice for this paper is technically simpler, though our results hold (with obvious modifications) for either formulation.

A *network* is an acyclic directed graph with a set of distinguished vertices called *inputs* and a disjoint set of distinguished vertices called *outputs*. A vertex that is neither an input nor an output will be called a *link*. We shall use the prefix “ (n, m) –” to the term “network” (and later to other nouns) to indicate that there are n inputs and m outputs; we shall use the prefix “ n –” in these cases in place of “ (n, n) –.”

A *route* in a network is a directed path from an input to an output. The *size* of a network is the number of edges in it. The *depth* of a network is the maximum number of edges in a route in it.

Two routes are *compatible* if their intersection is an initial segment (possibly empty) of both of them. A *state* of a network is a set of pairwise compatible routes. (A state may be regarded as a forest of vertex-disjoint trees, with their roots at the inputs and their leaves at the outputs. This alternative viewpoint is taken in many papers on networks with “generalized” properties.) The states of a network are partially ordered by inclusion, and we shall use the terms *above* and *below* to refer to this partial order. A vertex or edge is *busy* in a state if it appears in some route of that state; otherwise, it is *idle*.

A *concentration request* in a network is an input of the network. A concentration request is *satisfied* by a route if the route originates at the input of the request. A *generalized concentration assignment* is a multiset of concentration requests. A generalized concentration assignment is *realized* by a state if each request is satisfied in the state by a number of routes equal to its multiplicity.

A generalized concentration assignment is (a, r) -*limited* if each request has multiplicity at most r and the sum of the multiplicities is at most a . A state is (a, r) -*limited* if the generalized concentration assignment it realizes is (a, r) -*limited*. A concentration request is (a, r) -*limited* in a state if the generalized concentration assignment obtained by adjoining the request to the generalized concentration assignment realized by the state is (a, r) -*limited*.

A *wide-sense nonblocking (a, r) -limited generalized concentrator* is a network for which there exists a set of distinguished states, called the *safe* states, with the following properties: (1) the empty state is safe; (2) any state below a safe state is itself safe; and (3) given any safe state and any (a, r) -limited concentration request in that state, there

exists a safe state above the given state containing another route satisfying the given request. (In our previous paper [FFP], “generalized concentrators” were called “generalizers.” This change has been adopted to increase the uniformity of our terminology.)

A *wide-sense nonblocking a -limited concentrator* is a wide-sense nonblocking $(a, 1)$ -limited generalized concentrator. When discussing concentrators, we shall use the adjective “ a -limited” in place of “ $(a, 1)$ -limited.”

The key to the results of this paper is the following proposition.

PROPOSITION 1. *If F is a network of depth one in which every set of x inputs, where $1 \leq x \leq 2a$, has at least $(r + s)x$ neighbors, then F is a wide-sense nonblocking (as, r) -limited generalized concentrator.*

Proof. Let $F = (V, W, E)$ be a network of depth one, with inputs V , outputs W , and edges E . Suppose that if $X \subseteq V$, with $|X| = x$ and $1 \leq x \leq 2a$, then $|E(X)| \geq (r + s)x$. (Here $E(X)$ denotes the set of neighbors of X , and $|Y|$ denotes the cardinality of Y .) We shall show that F is a wide-sense nonblocking (as, r) -limited generalized concentrator.

The first step of the proof is the definition of the set of safe states; for this we shall need some auxiliary definitions. Let X be a set of inputs. We shall define the *assets* of X in S , $A_S(X)$, to be the number of idle neighbors of X in S . Let v be an input in V . We shall define the *liabilities* of v in S , $B_S(v)$, to be r minus the number of busy edges directed out of v in S . We shall define the *liabilities* of X in S , $B_S(X)$, to be the sum of $B_S(v)$ over inputs v in X . We define the *balance* of X in S , $C_S(X)$, to be $A_S(X) - B_S(X)$. We shall say that X is *solvent* in S if $C_S(X) \geq 0$, that X is *critical* in S if $C_S(X) = 0$, and that X is *bankrupt* in S if $C_S(X) < 0$. Finally, we shall say that S is *safe* if, for every $X \subseteq V$ with $|X| = x$ and $1 \leq x \leq 2a$, X is solvent in S .

Let O denote the empty state. We shall show that O is safe. Let X be a set of inputs with $|X| = x$ and $1 \leq x \leq 2a$. By the hypothesis on F , $|E(X)| \geq (r + s)x \geq rx$. Since all vertices are idle in O , $A_O(X) \geq rx$. Since $B_O(X) = rx$, X is solvent, and thus O is safe.

Let S be a state, let (v, w) be a route not in S , and suppose that $T = S \cup \{(v, w)\}$ is a safe state. We shall show that S is safe. We observe that (1) $A_T(X) = A_S(X) - 1$ if $w \in E(X)$, otherwise $A_T(X) = A_S(X)$; and (2) $B_T(X) = B_S(X) + 1$ if $v \in X$, otherwise $B_T(X) = B_S(X)$. Since $v \in X$ implies $w \in E(X)$, $C_S(X) \geq C_T(X) \geq 0$, and thus S is safe.

Finally, let S be a safe state and let v be an (as, r) -limited concentration request in S . We shall show that for some idle neighbor w of v in S , the state $S \cup \{(v, w)\}$ is safe. Suppose, to obtain a contradiction, that for every idle neighbor w of v , the state $S \cup \{(v, w)\}$ is unsafe. Then for every idle neighbor w of v , there is a set of inputs X_w with $|X_w| \leq 2a$ such that X_w is bankrupt in $S \cup \{(v, w)\}$. Of course, since S is safe, for every w , X_w is solvent in S . From these conditions, and from observations (1) and (2) above, it is easy to see that for every idle neighbor w of v we have (i) w belongs to $E(X_w)$, but v does not belong to X_w , and (ii) X_w is critical in S . We shall need the following lemmas concerning critical sets.

LEMMA 1.1. *If X is critical in S , then $|X| < a$.*

Proof. Suppose, to obtain a contradiction, that X with $|X| = x$ and $a \leq x \leq 2a$ is unsafe in S . By the hypothesis on F , $|E(X)| \geq (r + s)x$. Since the request v is (as, r) -limited in S , fewer than sa outputs are busy in S , so $A_S(X) > (r + s)x - sa$. Since $B_S(X) \leq B_O(X) = rx$ and $x \geq a$, we have $C_S(X) > 0$, contradicting the assumption that X is critical in S . $\square$

LEMMA 1.2. *For any state S , $C_S(X)$ is submodular in X ; that is, $C_S(X \cup Y) + C_S(X \cap Y) \leq C_S(X) + C_S(Y)$.*

Proof. Since $B_S(X)$ is defined by a sum over the inputs in X , it is *modular* in X ; that is, $B_S(X \cup Y) + B_S(X \cap Y) \leq B_S(X) + B_S(Y)$. Thus it remains to show that $A_S(X)$

is submodular in X . This follows from the relations $E(X \cup Y) = E(X) \cup E(Y)$ and $E(X \cap Y) \subseteq E(X) \cap E(Y)$, and the identity $|X \cup Y| + |X \cap Y| = |X| + |Y|$. $\square$

LEMMA 1.3. *If S is a safe state, $|X|, |Y| \leq 2a$, and X and Y are critical in S , then $|X \cup Y| \leq 2a$ and $X \cup Y$ is critical in S .*

Proof. Since X and Y are critical for S , Lemma 1.1 implies $|X|, |Y| < a$, and thus $|X \cup Y| \leq 2a$. Thus it remains to show that $X \cup Y$ is critical in S . Since X and Y are critical, $C_S(X) + C_S(Y) = 0$. By Lemma 1.2, $C_S(X \cup Y) + C_S(X \cap Y) \leq 0$. Since S is safe and $|X \cap Y| < a$, $X \cap Y$ is solvent. Thus $X \cup Y$ is critical. $\square$

Returning to the proof of Proposition 1, let the set of inputs X be the union of X_w over all idle neighbors w of v . Since each X_w is critical in S , Lemma 1.3 implies that X is critical in S , and Lemma 1.1 implies $|X| < a$. Now consider the set $X \cup \{v\}$. Since S is safe and $|X \cup \{v\}| \leq a$, $X \cup \{v\}$ is solvent in S . On the other hand, (i) and (ii) above imply that every idle neighbor of v belongs to $E(X)$, but v does not belong to X . Thus $A_S(X \cup \{v\}) = A_S(X)$ and, since the request v is (a, r) -limited in S ,

$$B_S(X \cup \{v\}) = B_S(X) + 1.$$

Thus, since X is critical in S , $X \cup \{v\}$ is bankrupt in S . This contradiction completes the proof of Proposition 1. $\square$

The argument used in the proof of Proposition 1 has also found application to some combinatorial problems concerning fault-tolerant networks (see Friedman and Pippenger [FP]) and processor-interconnection networks (see Peleg and Upfal [PU]). This argument suffers the defect that it provides no efficient algorithm for finding safe routes. Indeed, determining the solvency of a state is a minor variant of a co-NP-complete problem (see Blum et al. [BKVPY]). We shall present some results (Theorems 4 and 6) that avoid this defect; it is avoided in another way in the work of Peleg and Upfal [PU].

A wide-sense nonblocking (a, r) -limited generalized concentrator remains one if any number of inputs (together with the edges incident with them) are deleted and any number of outputs (with no incident edges) are added.

If F is an (n, m) -network and G is an (m, l) -network, let $F \circ G$ denote a network obtained by identifying the outputs of F with the inputs of G in any one-to-one fashion. If F is an n -network, define F^{*k} by letting $F^{*1} = F$ and $F^{*k} = F \circ F^{*(k-1)}$ for $k \geq 2$.

PROPOSITION 2. *If F is a wide-sense nonblocking (a, r) -limited generalized m -concentrator and G is a wide-sense nonblocking (a, s) -limited generalized m -concentrator, then $F \circ G$ is a wide-sense nonblocking (a, rs) -limited generalized m -concentrator.*

Proof. Let the safe states of $F \circ G$ be those in which the induced states of F and G are safe. Clearly, the empty state is safe and any state below a safe state is itself safe. Let S be a safe state and let v be an (a, rs) -limited concentration request in S . Let $S(F)$ be the state of F induced by S and let $S(G)$ be the state of G induced by S .

Since v is (a, rs) -limited, S contains at most $a - 1$ routes, at most $rs - 1$ of which are from v . Thus either (1) there are routes in $S(F)$ from v to at most $r - 1$ outputs of F , or (2) there is a route in $S(F)$ from v to some output w of F and there are routes in $S(G)$ from w to at most $s - 1$ outputs of G .

In case (1), the concentration request v is (a, r) -limited in $S(F)$, so there is a safe state $T(F)$ above $S(F)$ containing a route from v to some output w of F that is idle in $S(F)$. The concentration request w is then (a, s) -limited in $S(G)$, so there is a safe state $T(G)$ above $S(G)$ containing a route from w to some output of G that is idle in $S(G)$. Then the state obtained by adjoining to S the concatenation of the route from v to w in $T(F)$ with the route from w in $T(G)$ is a safe state above S containing a route from v to an output of $F \circ G$ that is idle in S .

In case (2), the concentration request w is (a, s) -limited in $S(G)$, so there is a safe state $T(G)$ above $S(G)$ containing a route from w to an output of G that is idle in $S(G)$. Then the state obtained by adjoining to S the concatenation of the route from v to w in $S(F)$ with the route from w in $T(G)$ is a safe state above S containing a route from v to an output of $F \circ G$ that is idle in S . $\square$

3. Generalized connectors. In this section we shall use Proposition 1 to construct wide-sense nonblocking generalized connectors. Our first goal is to define these networks.

A *connection request* in a network is an ordered pair comprising an input and an output. A connection request is *satisfied* by a route if the route originates at the input of the request and terminates at the output of the request. A *generalized connection assignment* is a set of connection requests, no two of which have an output in common. A generalized connection assignment is *realized* by a state if each request of the generalized connection assignment is satisfied by some route of the state.

A generalized connection assignment is (a, r) -*limited* if it contains at most a requests, of which at most r have an input in common. Observe that a state is (a, r) -limited if the generalized connection assignment that it realizes is (a, r) -limited. A connection request is (a, r) -*limited* in a state if (1) the output of the request is idle in the state and (2) the generalized connection assignment obtained by adjoining the connection request to the generalized connection assignment realized by the state is (a, r) -limited.

A *wide-sense nonblocking (a, r) -limited generalized connector* is a network for which there exists a set of distinguished states, called the *safe states*, with the following properties: (1) the empty state is safe; (2) any state below a safe state is itself safe; and (3) given any safe state and any (a, r) -limited connection request in that state, there exists a safe state above the given state containing a route satisfying the given request.

A *wide-sense nonblocking a -limited connector* is a wide-sense nonblocking $(a, 1)$ -limited generalized connector. When discussing (ungeneralized) connectors, we shall use the term “connection assignment” in place of “generalized connection assignment” and the adjective “ a -limited” in place of “ $(a, 1)$ -limited.” We shall also use the term “connector” in place of “ ∞ -limited connector.”

Our main result in this section is the following.

THEOREM 3. *For every fixed $k \geq 1$ and $n \rightarrow \infty$, a wide-sense nonblocking generalized n -connector exists with depth k and size $O(n^{1+1/k}(\log n)^{1-1/k})$.*

The bound of Theorem 3 is almost the best possible; Pippenger and Yao [PY] have shown that size $\Omega(n^{1+1/k})$ is necessary (even for networks satisfying a much weaker property, that of being “rearrangeable n -shifters” with depth k).

The previously best upper bound for wide-sense nonblocking generalized connectors was obtained from a construction due to Pippenger [P1] and Nassimi and Sahni [NS]. It yields wide-sense nonblocking generalized connectors with depth $j^2 - 3j + 3$ and size $O(n^{1+2/j})$. The special case of depth three and size $O(n^{5/3})$ was given (using a different method) by Masson and Jordan [MJ].

The construction of Pippenger [P1] and Nassimi and Sahni [NS] requires wide-sense nonblocking (ungeneralized) connectors as components. The previously best upper bound for these networks was obtained from constructions due to Clos [Cl], Cantor [Ca], and Pippenger [P2]. It yields networks of depth $2j - 1$ and size $O(n^{1+1/j})$. (These networks satisfy a stronger property, that of being “strictly nonblocking n -connectors” with depth $2j - 1$.)

The best upper bound for wide-sense nonblocking connectors is now given by Theorem 3. The best upper bound for “rearrangeable generalized connectors” is also now given by Theorem 3 (the previously best bound was $O(n \log n)^{1+1/k}$, due to Dolev et

al. [DDPW]). Indeed, the bound of Theorem 3 is not much larger than the currently best bound of $O(n^{1+1/k}(\log n)^{1/k})$ for “rearrangeable connectors,” due to Pippenger and Yao [PY].

The lower bound of $\Omega(n^2)$ (for any depth) due to Bassalygo and Pinsker [BP2] shows that Theorem 3 cannot be extended to “strictly nonblocking generalized connectors.” The lower bound of $\Omega(n^{1+1/(k-1)})$ (for depth k) due to Friedman [F] shows that it cannot even be extended to “strictly nonblocking connectors.”

We need the following lemma for the proof of Theorem 3.

LEMMA 3.1. *For every n and a , there exists a network of depth one with n inputs and $8a$ outputs in which (1) every input has $d = \lceil \log_2(2n^3) \rceil$ neighbors, and (2) every set of x inputs with $1 \leq x \leq 2a$ has at least $2x$ neighbors.*

Proof. Consider a random graph in which each input has a randomly chosen set of d outputs as neighbors, with all sets of d outputs being equally likely and with the random choices for distinct inputs being independent. Clearly (1) is satisfied; it remains to show that the probability that (2) is satisfied is strictly positive. The probability that (2) fails is at most

$$\sum_{1 \leq x \leq 2a} \binom{n}{x} \binom{8a}{2x} \left(\binom{2x}{d} / \binom{8a}{d} \right)^x.$$

Using the estimates

$$\binom{n}{x} \binom{8a}{2x} \leq n^{3x}$$

and

$$\binom{2x}{d} / \binom{8a}{d} \leq \left(\frac{2x}{8a} \right)^d \leq \left(\frac{1}{2} \right)^d,$$

we see that this probability is at most $\sum_{1 \leq x \leq 2a} (1/2)^x < 1$. $\square$

A wide-sense nonblocking (a, r) -limited generalized connector remains one if any number of inputs and outputs (together with the edges incident with them) are deleted.

The following propositions and their subsequent use are the counterparts for wide-sense nonblocking generalized connectors of a method used by Pippenger [P1] for “strictly nonblocking generalized connectors” and by Nassimi and Sahni [NS] for “rearrangeable generalized connectors.”

PROPOSITION 3.2. *If F is a wide-sense nonblocking a -limited (n, m) -concentrator and G is a wide-sense nonblocking (a, r) -limited generalized (m, l) -connector, then $F \circ G$ is a wide-sense nonblocking (a, r) -limited generalized (n, l) -connector.*

Proof. Let the safe states of $F \circ G$ be those in which the induced states of F and G are safe. Clearly, the empty state is safe and any state below a safe state is itself safe. Let S be a safe state and let (v, w) be an (a, r) -limited connection request in S . Let $S(F)$ be the state of F induced by S and let $S(G)$ be the state of G induced by S . Either (1) v is idle in S , or (2) S contains a route originating at v .

In case (1), the concentration request v is a -limited in $S(F)$, so there is a safe state $T(F)$ above $S(F)$ containing a route from v to an output u of F that is idle in $S(F)$. The connection request (u, w) is (a, r) -limited in $S(G)$, so there is a safe state $T(G)$ above $S(G)$ containing a route from u to w . Then the state obtained by adjoining to S the concatenation of the route from v to u in $T(F)$ with the route from u to w in $T(G)$ is a safe state above S containing a route from v to w .

In case (2), the route originating at v passes through some link u that is an input of G . The connection request (u, w) is (a, r) -limited in $S(G)$, so there is a safe state $T(G)$ above $S(G)$ containing a route from u to w . Then the state obtained by adjoining to S the concatenation of the route from v to u in $S(F)$ with the route from u to w in $T(G)$ is a safe state above S containing a route that from v to w . $\square$

If F is an (n, m) -network and G is an (n, l) -network, let $F \vdash G$ denote an $(n, m + l)$ -network obtained by identifying the inputs of F with the inputs of G in any one-to-one fashion. If F is any network, define F^{+t} by letting $F^{+1} = F$ and $F^{+t} = F \vdash F^{+(t-1)}$ for $t \geq 2$.

PROPOSITION 3.3. *If F is a wide-sense nonblocking (a, r) -limited generalized (n, m) -connector and G is a wide-sense nonblocking (a, r) -limited generalized (n, l) -connector, then $F \vdash G$ is a wide-sense nonblocking (a, r) -limited generalized $(n, m + l)$ -connector.*

Proof. Let the safe states of $F \vdash G$ be those that induce safe states of F and G . Clearly, the empty state is safe and any state below a safe state is itself safe. Let S be a safe state and let the connection request (v, w) be (a, r) -limited in S . Without loss of generality, we may suppose that w is an output of F .

Let $S(F)$ be the state of F induced by S . The connection request (v, w) is (a, r) -limited in $S(F)$, so there is a safe state $T(F)$ above $S(F)$ containing a route from v to w . Then the state obtained by adjoining this route to S is a safe state above S containing a route from v to w . $\square$

Proof of Theorem 3. We proceed by induction on k . For $k = 1$, the conclusion is trivial. For $k \geq 2$, take $a = \lceil n^{1-1/k}(\log n)^{1/k} \rceil$ in Lemma 3.1 and $r = s = 1$ in Proposition 1 to obtain a wide-sense nonblocking concentrator F with n inputs, $8a$ outputs, depth one, and size $O(n \log n)$. By inductive hypothesis, there exists a wide-sense nonblocking generalized $(8a)$ -connector G with depth $k - 1$ and size

$$O(a^{1+1/(k-1)}(\log a)^{1-1/(k-1)}) = O(n \log n).$$

By Proposition 3.2, $H = F \circ G$ is a wide-sense nonblocking generalized $(n, 8a)$ -connector with depth k and size $O(n \log n)$. Let $t = \lceil n/8a \rceil = O(n^{1/k}/(\log n)^{1/k})$. By Proposition 3.3, H^{+t} is a wide-sense nonblocking generalized $(n, 8at)$ -connector with depth k and size $O(n^{1+1/k}(\log n)^{1-1/k})$. Deleting all but n outputs of this network completes the proof. $\square$

The networks of Theorem 3 are obtained by the nonconstructive Lemma 3.1; moreover, since they rely on Proposition 1, no efficient algorithm is known for finding safe routes. The following remaining results of this section mitigate these defects in some special cases.

THEOREM 4. *Wide-sense nonblocking generalized n -connectors can be explicitly constructed with depth two and size $O(n^{5/3})$, and with depth three and size $O(n^{11/7})$.*

For the proof of Theorem 4 we shall need the following lemma.

LEMMA 4.1. *For every prime power q , a (q^3, q^2) -network with depth one can be explicitly constructed in which every input has q neighbors and every pair of distinct inputs has at most two common neighbors.*

Proof. Let $\text{GF}(q)$ be the field with q elements. Let the inputs be the elements of $\text{GF}(q)^3$ and let the outputs be the elements of $\text{GF}(q)^2$. Let the input (a, b, c) have the output (x, y) as a neighbor if and only if $ax^2 + bx + c = y$ (where all arithmetic is in $\text{GF}(q)$).

Every input has q neighbors, one for each possible value of x . If (x, y) is a common neighbor of (a, b, c) and (a', b', c') , then $ax^2 + bx + c = y$ and $a'x^2 + b'x + c' = y$. Subtracting these equations, we have $(a - a')x^2 + (b - b')x + (c - c') = 0$. If (a, b, c) is distinct from (a', b', c') , the coefficients of this quadratic equation do not all vanish,

and thus it can be satisfied by at most two values of x . Thus this pair of inputs can have at most two common neighbors, one for each of these values of x . $\square$

Let $F(q)$ denote the network constructed in the proof of Lemma 4.1.

PROPOSITION 4.2. *The network $F(5^\lambda)$ is a wide-sense nonblocking $(2 \cdot 5^{2\lambda-2})$ -limited concentrator.*

Proof. Let the safe states be those in which every input has at most $3 \cdot 5^{\lambda-1}$ neighbors that terminate routes originating at other inputs. Clearly, the empty state is safe and any state below a safe state is itself safe. Consider a safe state S in which some concentration request v cannot be satisfied. We shall show that at least $2 \cdot 5^{2\lambda-2}$ outputs are busy in S .

Since v is idle in S , v has at most $3 \cdot 5^{\lambda-1}$ busy neighbors in S . Thus it has at least $2 \cdot 5^{\lambda-1}$ idle neighbors in S . Say that an output w is *critical* if w is a neighbor of v , w is idle in S , and $S \cup \{(v, w)\}$ is not safe. Then there are at least $2 \cdot 5^{\lambda-1}$ critical outputs.

Say that an input u is *critical* if u is distinct from v and u has $3 \cdot 5^{\lambda-1}$ neighbors that terminate routes that originate at inputs other than u . Each critical output w must be the neighbor of some critical input, else $S \cup \{(v, w)\}$ would be safe. Each critical input u can have at most two critical outputs as neighbors, else the distinct inputs v and u would have more than two common neighbors. Thus there must be at least $5^{\lambda-1}$ critical inputs.

Let X be a set of $5^{\lambda-1}$ critical inputs. Each critical input in X has at least $3 \cdot 5^{\lambda-1}$ busy neighbors. Furthermore, each of the $\binom{5^{\lambda-1}}{2}$ pairs of distinct critical inputs in X has at most two busy neighbors in common. Thus there are at least

$$5^{\lambda-1}(3 \cdot 5^{\lambda-1}) - \binom{5^{\lambda-1}}{2} \cdot 2 \geq 2 \cdot 5^{2\lambda-2}$$

busy outputs in S . $\square$

Proof of Theorem 4. To prove the first assertion, we may assume that $n = 5^{3\lambda}$ for some integer $\lambda \geq 1$. Let $q = 5^\lambda$ and $a = 2 \cdot 5^{2\lambda-2}$. Let G be a wide-sense nonblocking generalized (q^2, a) -connector with depth one and size $q^2a = O(n^{4/3})$ (in which each input is joined by an edge to each output). By Proposition 3.2, $H = F(q) \circ G$ is a wide-sense nonblocking generalized (n, a) -connector with depth two and size $O(n^{4/3})$. Let $t = \lceil n/a \rceil = O(n^{1/3})$. By Proposition 3.3, $H^{t,t}$ is a wide-sense nonblocking generalized (n, at) -connector with depth 2 and size $O(n^{5/3})$. Deleting all but n outputs completes the proof of the first assertion.

To prove the second assertion, we may assume that $n = 5^{7\mu}$ for some integer $\mu \geq 1$. Let $q = 5^{3\mu}$ and $a = 2 \cdot 5^{6\mu-2}$. By deleting all but n inputs from $F(q)$, we obtain a wide-sense nonblocking a -limited (n, q^2) -concentrator F' with depth one and size $O(n^{10/7})$. Let G' be a wide-sense nonblocking generalized (q^2, a) -connector with depth two and size $O(n^{10/7})$ constructed by taking $\lambda = 2\mu$ in the proof of the first assertion and deleting all but a outputs. By Proposition 3.2, $H = F' \circ G'$ is a wide-sense nonblocking generalized (n, a) -connector with depth three and size $O(n^{10/7})$. Let $t = \lceil n/a \rceil = O(n^{1/7})$. By Proposition 3.3, $H^{t,t}$ is a wide-sense nonblocking generalized (n, at) -connector with depth three and size $O(n^{11/7})$. Deleting all but n outputs completes the proof of the second assertion. $\square$

We could extend Theorem 4 to yield size $O(n^{23/25})$ for depth four, and so forth. As $k \rightarrow \infty$, the sequence of exponents approaches, but remains greater than, $3/2$. In the following section, however, we shall obtain size $O(n^{3/2})$ for depth four, so this extension is of little interest.

4. Bi-generalized connectors. Our efforts to replace the probabilistic argument of the preceding section by an explicit construction lead naturally to consideration of the

best currently known explicit constructions for expanding graphs, due to Lubotzky, Phillips, and Sarnak [LPS] and Margulis [M2]. The networks produced with the aid of these constructions have greater size than those produced in the preceding section; they possess, however, a stronger property than that of being generalized connectors. We shall call them “bi-generalized connectors.” Our next goal is to define these networks.

Two routes are *bi-compatible* if their intersection is an initial segment or a final segment (possibly empty) of both of them. A *bi-state* is a set of pairwise bi-compatible routes. (A bi-state may be regarded as two forests with their leaves identified, one with its roots at the inputs and the other with its roots at the outputs.)

A *bi-generalized connection assignment* is a set of connection requests. A bi-generalized connection assignment is *realized* by a bi-state if each request of the assignment is satisfied by some route of the bi-state.

A bi-generalized connection assignment is *a-limited* if it contains at most a connection requests. A bi-state is *a-limited* if the bi-generalized connection assignment it realizes is *a-limited*. A connection request is *a-limited* in a bi-state if the bi-generalized connection assignment obtained by adjoining the connection request to the bi-generalized assignment realized by the bi-state is *a-limited*.

A *wide-sense nonblocking a-limited bi-generalized connector* is a network for which there exists a set of distinguished bi-states, called the *safe* bi-states, with the following properties: (1) the empty bi-state is safe; (2) any bi-state below a safe bi-state is itself safe; and (3) given any safe bi-state and any *a-limited* connection request in that bi-state, there exists a safe bi-state above the given bi-state and containing a route satisfying the given request.

We shall use the term “bi-generalized connector” in place of “bi-generalized ∞ -limited connector.”

Our main result in this section is the following.

THEOREM 5. *For every fixed $k \geq 2$ and $n \rightarrow \infty$, a wide-sense nonblocking bi-generalized n -connector can be explicitly constructed with depth k and size $O(n^{1+2/k})$.*

For the proof of Theorem 5 we shall need the following lemma.

LEMMA 5.1. *Let p and q be distinct primes congruent to one modulo four. Then a $(q+1)$ -network of depth one can be explicitly constructed in which every input has $p+1$ outputs as neighbors, every output is a neighbor of $p+1$ inputs and every set of x inputs with $1 \leq x \leq 12(q+1)/(p+1)$ has at least $x(p+1)/16$ neighbors.*

Proof. By the results of Lubotzky, Phillips, and Sarnak [LPS], an undirected graph G can be explicitly constructed in which there are $q+1$ vertices, $p+1$ edges incident with each vertex and in which the adjacency matrix $M(G)$ of G has $p+1$ as a simple eigenvalue and has all other eigenvalues at most $2p^{1/2}$ in absolute value. Let the network F be the double cover of G ; that is, let the inputs and outputs of F each correspond to the vertices of G and let the input v have the output w as a neighbor if and only if the vertices of G corresponding to v and w are adjacent. Clearly F is a $(q+1)$ -network of depth one in which every input has $p+1$ neighbors and every output is a neighbor of $p+1$ inputs. Let $M(F)$ denote the matrix whose rows are indexed by inputs of F , whose columns are indexed by outputs of F and whose (v, w) entry is the number of edges from v to w . Then $M(F)^T M(F) = M(G)^2$ has $(p+1)^2$ as a simple eigenvalue and all other eigenvalues at most $4p$ in absolute value. Let $\epsilon = 4p/(p+1)^2$ and let $f(\xi) = \xi/(\xi + \epsilon(1 - \xi))$. By the result of Tanner [T], if X is a set of inputs and $F(X)$ is the set of neighbors of X , then $|F(X)|/(q+1) \geq f(|X|/(q+1))$. Let $\delta = 4/(p+1)$. Then $\epsilon \leq \delta$ and $f(3\delta) \geq 3/4$. Since f is concave, if $|X| \leq 3\delta(q+1)$, then $|F(X)| \geq (q+1)f(|X|/(q+1)) \geq (|X|/3\delta)f(3\delta) \geq |X|/4\delta$. $\square$

A wide-sense nonblocking bi-generalized a -limited connector remains one if any number of inputs and outputs (together with the edges incident with them) are deleted.

A t -approximate a -limited m -juncator is an m -network in which, given any $(a - 1)$ -limited bi-state S , any set V containing t inputs and any set W containing t outputs, there exists a route ρ from some $v \in V$ to some $w \in W$ such that $S \cup \{\rho\}$ is a bi-state.

A t -approximate a -limited junctor remains one if any number of inputs and outputs (together with the edges incident with them) are deleted.

If F is an (n, m) -network, let F^* denote the (m, n) -network obtained by reversing the edges and exchanging the inputs and outputs of F . Clearly, $F^{**} = F$.

If S is a bi-state of F , let S^* denote the bi-state of F^* obtained by reversing the routes of S . Clearly, $S^{**} = S$.

Two routes are *anti-compatible* if their intersection is a final segment (possibly empty) of both of them. An *anti-state* is a set of pairwise anti-compatible routes. Clearly, if S is a state, then S^* is an anti-state.

PROPOSITION 5.2. *Let F be a wide-sense nonblocking $(a - 1 + t, t)$ -limited generalized (n, m) -concentrator and let G be an t -approximate a -limited m -juncator. Then $F \circ G \circ F^*$ is a wide-sense nonblocking a -limited bi-generalized n -connector.*

Proof. Let S be a bi-state of $F \circ G \circ F^*$. Say that S is safe if (1) S induces a safe state $S(F)$ of F and (2) S induces an anti-state $S(F^*)$ of F^* and $S(F^*)^*$ is a safe state of F . Clearly, the empty bi-state is safe and any bi-state below a safe bi-state is itself safe. Let S be a safe bi-state of $F \circ G \circ F^*$ and let (v, w) be an a -limited connection request in S . Since (v, w) is a -limited, S contains at most $a - 1$ routes.

The concentration request v is (a, t) -limited in $S(F)$, so there is a safe state $T(F)$ above $S(F)$ in which v has routes to all the outputs in some set V of t outputs of F . Similarly, there is an anti-state $T(F^*)$ above $S(F^*)$ in which all the inputs in some set W of t inputs of F^* have routes to w and such that $T(F^*)^*$ is a safe state for F .

The bi-state $S(G)$ of G induced by S contains at most $a - 1$ routes. Since G is a t -approximate a -limited m -juncator, there is a bi-state $T(G)$ above $S(G)$ containing a route from some $v' \in V$ to some $w' \in W$. Let the route ρ be the concatenation of the route from v to v' in $T(F)$, the route from v' to w' in $T(G)$, and the route from w' to w in $T(F^*)$. Then $S \cup \{\rho\}$ is a safe bi-state above S containing a route from v to w . $\square$

PROPOSITION 5.3. *Let F be an m -network with depth one in which every set of t inputs has at least $\lceil (m + a)/2 \rceil$ neighbors. Then $F \circ F^*$ is a t -approximate a -limited m -juncator.*

Proof. Let S be an $(a - 1)$ -limited bi-state of $F \circ F^*$. If V is a set of t inputs, then V has as neighbors all the outputs of F in some set V' of $\lceil (m + a)/2 \rceil$ outputs. Similarly, if W is a set of t outputs, then W has as neighbors all the inputs of F^* in some set W' of $\lceil (m + a)/2 \rceil$ inputs. Then $V' \cap W'$ contains at least $\lceil (m + a)/2 \rceil + \lceil (m + a)/2 \rceil - m \geq a$ links between F and F^* . Since S is $(a - 1)$ -limited, at most $a - 1$ links can be busy. Thus there is some idle link u in $V' \cap W'$. The link u has as neighbors some input v in V and some output w in W . Let ρ be the route from v through u to w in $F \circ F^*$. Then $S \cup \{\rho\}$ is a bi-state above S containing a route from $v \in V$ to $w \in W$. $\square$

COROLLARY 5.4. *Let q be a prime congruent to one modulo four with $32n \leq q + 1 \leq 128n$. For every $t \leq n$, a t -approximate n -limited $(q + 1)$ -juncator can be explicitly constructed with depth 2 and size $O(n^2/t)$.*

Proof. Let p be the smallest prime congruent to one modulo four such that $p + 1 \geq 12(q + 1)/t$. Then $p + 1 = O(n/t)$. Let F be the network constructed in the proof of Lemma 5.1. Then F has depth one and size $O(n^2/t)$. Furthermore, every set of t inputs has at least $3(q + 1)/4$ neighbors. It follows from Proposition 5.3 that F is a t -approximate n -limited $(q + 1)$ -juncator. $\square$

Proof of Theorem 5 for k even. For $k = 2$, the claim is trivial, so suppose $k \geq 4$.

Let q be a prime congruent to one modulo four such that $32n \leq q + 1 \leq 64n$. Let p be a prime congruent to one modulo four such that $32n^{2/k} \leq p + 1 \leq 64n^{2/k}$. Let F be the network constructed in the proof of Lemma 5.1.

Let $r = \lceil (p + 1)/32 \rceil$ and $a = \lfloor 6(q + 1)/(p + 1) \rfloor$. Then $ar \geq 2n$. By Lemma 5.1, every set of x inputs with $1 \leq x \leq 2a$ has at least $2rx$ neighbors. Then by Proposition 1 with $s = r$, F is a wide-sense nonblocking $(2n, r)$ -limited generalized $(q + 1)$ -concentrator.

Let $j = (k - 2)/2$ and $t = \lceil n^{1-2/k} \rceil$. Since $r \geq n^{2/k}$, $r^j \geq t$. By Proposition 2, $H = F^{*j}$ is a wide-sense nonblocking $(2n, t)$ -limited generalized $(q + 1)$ -concentrator.

By Corollary 5.4, a t -approximate n -limited $(q + 1)$ -network G can be explicitly constructed with depth two and size $O(n^2/t) = O(n^{1+2/k})$. By Proposition 5.2, $H \circ G \circ H^*$ is a wide-sense nonblocking n -limited bi-generalized $(q + 1)$ -connector with depth k and size $O(n^{1+2/k})$. Deleting all but n inputs and n outputs from this network completes the proof. $\square$

COROLLARY 5.5. *Let q be a prime congruent to one modulo four with $32n \leq q + 1 \leq 128n$. For every $t \leq n$, a t -approximate n -limited $(q + 1)$ -juncator can be explicitly constructed with depth one and size $O(n^3/t^2)$.*

Proof. In the network constructed in Corollary 5.4, replace each route of length two by a route of length one, then delete all links and edges incident with them. The resulting network is clearly a t -approximate n -limited $(q + 1)$ -juncator with depth one. Since in the original network each of the $(q + 1)$ links has $O(n/t)$ inputs and $O(n/t)$ outputs as neighbors, the resulting network has size $O(n^3/t^2)$. $\square$

Proof of Theorem 5 for k odd. For $k = 1$, the claim is trivial, so suppose $k \geq 3$.

Let q be a prime congruent to one modulo four such that $32n \leq q + 1 \leq 64n$. Let p be a prime congruent to one modulo four such that $32n^{2/k} \leq p + 1 \leq 64n^{2/k}$. Let F be the network constructed in the proof of Lemma 5.1.

Let $r = \lceil (p + 1)/32 \rceil$ and $a = \lfloor 6(q + 1)/(p + 1) \rfloor$. Then $ar \geq 2n$. By Lemma 5.1, every set of x inputs with $1 \leq x \leq 2a$ has at least $2rx$ neighbors. Then by Proposition 1 with $s = r$, F is a wide-sense nonblocking $(2n, r)$ -limited generalized $(q + 1)$ -concentrator.

Let $j = (k - 1)/2$ and $t = \lceil n^{1-1/k} \rceil$. Since $r \geq n^{2/k}$, $r^j \geq t$. By Proposition 2, $H = F^{*j}$ is a wide-sense nonblocking $(2n, t)$ -limited generalized $(q + 1)$ -concentrator.

By Corollary 5.4, a t -approximate n -limited $(q + 1)$ -network G can be explicitly constructed with depth one and size $O(n^3/t^2) = O(n^{1+2/k})$. By Proposition 5.2, $H \circ G \circ H^*$ is a wide-sense nonblocking n -limited bi-generalized $(q + 1)$ -connector with depth k and size $O(n^{1+2/k})$. Deleting all but n inputs and n outputs from this network completes the proof. $\square$

Since the networks constructed in Theorem 5 rely on Proposition 1, no efficient algorithm is known for finding safe routes. The remaining results of this section mitigate this defect in some special cases.

THEOREM 6. *Wide-sense nonblocking bi-generalized n -connectors can be explicitly constructed with depth three and size $O(n^{5/3})$, and with depth four and size $O(n^{3/2})$.*

For the proof of Theorem 6 we shall need the following lemma.

LEMMA 6.1. *For every prime power q , a (q^3) -network of depth one can be explicitly constructed in which every input has q^2 neighbors and any two distinct inputs have at most q common neighbors.*

Proof. Let $\text{GF}(q)$ be the field with q elements. Let the outputs correspond to the q^3 points in the three-dimensional affine space over $\text{GF}(q)$. Let the inputs correspond to any q^3 planes in this space. (There are $q^3 + q^2 + q$ planes in the space.) Let the input v have the output w as a neighbor if the plane corresponding to v contains the point corresponding to w . Clearly, each plane contains q^2 points and any two distinct planes have at most the q points of a line in common. $\square$

Let $F_3(q)$ denote the network constructed in the proof of Lemma 6.1.

PROPOSITION 6.2. *The network $F_3(2^\lambda)$ is a wide-sense nonblocking $(2^{3\lambda-5}, 2^{2\lambda-1})$ -limited generalized concentrator.*

Proof. Let the safe states be those in which every input has at most $2^{2\lambda-2}$ neighbors that terminate routes originating at other inputs. Clearly the empty state is safe, and any state below a safe state is itself safe. Let S be a safe state, let v be a $(2^{3\lambda-5}, 2^{2\lambda-1})$ -limited concentration request in S , and suppose that there is no safe state above S containing a route from v to an output that is idle in S . We shall derive a contradiction.

Since v is $(2^{3\lambda-5}, 2^{2\lambda-1})$ -limited in S , there are routes from v to at most $2^{\lambda-1}$ neighbors of v . Since S is safe, at most $2^{2\lambda-2}$ other neighbors of v are busy. Say that an output w is *critical* if w is a neighbor of v , w is idle in S and $S \cup \{(v, w)\}$ is not safe. Then there are at least $2^{2\lambda} - 2^{2\lambda-1} - 2^{2\lambda-2} = 2^{2\lambda-2}$ critical outputs.

Say that an input is *critical* if u is distinct from v and $2^{2\lambda-2}$ neighbors of u terminate routes that originate at inputs other than u . Each critical output w must be the neighbor of some critical input, else $S \cup \{(v, w)\}$ would be safe. Each critical input u can have at most 2^λ critical outputs as a neighbor, else the distinct inputs v and u would have more than 2^λ common neighbors. Thus there must be at least $2^{\lambda-2}$ critical inputs.

Let X be a set of $2^{\lambda-2}$ critical inputs. Each critical input in X has at least $2^{2\lambda-2}$ busy neighbors. Furthermore, each of the $\binom{2^{\lambda-2}}{2}$ pairs of distinct critical inputs in X has at most 2^λ busy neighbors in common. Thus there are at least

$$2^{\lambda-2} \cdot 2^{2\lambda-2} - \binom{2^{\lambda-2}}{2} \cdot 2^\lambda \geq 2^{3\lambda-5}$$

busy outputs in S . This contradicts the assumption that v is $(2^{3\lambda-5}, 2^{2\lambda-1})$ -limited in S . $\square$

Proof of the first assertion of Theorem 6. We may assume that $n = 2^{3\lambda-6}$ for some integer $\lambda \geq 3$. Let $t = 2^{2\lambda-1} = O(n^{2/3})$. Let F be the network obtained by deleting all but n inputs from $F_3(2^\lambda)$. By Proposition 6.2, F is a wide-sense nonblocking $(2n, t)$ -limited generalized $(n, 64n)$ -concentrator with depth one and size $O(n^{5/3})$.

Let q be a prime congruent to one modulo four with $64n \leq q + 1 \leq 128n$. By Corollary 5.5, a t -approximate n -limited $(q + 1)$ -junctor can be explicitly constructed with depth one and size $O(n^3/t^2) = O(n^{5/3})$. Let G be the network obtained by deleting all but $64n$ inputs and $64n$ outputs from this network. By Proposition 5.2, $F \circ G \circ F^*$ is a wide-sense nonblocking bi-generalized n -connector with depth three and size $O(n^{5/3})$, which completes the proof of the first assertion. $\square$

To prove the second assertion we shall need the following lemma.

LEMMA 6.3. *For every prime power q , a (q^2) -network of depth one can be explicitly constructed in which every input has q neighbors and any two distinct inputs have at most one common neighbor.*

Proof. Let $\text{GF}(q)$ be the field with q elements. Let the outputs correspond to the q^2 points in the affine plane over $\text{GF}(q)$. Let the inputs correspond to any q^2 lines in this plane. (There are $q^2 + q$ lines in the plane.) Let the input v have the output w as a neighbor if the line corresponding to v contains the point corresponding to w . Clearly, each line contains q points and any two distinct lines have at most one point in common. $\square$

Let $F_2(q)$ denote the network constructed in the proof of Lemma 6.3.

PROPOSITION 6.4. *The network $F_2(2^\lambda)$ is a wide-sense nonblocking $(2^{2\lambda-5}, 2^{\lambda-1})$ -limited generalized concentrator.*

Proof. Let the safe states be those in which every input has at most $2^{\lambda-2}$ neighbors that terminate routes originating at other inputs. Clearly the empty state is safe, and any state below a safe state is itself safe. Let S be a safe state, let v be a $(2^{2\lambda-5}, 2^{\lambda-1})$ -limited

concentration request in S , and suppose that there is no safe state above S containing a route from v to an output that is idle in S . We shall derive a contradiction.

Since v is $(2^{2\lambda-5}, 2^{\lambda-1})$ -limited in S , there are routes from v to at most $2^{\lambda-1}$ neighbors of v . Since S is safe, at most $2^{\lambda-2}$ other neighbors of v are busy. Say that an output w is *critical* if w is a neighbor of v , w is idle in S and $S \cup \{(v, w)\}$ is not safe. Then there are at least $2^\lambda - 2^{\lambda-1} - 2^{\lambda-2} = 2^{\lambda-2}$ critical outputs.

Say that an input u is *critical* if u is distinct from v and $2^{\lambda-2}$ neighbors of u terminate routes that originate at inputs other than u . Each critical output w must be the neighbor of some critical input, else $S \cup \{(v, w)\}$ would be safe. Each critical input u can have at most one critical output as a neighbor, else the distinct inputs v and u would have more than one common neighbor. Thus there must be at least $2^{\lambda-2}$ critical inputs.

Let X be a set of $2^{\lambda-2}$ critical inputs. Each critical input in X has at least $2^{\lambda-2}$ busy neighbors. Furthermore, each of the $\binom{2^{\lambda-2}}{2}$ pairs of distinct critical inputs in X has at most one busy neighbor in common. Thus there are at least

$$2^{\lambda-2} \cdot 2^{\lambda-2} - \binom{2^{\lambda-2}}{2} \cdot 1 \geq 2^{2\lambda-5}$$

busy outputs in S . This contradicts the assumption that v is $(2^{2\lambda-5}, 2^{\lambda-1})$ -limited in S . $\square$

Proof of the second assertion of Theorem 6. We may assume that $n = 2^{2\lambda-6}$ for some integer $\lambda \geq 4$. Let $t = 2^{\lambda-1} = O(n^{1/2})$. Let F be the network obtained by deleting all but n inputs from $F_2(2^\lambda)$. By Proposition 6.4, F is a wide-sense nonblocking $(2n, t)$ -limited generalized $(n, 64n)$ -concentrator with depth one and size $O(n^{3/2})$.

Let q be a prime congruent to one modulo four with $64n \leq q + 1 \leq 128n$. By Corollary 5.4, a t -approximate n -limited $(q + 1)$ -juncter can be explicitly constructed with depth two and size $O(n^2/t) = O(n^{3/2})$. Let G be the network obtained by deleting all but $64n$ inputs and $64n$ outputs from this network. By Proposition 5.2, $F \circ G \circ F^*$ is a wide-sense nonblocking bi-generalized n -connector with depth four and size $O(n^{3/2})$, which completes the proof of the second assertion. $\square$

5. Networks with unlimited depth. In previous sections we have used Proposition 1 to construct various networks with limited depth. In this section we shall use it in situations where depth is not constrained. We shall show that wide-sense nonblocking generalized n -connectors and bi-generalized n -connectors can be constructed with size $O(n \log n)$. That these results are best possible (to within constant factors) is shown by a result of Shannon [S] to the effect that even “rearrangeable n -connectors” require size $\Omega(n \log n)$. (Indeed, Pippenger and Valiant [PV] have shown that even networks satisfying a still weaker property, that of being “rearrangeable n -shifters,” must have size $\Omega(n \log n)$.)

The previously best upper bound for wide-sense nonblocking generalized connectors was $O(n(\log n)^2)$, due to Pippenger [P1]. This construction requires wide-sense nonblocking (ungeneralized) connectors as components. That size $O(n \log n)$ is sufficient for these networks (and indeed, even for networks that satisfy a stronger property, that of being “strictly nonblocking connectors”) is due to Bassalygo and Pinsker [BP1]. The proof of Bassalygo and Pinsker is nonconstructive (in the same sense as Lemma 3.1), but subsequent results by others ([M1], [GG], [JM], [LPS], [M2]) yield an explicit construction (see Pippenger [P3] for details).

That our results cannot be extended to “strictly nonblocking generalized connectors” is shown by the lower bound of $\Omega(n^2)$, due to Bassalygo and Pinsker [BP2].

The key to our results in this section is the following proposition.

PROPOSITION 7. *A wide-sense nonblocking generalized $(n, 18n)$ -concentrator can be explicitly constructed with depth $O(\log n)$ and size $O(n \log n)$.*

This proposition has the following immediate corollary.

COROLLARY 7.1. *A wide-sense nonblocking generalized n -connector can be explicitly constructed with depth $O(\log n)$ and size $O(n \log n)$.*

To prove Corollary 7.1 we shall need the following proposition.

PROPOSITION 7.2. *Let F be a wide-sense nonblocking (a, r) -limited generalized (n, m) -concentrator and let G be a wide-sense nonblocking (m, n) -connector. Then $F \circ G$ is a wide-sense nonblocking (a, r) -limited generalized n -connector.*

Proof. Let the safe states of $F \circ G$ be those that induce safe states of F and G . Clearly, the empty state is safe and any state below a safe state is itself safe. Let S be a safe state of $F \circ G$ and let the connection request (v, w) be (a, r) -limited in S .

The concentration request v is (a, r) -limited in $S(F)$, so there exists a safe state $T(F)$ above $S(F)$ containing a route from v to an output u of F that is idle in $S(F)$. The connection request (u, w) is a -limited in $S(G)$, so there exists a safe state $T(G)$ above $S(G)$ containing a route from u to w .

The state obtained by adjoining to S the concatenation of the route from v to u in $T(F)$ with the route from u to w in $T(G)$ is a safe state above S containing a route from v to w . $\square$

Proof of Corollary 7.1. Let F be the network constructed in the proof of Proposition 7, and let G be an explicitly constructed nonblocking $(18n, n)$ -connector with depth $O(\log n)$ and size $O(n \log n)$ (see [P3] for a construction). By Proposition 7.2, $F \circ G$ satisfies the claim. $\square$

We can also extend the foregoing corollary to bi-generalized connectors.

COROLLARY 7.3. *A wide-sense nonblocking bi-generalized n -connector can be explicitly constructed with depth $O(\log n)$ and size $O(n \log n)$.*

To prove Corollary 7.3 we shall need the following proposition.

PROPOSITION 7.4. *Let F be a wide-sense nonblocking (a, a) -limited generalized (n, m) -concentrator and let G be a wide-sense nonblocking a -limited m -connector. Then $F \circ G \circ F^*$ is a wide-sense nonblocking bi-generalized n -connector.*

Proof. Let S be a bi-state of $F \circ G \circ F^*$. Say that S is safe if (1) S induces a safe state $S(F)$ of F ; (2) S induces a safe state $S(G)$ of G ; and (3) S induces an anti-state $S(F^*)$ of F^* and $S(F^*)^*$ is a safe state of F . Clearly, the empty bi-state is safe and any bi-state below a safe bi-state is itself safe. Let S be a safe bi-state of $F \circ G \circ F^*$ and let the connection request (v, w) be a -limited in S .

The concentration request v is a -limited in $S(F)$, so there exists a safe state $T(F)$ above $S(F)$ containing a route from v to an output v' of F that is idle in $S(F)$. Similarly, there is an anti-state $T(F^*)$ above $S(F^*)$ containing a route from an input w' of F^* that is idle in $S(F^*)$ to w and such that $T(F^*)^*$ is a safe state for F . The connection request (v', w') is a -limited in $S(G)$, so there exists a safe state $T(G)$ above $S(G)$ containing a route from v' to w' .

Let the route ρ be the concatenation of the route from v to v' in $T(F)$, the route from v' to w' in $T(G)$, and the route from w' to w in $T(F^*)$. Then $S \cup \{\rho\}$ is a safe bi-state above S containing a route from v to w . $\square$

Proof of Corollary 7.3. Let F be the network constructed in the proof of Proposition 7, and let G be an explicitly constructed nonblocking $(18n)$ -connector with depth $O(\log n)$ and size $O(n \log n)$ (see [P3] for a construction). By Proposition 7.4, $F \circ G \circ F^*$ satisfies the claim. $\square$

Proof of Proposition 7. Let $p = 53$ and let q be a prime congruent to one modulo four with $9n \leq q + 1 \leq 18n$. Let F be the network of depth one and size $O(n)$ constructed in the proof of Lemma 5.1. Then every set of x inputs with $1 \leq x \leq 2n$ has at least $3x$ neighbors in F . By Proposition 1 with $r = 2$ and $s = 1$, F is a wide-sense nonblocking $(n, 2)$ -limited generalized $(q + 1)$ -concentrator.

Let $k = \lceil \log_2 n \rceil$. By Proposition 2, F^{*k} is a wide-sense nonblocking (n, n) -limited generalized $(q + 1)$ -concentrator. Deleting all but n inputs and adding $18n - (q + 1)$ isolated outputs completes the proof. $\square$

The results of this paper that rely on Proposition 1 do not provide efficient algorithms for finding safe routes. In special cases (Theorems 4 and 6) we have been able to avoid this defect. It may be possible to extend the methods used in these special cases to obtain a version of Proposition 1 that allows an efficient algorithm for finding safe routes, but such an extension has eluded our efforts thus far.

REFERENCES

- [B1] V. E. BENEŠ, *Heuristic remarks and mathematical problems regarding the theory of switching systems*, Bell System Tech. J., 41 (1962), pp. 1201–1247.
- [B2] ———, *Semilattice characterization of nonblocking networks*, Bell System Tech. J., 52 (1973), pp. 697–706.
- [B3] ———, *Blocking states in connecting networks made of square switches arranged in stages*, Bell System Tech. J., 60 (1981), pp. 511–521.
- [BKPVY] M. BLUM, R. M. KARP, C. H. PAPIDIMITRIOU, O. VORNBERGER, AND M. YANNAKAKIS, *The complexity of superconcentrators*, Inform. Process. Lett., 13 (1981), pp. 164–167.
- [BP1] L. A. BASSALYGO AND M. S. PINSKER, *Complexity of an optimum nonblocking switching network without reconnections*, Problems Inform. Transmission, 9 (1974), pp. 64–66.
- [BP2] ———, *Asymptotically optimal networks for generalized rearrangeable switching and generalized switching without rearrangement*, Problemy Peredachi Informatsii, 16 (1980), pp. 94–98.
- [Ca] D. G. CANTOR, *On non-blocking switching networks*, Networks, 1 (1971), pp. 367–377.
- [CI] C. CLOS, *A study of non-blocking switching networks*, Bell System Tech. J., 32 (1953), pp. 406–424.
- [DDPW] D. DOLEV, C. DWORK, N. PIPPENGER, AND A. WIDGERSON, *Superconcentrators, generalizers and generalized connectors with limited depth*, ACM Symposium on Theory of Computing, 15 (1983), pp. 42–51.
- [F] J. FRIEDMAN, *A lower bound for strictly non-blocking networks*, Combinatorica, to appear.
- [FFP] P. FELDMAN, J. FRIEDMAN, AND N. PIPPENGER, *Non-blocking networks*, ACM Symposium on Theory of Computing, 18 (1986), pp. 247–254.
- [FP] J. FRIEDMAN AND N. PIPPENGER, *Expanding graphs contain all small trees*, Combinatorica, 7 (1987), pp. 71–76.
- [GG] O. GABBER AND Z. GALIL, *Explicit construction of linear-sized superconcentrators*, J. Comput. System Sci., 22 (1981), pp. 407–420.
- [JM] S. JIMBO AND A. MARUOKA, *Expanders obtained from affine transformations*, ACM Symposium on Theory of Computing, 17 (1985), pp. 88–97.
- [LPS] A. LUBOTZKY, R. PHILLIPS, AND P. SARNAK, *Explicit expanders and the Ramanujan conjectures*, ACM Symposium on Theory of Computing, 18 (1986), pp. 240–246.
- [M1] G. A. MARGULIS, *Explicit constructions of concentrators*, Problems Inform. Transmission, 9 (1973), pp. 325–332.
- [M2] ———, *Arithmetic groups and graphs without short cycles*, Problemy Peredachi Informatsii, to appear.
- [MJ] G. M. MASSON AND B. W. JORDAN, JR., *Generalized multi-stage connection networks*, Networks, 2 (1972), pp. 191–209.
- [NS] D. NASSIMI AND S. SAHNI, *Parallel permutation and sorting algorithms and a new generalized connection network*, J. Assoc. Comput. Mach., 29 (1982), pp. 642–667.
- [O] YU. P. OFMAN, *A universal automaton*, Trans. Moscow Math. Soc., 14 (1965), pp. 200–215.
- [P1] N. PIPPENGER, *The complexity theory of switching networks*, Ph.D. thesis, Department of Electrical Engineering, Massachusetts Institute of Technology, Cambridge, MA, 1973.

- [P2] ———, *On rearrangeable and nonblocking switching networks*, J. Comput. System Sci., 17 (1978), pp. 145–162.
- [P3] ———, *Telephone switching networks*, AMS Proc. Symposium on Applied Mathematics, 26 (1982), pp. 101–133.
- [PU] D. PELEG AND E. UPFAL, *Constructing disjoint paths on expander graphs*, ACM Symposium on Theory of Computing, 19 (1987), pp. 264–273.
- [PV] N. PIPPENGER AND L. G. VALIANT, *Shifting graphs and their applications*, J. Assoc. Comput. Mach., 23 (1976), pp. 423–432.
- [PY] N. PIPPENGER AND A. C.-C. YAO, *Rearrangeable networks with limited depth*, SIAM J. Algebraic Discrete Methods, 3 (1982), pp. 411–417.
- [S] C. E. SHANNON, *Memory requirements in a telephone exchange*, Bell System Tech. J., 29 (1950), pp. 343–349.
- [T] R. M. TANNER, *Explicit construction of concentrators from generalized N -gons*, SIAM J. Algebraic Discrete Methods, 5 (1984), pp. 287–293.

FROM LINEAR SEPARABILITY TO UNIMODALITY: A HIERARCHY OF PSEUDO-BOOLEAN FUNCTIONS*

P. L. HAMMER†‡, B. SIMEONE†‡¶, TH. M. LIEBLING§, AND D. DE WERRA§¶

Abstract. When an injective pseudo-Boolean function $f : B^n \rightarrow \mathbb{R}$ is minimized, where $B^n = \{0, 1\}^n$ is the set of vertices of the unit-hypercube, it is natural to consider so-called greedy vertex-following algorithms. These algorithms construct a sequence of neighbouring (Hamming distance 1) vertices with decreasing f -value. The question arises as to when such algorithms will find the global optimum given any starting point. This paper describes a hierarchy of such classes of functions that are shown to strictly contain each other. These classes are, in increasing order of generality, the threshold, the saddle-free, the pseudomodular, the completely unimodal, the unimodal, and the unimin (respectively, unimax) functions. Some considerations as to the complexity of the above-mentioned class of algorithms are also made.

Key words. pseudo-Boolean functions, combinatorial optimization, local improvement algorithms, separability, convexity, unimodality

AMS(MOS) subject classification. 05C

1. Introduction. If $B^n = \{0, 1\}^n$ is the Boolean n -cube, a *pseudo-Boolean function* f associates with every n -tuple $\mathbf{x} = (x_1, \dots, x_n)$ in B^n a real value $f(\mathbf{x})$. A Boolean function associates with every $\mathbf{x}$ in B^n a value in $B^1 = \{0, 1\}$.

Our purpose is to describe several classes of pseudo-Boolean functions that are in some sense extensions of the notion of threshold Boolean functions. The relations between these classes will be described and this will lead us to describe a hierarchy of classes.

For the classes studied in this paper, simplex-type algorithms may be used for optimizing the functions; our purpose here is to exhibit several classes of functions and show the equivalence of some properties rather than to develop efficient algorithms for optimizing such functions.

For all definitions and concepts in pseudo-Boolean optimization, the reader is referred to [2].

2. Linear separability and saddles. Throughout this paper we shall assume that all pseudo-Boolean functions f are *injective*, i.e., for any $\mathbf{x}, \mathbf{y} \in B^n$ with $\mathbf{x} \neq \mathbf{y}$, we have $f(\mathbf{x}) \neq f(\mathbf{y})$. This assumption will be crucial in some of the following proofs.

A pseudo-Boolean function f is called a *threshold function* if for any real number r there exists a vector $\mathbf{a}' \in \mathbb{R}^n$ and a real number b' such that

$$f(\mathbf{x}) \geq r \quad \text{iff} \quad \mathbf{a}'\mathbf{x} \geq b'.$$

This means that for each real value r there exists a hyperplane $H_r = \{\mathbf{y} | \mathbf{a}'\mathbf{y} = b'\}$ that separates the vectors $\mathbf{x}$ in B^n with $f(\mathbf{x}) \geq r$ from those with $f(\mathbf{x}) < r$. Observe that this definition is a direct extension of the concept of threshold Boolean functions (for these functions only the value $r = 1$ has to be considered).

Recall that the *Hamming distance* $d^H(\mathbf{x}, \mathbf{y})$ between two vectors $\mathbf{x}, \mathbf{y}$ in B^n is defined by $d^H(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^n |x_i - y_i|$. We say that $\mathbf{x}$ and $\mathbf{y}$ are *neighbours* if $d^H(\mathbf{x}, \mathbf{y}) = 1$. The

* Received by the editors July 30, 1984; accepted for publication (in revised form) May 12, 1987. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada, U.S. Air Force Office of Scientific Research grant AFOSR-0271, and National Science Foundation grant ECS-8503212.

† RUTCOR, Rutgers University, New Brunswick, New Jersey 08903.

‡ Department of Statistics, University of Rome, La Sapienza, Italy.

§ Department of Mathematics, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland.

¶ This research was carried out while this author was at the University of Waterloo, Ontario, Canada during the spring and summer of 1983.

hypercube defined by B^n will sometimes be considered as a graph, where each node corresponds to a vector $\mathbf{x}$ in B^n and where $\mathbf{x}$ and $\mathbf{y}$ are linked if they are neighbours.

Given any $\mathbf{x}$ in B^n , let $N^+(\mathbf{x})$ (respectively, $N^-(\mathbf{x})$) be the set of vectors $\mathbf{y}$ with $f(\mathbf{y}) > f(\mathbf{x})$ (respectively, $f(\mathbf{y}) < f(\mathbf{x})$). One can easily prove the following proposition.

PROPOSITION 2.1. *Let f be a pseudo-Boolean function. The following two conditions are equivalent:*

- (a) *f is a threshold function;*
- (b) *Given any $\mathbf{x}$ in B^n one cannot find coefficients*

$$\alpha(\mathbf{y}) \geq 0, \quad \beta(\mathbf{z}) \geq 0 \quad (\mathbf{y} \in N^-(\mathbf{x}), \mathbf{z} \in N^+(\mathbf{x}))$$

with

$$(2.1) \quad \mathbf{x} = \sum_{\mathbf{y} \in N^-(\mathbf{x})} \alpha(\mathbf{y})\mathbf{y} - \sum_{\mathbf{z} \in N^+(\mathbf{x})} \beta(\mathbf{z})\mathbf{z},$$

$$(2.2) \quad 1 = \sum_{\mathbf{y} \in N^-(\mathbf{x})} \alpha(\mathbf{y}) - \sum_{\mathbf{z} \in N^+(\mathbf{x})} \beta(\mathbf{z}).$$

We shall now consider a collection of classes of pseudo-Boolean functions. The properties characterizing these classes are weaker than the "linear separability property" of threshold functions.

Let $\mathbf{x} = (x_1, \dots, x_n)$ be a vector in B^n . It will be convenient to consider partitions of the index set of its coordinates into three subsets A, B, C . We will then write $\mathbf{x} = (\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C)$. When we write $\bar{\mathbf{x}}_A$, this will mean that all components x_i with $i \in A$ are replaced by $\bar{x}_i = 1 - x_i$.

A pseudo-Boolean function f will be called *saddle-free* if for any $\mathbf{x}$ in B^n and for any partition A, B, C of its index set we have

$$(2.3) \quad \min \{f(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C), f(\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C)\} \leq \max \{f(\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C), f(\mathbf{x}_A, \mathbf{x}_B, \bar{\mathbf{x}}_C)\}.$$

This property may be intuitively interpreted as follows: the four vectors derived from $\mathbf{x}$ form a rectangle in B^n ; its diagonals are $(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C)$, $(\mathbf{x}_A, \bar{\mathbf{x}}_B, \bar{\mathbf{x}}_C)$ and $(\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C)$, $(\mathbf{x}_A, \mathbf{x}_B, \bar{\mathbf{x}}_C)$. The property simply says that there cannot exist a dominating diagonal, in the sense that f cannot be larger at both endpoints of one diagonal than at both endpoints of the other.

PROPOSITION 2.2. *If f is a threshold function, then f is saddle-free.*

Proof. Let f be a threshold function and assume f is not saddle-free. Then there exists a vector $\mathbf{x}$ and a partition A, B, C of the index set $\{1, \dots, n\}$ such that if $r = \min \{f(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C), f(\mathbf{x}_A, \bar{\mathbf{x}}_B, \bar{\mathbf{x}}_C)\}$ we have

$$r > \max (f(\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C), f(\mathbf{x}_A, \mathbf{x}_B, \bar{\mathbf{x}}_C)).$$

Since f is a threshold function, there exists a vector $\mathbf{a}'$ and a real number b' such that

$$\begin{aligned} \mathbf{a}'(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C) &\geq b', & \mathbf{a}'(\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C) &< b', \\ \mathbf{a}'(\mathbf{x}_A, \bar{\mathbf{x}}_B, \bar{\mathbf{x}}_C) &\geq b', & \mathbf{a}'(\mathbf{x}_A, \mathbf{x}_B, \bar{\mathbf{x}}_C) &< b'. \end{aligned}$$

By summing the left inequalities and the right inequalities we obtain

$$\begin{aligned} 2b' &\leq \mathbf{a}'((\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C) + (\mathbf{x}_A, \bar{\mathbf{x}}_B, \bar{\mathbf{x}}_C)) \\ &= \mathbf{a}'((\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C) + (\mathbf{x}_A, \mathbf{x}_B, \bar{\mathbf{x}}_C)) < 2b', \end{aligned}$$

which is a contradiction; hence f must be saddle-free. $\square$

Remark 2.1. A saddle-free pseudo-Boolean function need not be a threshold function; in [3] one can find examples by E. F. Moore (1957) and I. J. Gabelmann (1961)

illustrating a similar situation for completely monotonic Boolean functions (resembling saddle-free pseudo-Boolean functions) not having the threshold property. Recently, K. W. Hoke [7] has shown in her Ph.D. thesis that the above example may be transformed into an injective pseudo-Boolean function which is saddle-free but not threshold. Thereby it is established that threshold functions are a strict subclass of saddle-free functions.

We shall now consider a property that is closely related but not equivalent to the property examined in the previous section.

A pseudo-Boolean function f is *pseudomodular* if, for any two vectors $\mathbf{x}, \mathbf{y}$ in B^n ,

$$(2.4) \quad \min \{f(\mathbf{x}), f(\mathbf{y})\} \leq \max \{f(\mathbf{x} \wedge \mathbf{y}), f(\mathbf{x} \vee \mathbf{y})\},$$

$$(2.5) \quad \min \{f(\mathbf{x} \wedge \mathbf{y}), f(\mathbf{x} \vee \mathbf{y})\} \leq \max \{f(\mathbf{x}), f(\mathbf{y})\}.$$

Recall that $\mathbf{z} = \mathbf{x} \wedge \mathbf{y}$ (respectively, $\mathbf{x} \vee \mathbf{y}$) means that

$$z_i = \min \{x_i, y_i\} \text{ (resp. } \max \{x_i, y_i\} \text{)}.$$

Notice that there are pseudomodular functions which are not saddle-free, as can be seen in Fig. 2.1, where the values of such a function defined on B^3 are represented. This function is not saddle-free since for $A = \emptyset$, $B = \{1, 2\}$, $C = \{3\}$, $\mathbf{x}_B = (10)$, $\mathbf{x}_C = 1$,

$$f(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C) = f(101) = 4, \quad f(\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C) = f(011) = 3,$$

$$f(\mathbf{x}_A, \bar{\mathbf{x}}_B, \bar{\mathbf{x}}_C) = f(010) = 5, \quad f(\mathbf{x}_A, \mathbf{x}_B, \bar{\mathbf{x}}_C) = f(100) = 2$$

and $\min \{f(101), f(010)\} = 4 < \max \{f(011), f(100)\} = 3$. It is however pseudomodular, as can be verified (notice that the four points (101), (010), (011), and (100) cannot be put in 1-1 correspondence with a set of the form $(\mathbf{x}, \mathbf{y}, \mathbf{x} \wedge \mathbf{y}, \mathbf{x} \vee \mathbf{y})$).

The class of pseudomodular functions includes the class of saddle-free functions since by definition a pseudomodular function has to satisfy conditions (2.3) only for some specific sets of four nodes $(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C)$, $(\mathbf{x}_A, \bar{\mathbf{x}}_B, \bar{\mathbf{x}}_C)$, $(\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C)$, $(\mathbf{x}_A, \mathbf{x}_B, \bar{\mathbf{x}}_C)$, namely those for which there is a 1-1 correspondence with a set of the form $\mathbf{x}, \mathbf{y}$,

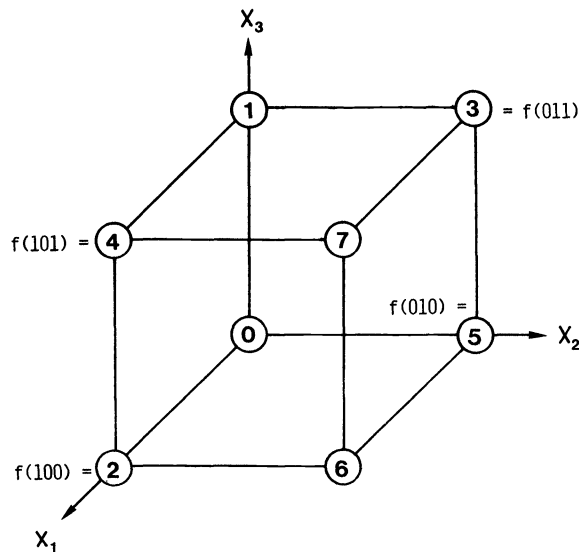


FIG. 2.1

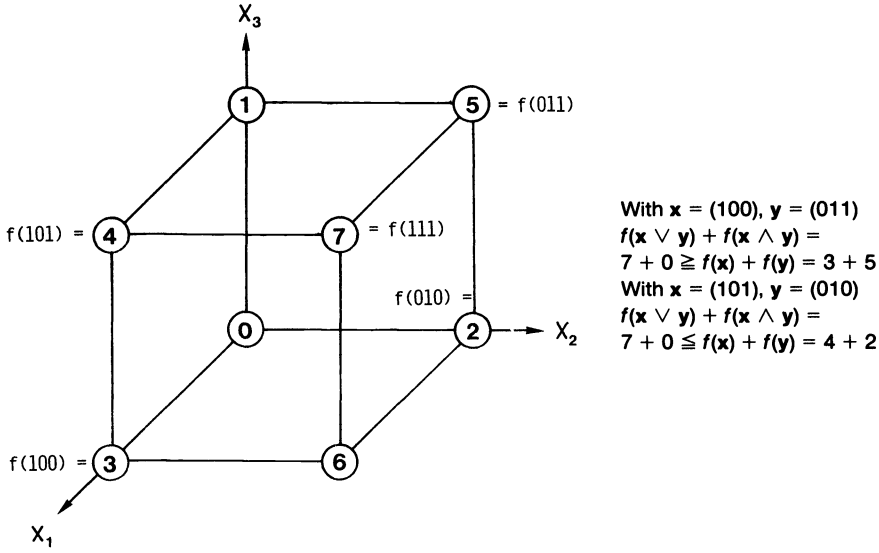


FIG. 2.2. A threshold function that is neither supermodular nor submodular.

$\mathbf{x} \wedge \mathbf{y}$, $\mathbf{x} \vee \mathbf{y}$. Notice that every subset of the form $\mathbf{x}$, $\mathbf{y}$, $\mathbf{x} \wedge \mathbf{y}$, $\mathbf{x} \vee \mathbf{y}$ can be written in the form $(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C)$, $(\mathbf{x}_A, \bar{\mathbf{x}}_B, \bar{\mathbf{x}}_C)$, $(\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C)$, $(\mathbf{x}_A, \mathbf{x}_B, \bar{\mathbf{x}}_C)$ by setting

$$A = \{i | x_i = y_i\}, \quad B = \{i | x_i = 1, y_i = 0\},$$

$$C = \{i | x_i = 0, y_i = 1\} \quad \text{and} \quad \mathbf{x} = (\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C),$$

$$\mathbf{y} = (\mathbf{x}_A, \bar{\mathbf{x}}_B, \bar{\mathbf{x}}_C), \quad \mathbf{x} \wedge \mathbf{y} = (\mathbf{x}_A, \bar{\mathbf{x}}_B, \mathbf{x}_C), \quad \mathbf{x} \vee \mathbf{y} = (\mathbf{x}_A, \mathbf{x}_B, \bar{\mathbf{x}}_C).$$

Therefore we can state the following proposition.

PROPOSITION 2.3. *If f is a saddle-free function, then f is pseudomodular.*

The name of pseudomodular functions may suggest that there exists some relation with submodular or supermodular functions. Recall that a pseudo-Boolean function is submodular if for any $\mathbf{x}, \mathbf{y}$ in B^n ,

$$f(\mathbf{x} \vee \mathbf{y}) + f(\mathbf{x} \wedge \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y}).$$

In fact, if f is a submodular function, then f satisfies (2.5); this holds because

$$\begin{aligned} 2 \min \{f(\mathbf{x} \vee \mathbf{y}), f(\mathbf{x} \wedge \mathbf{y})\} &\leq f(\mathbf{x} \vee \mathbf{y}) + f(\mathbf{x} \wedge \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y}) \\ &\leq 2 \max (f(\mathbf{x}), f(\mathbf{y})). \end{aligned}$$

Clearly, if f is supermodular, then it satisfies (2.4), so that modular functions (i.e., functions for which $f(\mathbf{x} \vee \mathbf{y}) + f(\mathbf{x} \wedge \mathbf{y}) = f(\mathbf{x}) + f(\mathbf{y})$ for any $\mathbf{x}, \mathbf{y}$) are pseudomodular. We should mention that the class of modular pseudo-Boolean functions is precisely the class of linear functions. There are, however, pseudomodular functions which are not modular (see, for instance, the example in Fig. 2.1).

Remark 2.2. There are threshold functions (and hence pseudomodular functions) that are neither supermodular nor submodular. An example is given in Fig. 2.2.

3. Unique local optima. We will now examine several properties that are related to the faces of the hypercube B^n . Recall that a face of dimension p is the collection of vectors obtained by fixing $n - p$ arbitrary coordinates and letting the p remaining coordinates take all possible values.

In the hypercube B^n a path $P = (x_1, x_2, \dots, x_k)$ going through nodes $x_1, x_2, \dots, x_k$ is *increasing* if $f(x_1) < f(x_2) < \dots < f(x_k)$.

Let us consider a pseudo-Boolean function defined on B^n such that for any face F the following holds: for any x in F , there is an increasing path in F from x to the (unique) maximum x^* of f in F . The unicity of x^* follows simply from the fact that f is injective.

Later we shall consider functions for which the above property holds only on the face $F = B^n$.

PROPOSITION 3.1. *For a pseudo-Boolean function f the following statements are equivalent:*

- (a) *For any face F and for any x in F there exists an increasing path in F from x to the (unique) maximum x^* of f in F .*
- (b) *f has a unique local maximum in each face.*
- (c) *For any face F and for any x in F there exists an increasing path in F from the (unique) minimum x^* of f in F to x .*
- (d) *f has a unique local minimum in each face.*

Proof of Proposition 3.1. We shall only prove here the equivalence of (b) and (d); the other equivalences are easy to establish.

(b) $\Leftrightarrow$ (d). Although this result can be found in Brøndsted [1], we give a proof here to make the paper self contained. We shall show that (b) implies (d); a similar reasoning would show the reverse implication.

Let f be a function satisfying (b) and assume it does not have property (d). Then there exist a face F and two vectors x, y in F which are local minima. We choose a face F containing two local minima x and y and of smallest dimension. The distance $d^H(x, y)$ is the dimension of F . Clearly $d^H(x, y) \geq 2$; if $d^H(x, y) = 2$ the face F has dimension 2; if x, y are local minima, the other nodes x', y' of F are distinct local maxima, and so f does not satisfy (b). Therefore we may assume that $d_H(x, y) \geq 3$. $\square$

CLAIM. *For any neighbour u of x or of y in F , there is an increasing path in F from x to u and from y to u .*

Proof of the claim. Let u be a neighbour of x . There is an increasing path from x to u since x is a local minimum of F . Now u is not a neighbour of y (since $d^H(x, y) \geq 3$). Consider a face F' containing y and u and of smallest dimension; since $d^H(y, u) < d^H(x, y)$, F' has a dimension smaller than F . So by the minimality of F , there is a unique local minimum in F' (it is y) and by the equivalence of (c) and (d), there is an increasing path from y to u . A similar line of reasoning can be followed for a neighbour u of y , and the claim is proven. $\square$

Now consider a neighbour u of x . An increasing path will be constructed backwards as follows: from the claim, there is in F an increasing path P from y to u ; let v be the first vertex after y on P ; take the subpath $P(u, v)$ (decreasing path from u to v). Notice that $v \neq u$ (since $d^H(x, y) \geq 3$). According to the claim, there is an increasing path Q from x to v in F ; let u' be the first vertex after x on Q . We take the subpath $Q(v, u')$ (decreasing path from v to u'). If $u' = u$, $P(u, v)$ is a decreasing path from u to v and $Q(v, u)$ a decreasing path from v to u . This is impossible since f is injective by assumption.

If $u' \neq u$, we continue to extend the decreasing path formed with $P(u, v)$, $Q(v, u')$ by introducing a subpath $P(u', v')$ (where v' is a neighbour of v) which again exists from the claim. We continue this construction; since the number of neighbours of x and of y is finite, we shall reach a vertex $u^{(k)}$ or $v^{(k)}$ which has already been visited. This will create a closed decreasing path. This again is impossible since f is injective. Hence the case $d^H(x, y) \geq 3$ is not possible. This ends the proof. $\square$

In the remainder of the paper, the functions having the properties described in Proposition 3.1 will simply be called *completely unimodal*. We shall now show that the class of completely unimodal functions includes the pseudomodular functions.

With the above claim, the following remarkable property of completely unimodal functions is also established.

PROPOSITION 3.2. *If any one of the statements in Proposition 3.1 holds for every two-dimensional face of B^n , then it also holds for all faces of higher dimension.*

PROPOSITION 3.3. *If f is pseudomodular, then f is completely unimodal.*

Proof. Assume that f is not completely unimodal. Then there exists a face F containing two local maxima x and y ; we may choose an F with the smallest possible dimension. So $d^H(x, y)$ is the dimension of F . If $x \not\leq y$ and $y \not\leq x$ we may consider the four distinct vertices $x, y, x \vee y, x \wedge y$; clearly $\max \{d^H(x \vee y, x), d^H(x \wedge y, x)\} < d^H(x, y)$. So by our minimality assumption, there is an increasing path from $x \vee y$ to x in the smallest face containing these vertices (x is a local maximum in such a face); there is also an increasing path from $x \wedge y$ to x , and similarly there are increasing paths from $x \wedge y$ to y and from $x \vee y$ to y . Hence $f(x) > \max \{f(x \vee y), f(x \wedge y)\}$ and $f(y) > \max \{f(x \vee y), f(x \wedge y)\}$; this contradicts (2.4).

If $x \leq y$, then we must have $d^H(x, y) \geq 2$ since x and y are local maxima; in the smallest face containing x and y we can find two vectors x', y' such that $x' \wedge y' = x, x' \vee y' = y$; since $\max \{d^H(x' \wedge y', x'), d^H(x' \vee y', x')\} < d^H(x, y)$, we can apply the same reasoning as in the first case. Hence f is not pseudomodular. $\square$

Remark 3.1. The class of pseudomodular functions is strictly included in the class of completely unimodal functions: Fig. 3.1 represents a function f defined on B^3 which is completely unimodal but not pseudomodular. f is not pseudomodular since for $x = (110), y = (001)$, (2.4) is violated. We can verify that f is completely unimodal since it satisfies (b) in Proposition 3.1.

Given a pseudo-Boolean function f we shall consider, for any real number r , the subgraph G_+^r of the hypercube spanned by all y with $f(y) \geq r$. In the same way, for any real number r we shall call G_-^r the subgraph of the hypercube spanned by all y with $f(y) < r$. We shall now relate properties of connectivity of G_+^r and G_-^r to properties concerning the uniqueness of local optima in B^n .

For the function f defined on B^n , let $x^* = \arg \min_{B^n} f(x)$ and $\bar{x}^* = \arg \max_{B^n} f(x)$. Call f *unimax* if in B^n there is a unique local maximum $\bar{x}^*$. Similarly, a function will be

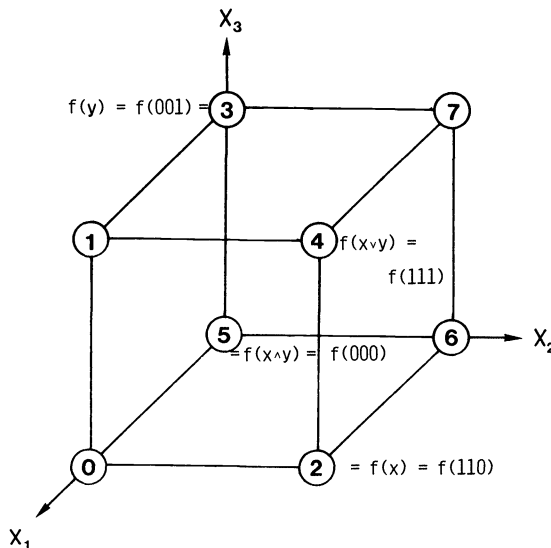


FIG. 3.1. A completely unimodal function that is not pseudomodular.

unimin if in B^n , there is a unique local minimum $\mathbf{x}^*$. It will be called *unimodal* if it is both *unimin* and *unimax*.

Remark 3.2. A *unimax* function may not be *unimin* (and conversely) as can be seen in Fig. 3.2. The vertices where f is 0 and 1 are local minima; one can verify that f has a unique local maximum. Notice also that for $r = 2$, the subgraph of G'_- induced by all vertices $\mathbf{x}$ with $f(\mathbf{x}) < 2$ is not connected. However for any r , G'_+ is connected.

PROPOSITION 3.4. *For a pseudo-Boolean function f , the following statements are equivalent:*

- (a) f is *unimodal*;
- (b) For any r , G'_+ and G'_- are connected;
- (c) For any $\mathbf{x}$ in B^n there exists an increasing path in B^n from $\mathbf{x}_*$ to $\mathbf{x}^*$ through $\mathbf{x}$.

We shall not prove these equivalences; they will follow from Proposition 3.5.

PROPOSITION 3.5. *If f is completely unimodal, then f is unimodal.*

Proof. We have to show that if f has a unique local maximum in each face, then for any r the graph G'_+ is connected. According to Proposition 3.1, f also has a unique minimum in each face and a similar proof would show that G'_- is connected. By Proposition 3.4 f will be *unimodal*. So let f be a completely *unimodal* function; let us take a real value r and two vertices $\mathbf{x}, \mathbf{y}$ in G'_+ , with $\min \{f(\mathbf{x}), f(\mathbf{y})\} \geq r$.

Consider any face F containing $\mathbf{x}$ and $\mathbf{y}$; if $d(\mathbf{x}, \mathbf{y}) = n$, assume $\mathbf{x}$ is not a maximum and choose $\mathbf{z}$ in $N^*(\mathbf{x})$. Then $d(\mathbf{z}, \mathbf{y}) = n - 1$. Consider any facet F containing $\mathbf{y}$ and $\mathbf{z}$. Let $\mathbf{x}^*$ be the maximum on F . Since f is completely *unimodal*, there is an increasing path from $\mathbf{x}$ to $\mathbf{x}^*$ and also from $\mathbf{y}$ to $\mathbf{x}^*$. All vectors $\mathbf{u}$ in these paths satisfy $f(\mathbf{u}) > r$; hence $\mathbf{x}$ and $\mathbf{y}$ are linked in G'_+ . Notice that the conclusion still holds if $\mathbf{x}$ or $\mathbf{y}$ coincides with $\mathbf{x}^*$. Hence G'_+ is connected. $\square$

We should note that the class of completely *unimodal* functions is strictly included in the class of *unimodal* functions. For instance, the function represented in Fig. 3.3 is *unimodal* but there are faces with several local minima (hence f is not completely *unimodal*): in the "central" cube there are two local minima. This also occurs for the two-dimensional face defined by the vertices, where f takes values 4, 6, 7, and 10.

Recall that a function f defined on B^n is *unimax* if for any $\mathbf{x} \in B^n$ there is a unique local maximum $\mathbf{x}^*$ of f in B^n . It is *unimin* if there is a unique local minimum $\mathbf{x}^*$ in B^n .

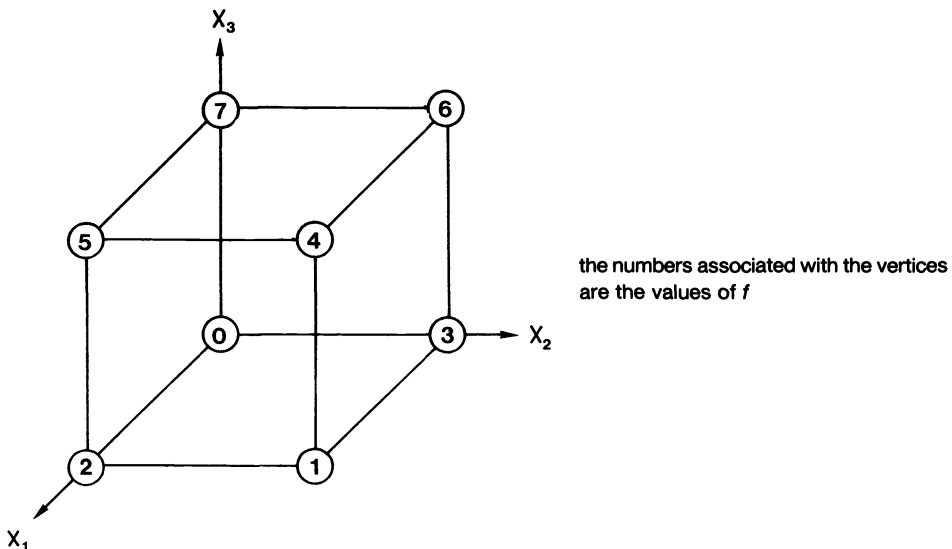


FIG. 3.2. A *unimax* function that is not *unimin*.

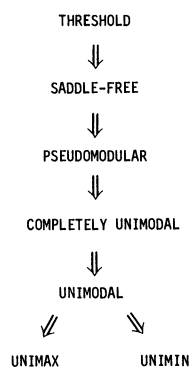


FIG. 3.5

in the same component of G_+^r . Every u on a path P joining x and y in G_+^r satisfies $f(u) > f(x) = r$; in particular the vertex u following x on P satisfies this inequality. This contradicts the fact that x is a local maximum. Hence f has a unique local maximum in B^n if G_+^r is connected.

(a) $\Rightarrow$ (c). This follows from the proof of Proposition 3.1 (simply take $F = B^n$). $\square$

We can show similarly that the following statements are equivalent:

- (1) f is unimin;
- (2) For any real r the graph G_-^r is connected;
- (3) For any x in B^n there is a decreasing path from x to x^* .

Remark 3.3. The class of unimodal functions is strictly included in the class of unimax functions. The function represented in Fig. 3.4 is unimax but not unimodal: the vertex x with $f(x) = 1$ is not contained in any increasing path from x_* with $f(x_*) = 0$ to x^* with $f(x^*) = 7$. However x^* is the unique local maximum in B^3 .

Figure 3.5 summarizes the relationships between the classes of functions studied here.

4. Remarks about algorithms. We recall that our purpose is to study classes of pseudo-Boolean functions for which simplex-type algorithms might be used for finding the maximum on B^n . In fact, the values of f are not interesting in themselves; we are concerned only with their ordering. We might try to characterize some of the previous classes of functions by orientations of the edges of the hypercube (an edge between x and y would be oriented from x to y if $f(x) < f(y)$).

In particular, unimodal functions would correspond to acyclic orientations of the edges of the hypercube such that there is exactly one node with no outgoing arc and one node with no incoming arc.

In order to maximize f we might consider vertex following algorithms moving from one vertex x in B^n to the neighbour y with $f(y) = \max \{f(z) | z \text{ neighbour of } x\}$. Such algorithms could be called greedy. We would obtain a good algorithm if we showed that we could reach the maximum x^* of f in B^n after a number of steps which is bounded by a polynomial in n (even for threshold functions, we may have a number of steps which is larger than n ; see Fig. 4.1). It is not known whether such a polynomial algorithm exists for threshold functions.

Although the worst-case performance of the greedy algorithm is not known yet, in view of the following result all hope may not be lost for finding a good algorithm to maximize completely unimodal functions.

PROPOSITION 4.1. *Let f be a completely unimodal pseudo-Boolean function defined on B^n ; then for any $x \in B^n$, there is an increasing path of length at most n from x to the maximum x^* of f in B^n .*

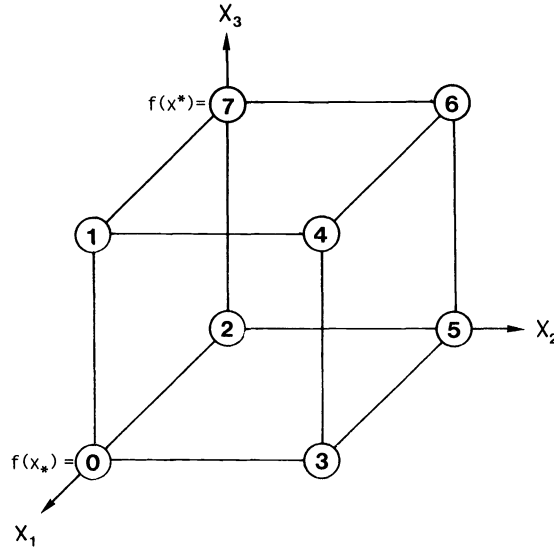


FIG. 4.1. A threshold function defined in B^3 for which the greedy algorithm needs $4 > n$ steps to go from $\mathbf{x}_*$ to $\mathbf{x}^*$.

Proof. We proceed by induction on $d^H(\mathbf{x}, \mathbf{x}^*)$. We define assertion P_k : if $d^H(\mathbf{x}, \mathbf{x}^*) = k$, then there is an increasing path of length at most k from $\mathbf{x}$ to $\mathbf{x}^*$. Clearly P_1 is true. Assume we have obtained P_k ; consider $\mathbf{x}, \mathbf{x}^*$ with $d^H(\mathbf{x}, \mathbf{x}^*) = k + 1$ and let F be a face containing $\mathbf{x}$ and $\mathbf{x}^*$ and of smallest dimension. There exists at least one neighbour $\mathbf{y}$ of $\mathbf{x}$ in F such that $f(\mathbf{x}) < f(\mathbf{y})$ (otherwise $\mathbf{x}$ would be a local maximum in F with $\mathbf{x} \neq \mathbf{x}^*$). By P_k there is an increasing path of length at most k from $\mathbf{y}$ to $\mathbf{x}^*$. Hence there exists an increasing path of length at most $k + 1$ from $\mathbf{x}$ to $\mathbf{x}^*$. The result now follows from P_n . $\square$

On the other hand, the following proposition shows that there are unimax functions for which the complexity of greedy algorithms can come close to that of complete enumeration!

PROPOSITION 4.2. *There exist unimax functions $f: B^{2^n} \rightarrow \mathbb{R}$ such that any greedy algorithm for maximizing them, when started at the minimum, will visit $2^{n+1} - 1$ adjacent vertices before terminating.*

Proof. We shall construct such a function f . Let $X = (\mathbf{x}_1, \dots, \mathbf{x}_p)$ be a sequence of distinct vertices of B^{2^n} having the following properties:

- (i) $\mathbf{x}_i$ is adjacent to $\mathbf{x}_{i+1}$ for $i = 2, \dots, p$;
- (ii) $\mathbf{x}_k$ is not adjacent to $\mathbf{x}_1, \dots, \mathbf{x}_{k-2}$ for $k = 3, \dots, p$;
- (iii) $p = 2^{n+1} - 1$.

We give a construction of a sequence X with properties (i)–(iii): partition the Boolean vectors $\mathbf{x} \in B^{2^n}$ into $\mathbf{x}_A$ and $\mathbf{x}_B$ each in B^n , to form $\mathbf{x} = (\mathbf{x}_A, \mathbf{x}_B)$. Now construct a Hamiltonian path on B^n , say $(\mathbf{w}_1, \dots, \mathbf{w}_N)$, with $N = 2^n$. Then set for $i = 1, \dots, 2N - 1$

$$\mathbf{x}_i = \begin{cases} (\mathbf{w}_{\lceil i/2 \rceil}, \mathbf{w}_{\lceil i/2 \rceil}) & \text{for } i \text{ odd,} \\ (\mathbf{w}_{i/2}, \mathbf{w}_{i/2+1}) & \text{for } i \text{ even.} \end{cases}$$

Herein $\lceil \cdot \rceil$ is the notation for integer roundup.

We immediately verify that $\mathbf{x}_i$ is adjacent to $\mathbf{x}_{i+1}$ for $i = 1, \dots, 2N - 1$ and that $\mathbf{x}_i$ is not adjacent to $\mathbf{x}_k$ for $k = 1, \dots, i - 2$ and $i = 3, \dots, 2N - 1$.

Now we construct f as follows:

(a) Set $f(\mathbf{x}_1) := 1$ and $f(\mathbf{x}_i) := 2^{2^n} - p + i$ for $i := 2, \dots, p$. Let $Y = (\mathbf{y}_1, \dots, \mathbf{y}_M)$ with $M = 2^{2^n}$ be a Hamiltonian path on B^{2^n} , i.e., a permutation of the elements of B^{2^n} such that $\mathbf{y}_{i+1}$ is adjacent to $\mathbf{y}_i$ for $i = 1, \dots, M - 1$. Such a path can be constructed inductively as follows: Let $(\hat{\mathbf{y}}_1, \dots, \hat{\mathbf{y}}_K)$ be a Hamiltonian path on B^k (i.e., $K = 2^k$). Then the sequence $((\hat{\mathbf{y}}_1, 0), \dots, (\hat{\mathbf{y}}_{K-1}, 0), (\hat{\mathbf{y}}_K, 0), (\hat{\mathbf{y}}_K, 1), (\hat{\mathbf{y}}_{K-1}, 1), \dots, (\hat{\mathbf{y}}_1, 1))$ is a Hamiltonian path on B^{k+1} . Now let $Z = (\mathbf{z}_1, \dots, \mathbf{z}_L)$ with $L = 2^{2^n} - p$ be the subsequence obtained from Y by deleting all elements of X .

(b) Set $f(\mathbf{z}_i) := i + 1$ for $i := 1, \dots, L$.

The function constructed by (a) and (b) is unimax, since there is an increasing path ending in $\mathbf{x}_p$ for all elements in X and a monotonic path in the sequence Z ending in an element adjacent to an element of X and thus having a higher value than all of its predecessors.

Now, by construction, each $\mathbf{x}_i \in X - \{\mathbf{x}_p\}$ has exactly one neighbour on B^{2^n} with a higher f -value, namely $\mathbf{x}_{i+1}$. Therefore any vertex-following algorithm started at $\mathbf{x}_1$ will visit $p = 2^{n+1} - 1$ adjacent points before terminating, as stated. $\square$

A referee pointed out to us that in his technical report [6], C. A. Tovey gives a result similar to Proposition 4.2, using a bound of Danzer and Klee on the lengths of snakes in boxes [4].

5. Final remarks. When we defined the class of completely unimodal functions as well as the following more general classes, the structural properties of the graph associated with the hypercube B^n did not play an essential role. We may therefore think of the following extension: Consider any graph G ; with G we associate a collection $\mathcal{F}$ of subsets of nodes called *faces* with the following properties: given any two nodes x and y , there exists a unique minimal face F containing x and y . F consists of x , y and all nodes which are on shortest chains between x and y . We may then define, for instance, completely unimodal functions by properties which hold on the faces of G .

f will be defined on the nodes of G ; from these values one can obtain an acyclic orientation of the edges of G (we still assume f to be injective) and the problem consists in characterizing orientations of a graph which satisfy specific conditions on a collection of subgraphs.

One may also specialize the graph G in this formulation by taking the graph of a convex polyhedron.

Acknowledgments. The authors express their gratitude to their colleagues R. Burkard and M. Minoux for the remarks and comments they made at various stages of the completion of this work.

REFERENCES

- [1] A. BRØNDSTED, *Introduction to Convex Polytopes*, Springer-Verlag, Berlin, 1983.
- [2] P. L. HAMMER AND S. RUDEANU, *Boolean Methods in Operations Research and Related Areas*, Springer-Verlag, New York, 1968.
- [3] S. T. HU, *Threshold Logic*, University of California Press, Berkeley, 1965.
- [4] V. KLEE, *What is the maximum length of a d-dimensional snake*, Amer. Math. Monthly, 77 (1970), pp. 63–65.
- [5] S. MUROGA, *Threshold Logic and Its Applications*, Wiley-Interscience, Toronto, 1971.
- [6] C. A. TOVEY, *Polynomial local improvement algorithms in combinatorial optimization*, Technical Report SOL 81-21, Stanford University, Stanford, CA, 1981.
- [7] K. W. HOKE, *Valid numberings of the vertices of the d-cube*, Ph.D. thesis, Univ. of North Carolina, Chapel Hill, NC, 1985, Appendix G.

ON SHORT NONCONTRACTIBLE CYCLES IN EMBEDDED GRAPHS*

JOAN P. HUTCHINSON†

Abstract. This paper contains a proof that every triangulation of an orientable surface of genus $g > 1$ with n vertices contains a noncontractible cycle of length $O(\sqrt{n/g} \log g)$. These bounds are tighter than those previously known, but are not the conjectured optimal. These results and techniques are related to "planarizing" problems in which a small set of vertices is sought whose removal leaves a planar graph and to "separator" problems in which a small set of vertices is sought whose removal leaves a graph with all components small.

Key words. genus, embedded graph, noncontractible cycle

AMS(MOS) subject classifications. primary 05C10; secondary 05C38

1. Introduction. Many results for graphs of positive genus have been derived from related results for planar graphs. Sometimes these results are possible since graphs embedded on surfaces act like planar graphs (e.g., see [1], [10]), and other times these results are derived by reducing nonplanar graphs to planar ones by deleting vertices and edges (see [7]). Thus two central problems in topological graph theory are finding a set of vertices and edges whose deletion reduces the genus of a graph and finding a set of vertices and edges whose deletion leaves a planar graph.

Regarding the first problem, it is known that in every triangulation of an orientable surface with n vertices there is a noncontractible cycle of length at most $\sqrt{2n}$ whose deletion decreases the genus by at least one [1]; this bound is not the best possible, which is not surprising because it is independent of the genus. On the other hand, a crucial step in the recent generalization of Kuratowski's theorem to nonplanar surfaces [10] involves conditions under which an embedded graph contains a constant-length noncontractible cycle.

CONJECTURE. *In a triangulation of a surface of genus $g > 0$ with n vertices, there is a noncontractible cycle of length $O(\sqrt{n/g})$.*

In this paper we provide evidence for this conjecture with examples achieving this upper bound, and we prove the following theorem.

THEOREM. *In a triangulation of an orientable surface of genus $g > 1$ with n vertices*
(a) *if $g \leq n$, there is a noncontractible cycle of length $O(\sqrt{n/g} \log g)$, and*
(b) *if $g > n$, there is a noncontractible cycle of length $O(\log g)$.*

Much of the work in this area has focused on the case of $g \leq n$ for the following reason. It has been conjectured and shown that there is always a "planarizing" set of $O(\sqrt{gn})$ vertices whose deletion leaves a planar graph [7], [4], [8]; in this context it is natural to assume $g \leq n$, for otherwise the deletion of all $n = O(\sqrt{gn})$ vertices clearly leaves a planar graph. The theorem of this paper implies a weaker solution to the planarizing problem in the case of $g \leq n$. A noncontractible cycle in an embedded graph may be separating and dividing the graph into two graphs of genus g_1 and g_2 where $g_1 + g_2 \leq g$ and $0 < g_i < g$, $i = 1, 2$. Otherwise a noncontractible cycle is nonseparating, and its deletion decreases the genus by at least one (see Fig. 1). Since deleting at most $g - 1$ of the former cycles and at most g of the latter produces a planar graph, the theorem of this paper shows that by deleting $O(\sqrt{gn} \log g)$ vertices a planar graph remains; this

* Received by the editors April 6, 1987; accepted for publication November 5, 1987.

† Department of Mathematics, Smith College, Northampton, Massachusetts 01063. This research was done in part while the author was visiting the Mathematical Sciences Research Institute, Berkeley, California 94720 and was supported in part by National Science Foundation grant DCR-8411690.

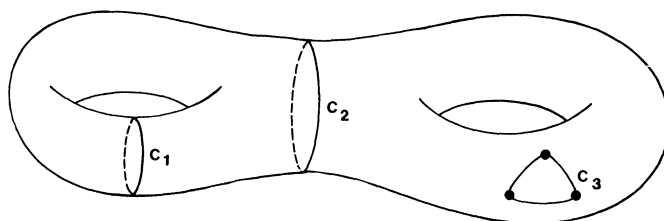


FIG. 1. C_1 noncontractible and nonseparating, C_2 noncontractible and separating, C_3 contractible.

result was proved earlier in [7]. By the same reasoning a proof of the conjecture would prove the $O(\sqrt{gn})$ planarizing result, but the ideas at hand seem not strong enough to yield the conjecture.

These results are also closely related to some “separator” theorems. These state that in a planar graph there is a set of $O(\sqrt{n})$ vertices and in a graph of genus $g > 0$ there is a set of $O(\sqrt{gn})$ vertices whose removal leaves all components with at most $2n/3$ vertices [5], [6], [7], [9]. Furthermore the “separator” and “planarizing” results all lead to efficient divide-and-conquer procedures for finding the specified sets of vertices in embedded graphs. The same algorithmic ideas can be used to find the cycle promised by the theorem of this paper.

2. Background and examples. We use the terminology of [3] and [12], but here are the main definitions. A graph is said to *embed* on a surface of genus $g \geq 0$ if it can be drawn on the sphere with g handles, denoted $S(g)$, so that no two edges cross. (In this paper we consider only orientable surfaces.) The *genus* of a graph G is the least integer g for which G embeds on $S(g)$. A *face* of an embedding of G on $S(g)$ is a connected component of $S(g) \setminus G$ and is called a *2-cell* if it is homeomorphic to the (open) interior of a disc. An embedding is called a *2-cell embedding* if every face is a 2-cell and a *triangulation* if every face is bounded by three edges. The crucial result for embedded graphs is Euler’s Formula.

THEOREM. *If G has a 2-cell embedding on $S(g)$, $g \geq 0$, then $n - e + f = 2 - 2g$ where n , e and f are, respectively, the number of vertices, edges, and faces of the embedded graph.*

In addition we use the following definition, which is stronger than that for a 2-cell. A region of a surface is said to be *contractible* if it is contained within the homeomorphic image of a simple closed disc (i.e., a disc with boundary such that no two points on the boundary are identified). A subgraph of an embedded graph is said to be *contractible* if it is contained within a contractible region. For example, in Fig. 3 the region to the right of the arc labeled L_{s-1} is a 2-cell, but not a contractible region.

Some familiar classes of graphs, when embedded, are known to have constant-length noncontractible cycles. For example, the complete graph on n vertices, the complete bipartite graph on n vertices and the k -cube with $n = 2^k$ vertices contain noncontractible 3-, 4- and 4-cycles, respectively, in every embedding. Their genus is, respectively, $O(n^2)$, $O(n^2)$ and $O(n \log n)$. However when the genus is at most n , some natural graphs of genus g have shortest noncontractible cycles of length $O(\sqrt{n/g})$. Two different such examples are described in [7] and [11]; we present another example, which is easily visualized and verified in the case $g = O(n)$.

A surface of genus g may be represented as a $4g$ -sided (planar) figure with $2g$ pairs of sides identified. In Fig. 2 we present a circle representation of a surface of genus 2.

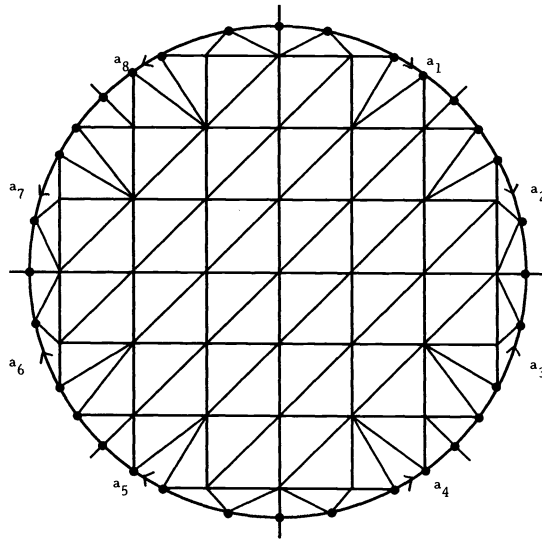


FIG. 2

More generally, given positive integers n and $g = O(n)$, we construct a circle, centered at the origin with radius $(2/\pi)\sqrt{gn}$ and divide the circumference into $4g$ pieces, each of length $\sqrt{n/g}$. If we label the arcs $a_1, a_2, \dots, a_{4g}$ and identify a_{4i+1} with a_{4i+3} and a_{4i+2} with a_{4i+4} for $i = 0, \dots, g-1$, then we have a representation of a surface of genus g .

To form an embedded graph we first choose $\lfloor \sqrt{n/g} \rfloor$ equally spaced vertices on each arc $a_1, \dots, a_{4g}$ plus one vertex at the end point of the arcs; we join successive pairs of these vertices with edges lying on the boundary of the circle. Inside the circle we choose as vertices all points with coordinates (x, y) where x and y are integer multiples of $m = (2/\pi)\sqrt{g}$. Using the standard lattice triangulation of the plane, the vertex at (im, jm) is adjacent to the vertices $((i-1)m, (j-1)m)$, $((i-1)m, jm)$, $(im, (j-1)m)$, $(im, (j+1)m)$, $((i+1)m, jm)$ and $((i+1)m, (j+1)m)$ that lie within the circle. See Fig. 2 for an example. Extend the graph to a triangulation by joining interior vertices to boundary vertices on the nearest arc a_i .

The shortest noncontractible cycles in the resulting graph are those on the boundary of the circle, of length $\lfloor \sqrt{n/g} \rfloor + 1$. Inside the circle is a grid of vertices, containing at most an $s \times s$ square of vertices where

$$s \leq 2 \left(\frac{2}{\pi} \sqrt{gn} / \left\lfloor \frac{2}{\pi} \sqrt{g} \right\rfloor \right) + 1 \leq 2\sqrt{n} + 1.$$

Thus there are at most $(2\sqrt{n} + 1)^2$ interior vertices and $\lfloor \sqrt{n/g} \rfloor 2g + 1$ boundary vertices for a total of at most

$$4n + 4\sqrt{n} + 1 + 2\sqrt{gn} + 1 = O(n)$$

vertices since $g = O(n)$.

3. The main result. The proof of the following theorem is related to, but significantly simpler than, the proof of the $O(\sqrt{gn})$ planarizing theorem of [4], [8]. Proofs of some technical lemmas can be found in [8].

THEOREM. *If G is an n -vertex graph embedded as a triangulation of a surface of genus $g > 1$, then*

- (a) *if $g \leq n$, G contains a noncontractible cycle of length $O(\sqrt{n/g} \log g)$, and*
- (b) *if $g > n$, G contains a noncontractible cycle of length $O(\log g)$.*

Proof. We assume that $g > 1$ so that $\log g$ is nonzero; it follows that $n > 7$ since otherwise the graph has genus zero or one. (Note, however, that for $g = 1$, by the results of [1] G contains an $O(\sqrt{n}) = O(\sqrt{n/g})$ noncontractible cycle.)

We begin by finding a breadth first spanning tree T with levels $L_0, L_1, \dots, L_r$ where L_i consists of all vertices at distance i from a designated root t and where r is the radius of T , as shown in Fig. 3. We define the *level* of an edge and of a triangle to be the maximum of the levels of the vertices of the edge and triangle, respectively. Let S_i denote the region of the surface consisting of all triangles of level at most i , and let G_i denote the subgraph of G of all vertices and edges in S_i . Finally let F_i be the subgraph of G_i formed from the modulo two sum of all edges of triangles in S_i . The following remarks contain simple, but relevant, properties of the cycles of F_i .

Remark 1. It follows from the definition that F_i is an edge-disjoint union of simple cycles. In addition an edge e is in F_i if and only if e joins two vertices of L_i and is incident with a triangle of level $i + 1$ and one of level i .

Remark 2. The vertices of F_i separate G in the sense that every path from a vertex on a level greater than i to the root t contains a vertex of F_i . Let P be such a path containing adjacent vertices y and z of levels i and $i + 1$, respectively; then y is in F_i . To see this, let x be a neighbor of y on level $i - 1$, and search counterclockwise through the neighbors of y from z to x to find a vertex w on level i that is also incident with triangles of levels $i + 1$ and i . There must be such a vertex w since G is a triangulation. Thus (y, w) is in F_i as is y .

Let P be a shortest path from a vertex on a level greater than i to t . Let x, y , and z be vertices of P on levels $i - 1, i$ and $i + 1$, respectively. Then a simple cycle C through y is said to *cross* P if in the cyclic permutation of neighbors of y the two neighbors of y on C are separated by x and z .

Remark 3. Suppose C is a contractible simple cycle of F_i that crosses P at y . C either does or does not contain t in its contractible interior. In the latter case, all vertices in its contractible interior must be on levels at least $i + 1$ by the Jordan Curve Theorem. Note also that C must contain in its contractible interior all vertices on levels at least $i + 1$ that are joined by a path of vertices, all on levels at least $i + 1$, to a vertex of P . On the other hand, if C does contain t in its contractible interior, then it contains in and on the boundary of the contractible interior all of G_i by the Jordan Curve Theorem. By the

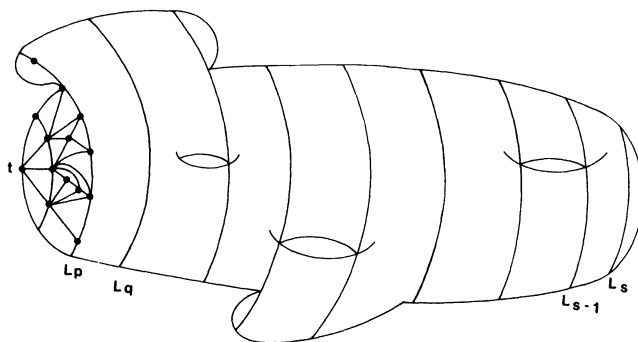


FIG. 3

same reasoning t cannot be contained in the contractible interior of more than one simple contractible cycle of F_i .

LEMMA 1. *Let P be a shortest path from a vertex on level at least $i + 1$ to t , and let $x_0 = P \cap L_i$. Suppose that every simple cycle of F_i through x_0 is contractible and does not contain t in its contractible interior. Then one such cycle of F_i must cross P .*

Proof. Let the neighbors of x_0 be labeled counterclockwise $x_1, \dots, x_j, \dots$ with $x_1 = P \cap L_{i+1}$ and $x_j = P \cap L_{i-1}$. Note that by Remark 2, x_0 and an incident edge lie in F_i . Let $e = (x_0, x_k)$ be such an edge with k as small as possible. Then the level of x_{k-1} is $i + 1$ and the level of x_{k+1} is $i - 1$ or i . Extend e to a simple cycle C of F_i . C is contractible and does not contain t in its contractible interior by hypothesis; we may assume C does not cross P . Among all such cycles choose C to be the one that returns to x_0 on the edge (x_0, x_h) with h as small as possible; by the choice of e , $h > k$. Since t is not inside C , all vertices there are on levels at least $i + 1$. But the level of x_{k+1} is at most i , and so x_{k+1} must lie on C . Furthermore, by the choice of (x_0, x_h) in C , $x_{k+1}, x_{k+2}, \dots, x_h$ are all on level i and so lie on C . Since (x_0, x_h) is in F_i and the level of x_{h-1} is i , the level of x_{h+1} must be $i + 1$. Then we can repeat the argument, replacing x_1 with x_{h+1} , finding another contractible cycle through x_0 . Eventually we must find a contractible cycle in F_i that crosses P or else reach a contradiction since the degree of x_0 is finite. $\square$

We return to the main argument of the proof. Suppose there is no noncontractible cycle in G of length at most $16\sqrt{n}/g$ if $g \leq n$ and of length at most 16 if $g > n$. Because these parameters arise so often we define

$$K = \begin{cases} 16\sqrt{n}/g & \text{if } g \leq n, \\ 16 & \text{if } g > n. \end{cases}$$

(The constant 16 and others below are chosen to yield near-optimal constants in the theorem; see further comments at the end of this proof.)

Let q be the largest integer such that G_q is contractible. Let s be the smallest integer such that $G \setminus G_s$ is contractible and all simple cycles of F_s are contractible. See Fig. 3. Intuitively the levels $L_{q+1}, \dots, L_{s-1}$ are the “nonplanar” part of the graph where we should search for noncontractible cycles. The next lemma should also be intuitively clear.

LEMMA 2. (a) *For $q < i$, t lies in the contractible interior of no simple contractible cycle of F_i .*

(b) *For $i \leq q$, there is a unique simple contractible cycle of F_i containing G_i within its contractible region.*

(c) *For $q < i < s$, not all simple cycles of F_i are contractible.*

(d) *Every vertex of $G \setminus G_s$ lies in the contractible interior of some simple contractible cycle of F_s .*

Proof. Part (a) follows from Remark 3 and the choice of q .

Parts (b) and (c). If $i = s - 1$ and $G \setminus G_{s-1}$ is contractible, then F_{s-1} contains a noncontractible cycle by the choice of s . Otherwise for all $i < s$, $G \setminus G_i$ contains a noncontractible cycle C^* since it contains either $G \setminus G_{s-1}$ or F_{s-1} . Let P be a shortest path from C^* to t , and let $x_0 = P \cap L_i$. Suppose the hypotheses of Lemma 1 are met so that a simple contractible cycle C that does not contain t crosses P . By Remark 3, C contains C^* in its contractible interior, a contradiction. We conclude that either some cycle C of F_i is noncontractible and thus $i > q$, or that some contractible cycle c_i contains t . In the latter case, G_i is also contained within c_i so that $i \leq q$ and the cycle c_i is unique.

Part (d). Let P be a path from a vertex x of $G \setminus G_s$ to t , and let $y = P \cap L_s$. From the definition of s and part (a), Lemma 1 applies and insures that a simple contractible

cycle of F_s must cross P at y . It does not contain t and so contains x in its contractible interior. $\square$

For $i \leq q$ we define c_i to be the unique simple cycle given in part (b).

Now let $p \leq q$ be the largest integer such that $|c_p| \leq K$. We focus our study on the following subgraph of G . First delete all vertices within the contractible interior of the cycle c_p . Next delete all vertices of $G \setminus G_s$, and call the remaining graph $G_{p,s}$. Since c_p is contractible and by Lemma 2(d) $G \setminus G_s$ lies in (the union of) contractible regions, $G_{p,s}$ has a 2-cell embedding in $S(g)$.

Now if the number of levels of $G_{p,s}$, $s - p + 1$, was $O(\sqrt{n/g})$, we would argue that $G_{p,s}$ and G contain an $O(\sqrt{n/g})$ noncontractible cycle (and hence if $g > n$ an $O(1)$ noncontractible cycle). Let $|H|$ denote the number of vertices in a subgraph H . Notice that $|L_i| > K$ for $p < i < s$ since $c_i \subseteq L_i$ when $p < i \leq q$ and by Lemma 2(c) when $q < i < s$. Thus we only know that $G_{p,s}$ contains at most $n/(16\sqrt{n/g}) = O(\sqrt{gn})$ (respectively, $n/16 = O(n)$) levels. We proceed by dividing up these levels (except for c_p) into b "bands" of radius $r' = \lceil \sqrt{n/g} \rceil$ where $b = \lceil (s - p)/r' \rceil$. See Fig. 4(a). For $i = 1, \dots, b - 1$ we let

$$B_i = L_{p+(i-1)r'+1} \cup \dots \cup L_{p+ir'}, \quad \text{and}$$

$$B_b = L_{p+(b-1)r'+1} \cup \dots \cup L_s.$$

(Thus when $g > n$, $r' = 1$ and $B_i = L_{p+i}$.) Let $|B_i| = n_i$ and for $i = 1, \dots, b$ let $L_{j(i)}$ be the level of smallest cardinality in B_i . Then $|L_{j(i)}| \leq n_i/r'$. For future reference we set $s' = \lceil 2\sqrt{n/g} \rceil$ and note that

$$(*) \quad \begin{aligned} n_i &> Kr' && \text{since all levels have size } > K \\ &> 5s'r' && \text{since } 5s' \leq 10\sqrt{n/g} + 5 < 16\sqrt{n/g} \text{ if } g \leq n (< 16 \text{ if } g > n). \end{aligned}$$

(For convenience of description in this paragraph, imagine a new root t^* adjacent to all vertices in the face bounded by c_p .) Consider $F_{j(i)} \subseteq L_{j(i)}$, and let H be a connected component of $F_{j(i)}$ such that $|H| \leq K$. Then H contains only contractible simple cycles, none of which contains t^* in its contractible interior. (If $j(i) \leq q$, then $c_{j(i)} \not\subseteq H$ and $c_{j(i)}$ is the unique simple contractible cycle of $F_{j(i)}$ that contains t^* . If $q < j(i)$, apply Lemma 2(a).) We delete all vertices in the interior of each contractible simple cycle of H and we redefine $F_{j(i)}$ to be $F_{j(i)}$ minus the vertices and edges of H (i.e., H becomes part of $L_{j(i)} \setminus F_{j(i)}$); we perform this deletion for each "small" component H of $F_{j(i)}$. The embedding of (the remaining) $G_{p,s}$ still is a 2-cell embedding, and every component of (the remaining) $F_{j(i)}$ has at least K vertices. In [8] we prove the following (actually Corollary 6 of Lemma 5).

LEMMA. *A graph with n vertices and each connected component having at least m vertices has a spanning forest of radius r with at most $\lfloor n/(r+1) \rfloor + n/m$ components.*

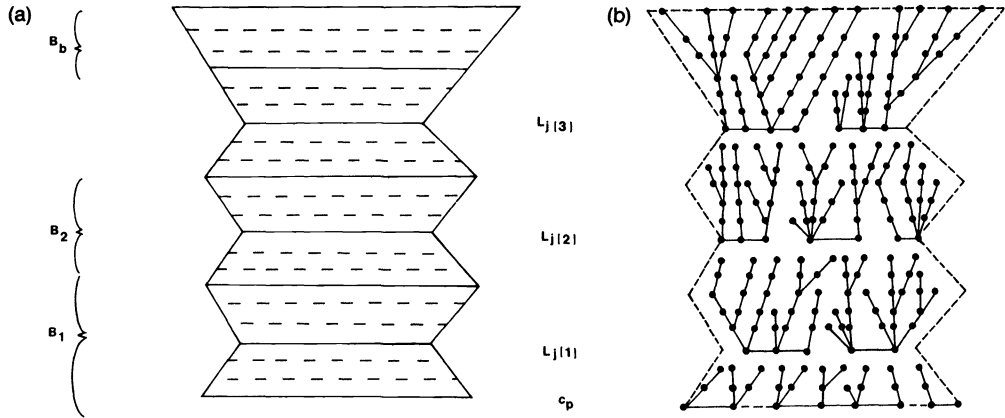
By the lemma the remaining parts of $F_{j(i)}$ can be covered by at most

$$|F_{j(i)}|/(s'+1) + |F_{j(i)}|/K$$

trees of radius at most s' . For $i = 2, \dots, b - 1$, let these parts be covered by trees $T_1, T_2, \dots, T_u$.

We also cover the bottom level of $G_{p,s}$, c_p , by at most four trees of radius s' (i.e., by paths of length $2s'$) since c_p is a simple cycle of length at most K ; call these trees $P_1, \dots, P_v$.

From these pieces we construct a spanning forest F of $G_{p,s}$, as shown in Fig. 4(b). We cover $c_p, F_{j(2)}, F_{j(3)}, \dots, F_{j(b-1)}$ with the trees $P_1, \dots, P_v, T_1, \dots, T_u$. Then we use the portion of the original tree T that extends from c_p up to and including vertices in $L_{j(2)} \setminus F_{j(2)}$ (but not including $F_{j(2)}$), for $i = 2$ to $b - 2$ from $F_{j(i)}$ up to and including

FIG. 4. (a) $G_{p,s}$. (b) A spanning forest of $G_{p,s}$.

$L_{j(i+1)} \setminus F_{j(i+1)}$, and from $F_{j(b-1)}$ up through L_s . See Fig. 4(b). F is a spanning forest of the remaining graph. (The only possible difficulty might be if a vertex x in $L_{j(i)+1}$ were adjacent in the tree T to a vertex y in $L_{j(i)} \setminus F_{j(i)}$, but applying Lemma 1 and Remark 3, x must be inside the contractible region of a contractible cycle of a “small” component of the original $F_{j(i)}$ and so x was deleted.) Each portion of T involves at most $2r'$ levels and so the resulting trees in F have radius at most $s' + 2r' \leq 4\sqrt{n/g} + 3 \leq 7\sqrt{n/g}$ if $g \leq n$ (< 7 if $g > n$).

Next we count the number of components of F . On levels $L_{j(2)}$ and up we have at most

$$\begin{aligned} \sum_{i=2}^{b-1} (|L_{j(i)}|/s' + |L_{j(i)}|/K) &\leq \sum_{i=2}^{b-1} (n_i/s'r' + n_i/r'K) \\ &\leq n/s'r' + n/r'K - n_1/s'r' - n_1/r'K \\ &< g/2 + g/16 - 6 \quad \text{from } (*). \end{aligned}$$

The cycle c_p is covered by four trees and so in total F contains at most $9g/16 - 2$ components.

Next we “prune” the embedded graph $G_{p,s}$ so that we can find and estimate the lengths of the noncontractible cycles in it. We repeatedly delete edges of $G_{p,s}$ that are not in the forest F and that are incident with two distinct faces in the embedding; it is not hard to see that when two faces are so merged into one, the new face remains a 2-cell (see [7]). If the remaining graph $G'_{p,s}$ has n' vertices and e' edges, then by Euler’s formula

$$n' - e' + 1 = 2 - 2g \quad \text{and} \quad e' = n' + 2g - 1.$$

Furthermore, in $G'_{p,s}$ every cycle is noncontractible since all contractible cycles have been destroyed.

Suppose there are exactly f components in the spanning forest F and thus $n' - f$ edges in the forest. We may assume that the remaining $2g + f - 1$ nonforest edges each join two different components in F , for if a nonforest edge joined two vertices in the same component, then it would form a noncontractible cycle of length at most $2(s' + 2r') + 1 \leq 8\sqrt{n/g} + 7 < K$. To estimate the length of the cycles in $G'_{p,s}$ we contract each forest component to a vertex, leaving a graph $G''_{p,s}$ with f vertices and $m = 2g +$

$f - 1$ edges. Since $f \leq 9g/16 - 2$, $2f \leq m = 2g + f - 1$, and we apply the result that such a graph contains a cycle of length $O(\log f)$ (see [2, p. 126]). More precisely in $G''_{p,s}$ there is a cycle of length at most

$$\begin{aligned} 2 \log f / \log \lfloor m/f \rfloor + 2 &\leq 2 \log f / \log ((2g-1)/(9g/16)) + 1 + 2 \\ &\leq 2 \log f / \log 3 + 2 < 1.27 \log f + 2 \\ &< 1.27 \log g + .95. \end{aligned}$$

Each contracted vertex was derived from a spanning tree in $G'_{p,s}$ of radius at most $s' + 2r'$ and so the noncontractible cycles in $G'_{p,s}$ are ones of length at most

$$\begin{aligned} (1.27 \log g + .95)(2(s' + 2r') + 1) &\leq (1.27 \log g + .95)(8\sqrt{n/g} + 7) \\ &= 10.16\sqrt{n/g} \log g + 7.6\sqrt{n/g} + 8.89 \log g + 6.65 \\ &\leq 33.3\sqrt{n/g} \log g \quad \text{when } 1 < g \leq n, \\ &33.3 \log g \quad \text{when } 1 < n < g. \quad \square \end{aligned}$$

The constants are neither best possible nor best possible for this derivation. The constants in K , r' , and s' can be varied, and a smaller bound can be achieved with $K = 17.88\sqrt{n/g}$, $r' = \lceil c\sqrt{n/g} \rceil$ and $s' = \lceil 2c\sqrt{n/g} \rceil$ where $c = 2^{(1/\ln 2) - 1}$. The resulting bound is $25.27\sqrt{n/g} \log g$ and $25.27 \log g$, but clearly these are more awkward coefficients with which to work.

In conclusion we must confess that this result is mainly of theoretical interest. Specifically, the bound of $\sqrt{2n}$ on noncontractible cycles of [1] is a better bound for all graphs of genus roughly less than 2^{17} .

Acknowledgments. The author thanks Gary Miller, Carsten Thomassen, and Stan Wagon for many helpful conversations on this material.

REFERENCES

- [1] M. O. ALBERTSON AND J. P. HUTCHINSON, *On the independence ratio of a graph*, J. Graph Theory, 2 (1978), pp. 1-8.
- [2] B. BOLLOBAS, *Extremal Graph Theory*, Academic Press, London, 1978.
- [3] J. A. BONDY AND U. S. R. MURTY, *Graph Theory with Applications*, American Elsevier, New York, 1976.
- [4] H. N. DJIDJEV, *Genus reduction in nonplanar graphs*, preprint.
- [5] ———, *On the problem of partitioning planar graphs*, SIAM J. Algebraic Discrete Methods, 3 (1982), pp. 229-240.
- [6] ———, *A separator theorem*, Comptes rendus de l'Académie bulgare des Sciences, 34 (1981), pp. 643-645.
- [7] J. R. GILBERT, J. P. HUTCHINSON, AND R. E. TARJAN, *A separator theorem for graphs of bounded genus*, J. Algorithms, 5 (1984), pp. 391-407.
- [8] J. P. HUTCHINSON AND G. L. MILLER, *On deleting vertices to make a graph of positive genus planar*, Discrete Algorithms and Complexity Theory, Proc. Japan-U.S. Joint Seminar, 1986, Kyoto, Perspectives in Computing, Vol. 15, Academic Press, Boston, 1987, pp. 81-98.
- [9] R. J. LIPTON AND R. E. TARJAN, *A separator theorem for planar graphs*, SIAM J. Appl. Math., 36 (1979), pp. 177-189.
- [10] N. ROBERTSON AND P. D. SEYMOUR, *Generalizing Kuratowski's theorem*, Congress. Numer., 45 (1984), pp. 129-138.
- [11] N. N. SIDER, *Partial colorings and limiting chromatic numbers*, Ph.D. thesis, Syracuse University, Syracuse, NY, 1971.
- [12] A. T. WHITE, *Graphs, Groups and Surfaces*, North-Holland, Amsterdam, 1973.

SELF-COMPLEMENTARY NORMAL BASES IN FINITE FIELDS*

ABRAHAM LEMPEL† AND MARCELO J. WEINBERGER†

Abstract. It is shown that $\text{GF}(q^n)$ has a self-complementary normal basis over $\text{GF}(q)$ if and only if n is odd or $n \equiv 2 \pmod{4}$ and q is even. All existence proofs are constructive and can be readily employed to obtain such bases.

Key words. finite fields, trace, basis, self-complementary basis, normal basis

AMS(MOS) subject classification. 12

1. Introduction. Let $F = \text{GF}(q)$ be a finite field of q elements and let $\Phi = \text{GF}(q^n)$ be an extension of degree n of F . For $\alpha \in \Phi$, the *trace* of α over F is defined by

$$\text{tr}(\alpha) = \sum_{i=0}^{n-1} \alpha^{q^i}.$$

A basis $\Gamma = \{\gamma_1, \gamma_2, \dots, \gamma_n\}$ of Φ over F is called *self-complementary* if

$$\text{tr}(\gamma_i \gamma_j) = \delta_{i,j} \triangleq \begin{cases} 1 & \text{if } i=j, \\ 0 & \text{otherwise.} \end{cases}$$

A basis Γ of Φ over F is called *normal* if $\Gamma = \{\gamma, \gamma^q, \gamma^{q^2}, \dots, \gamma^{q^{n-1}}\}$, $\gamma \in \Phi$.

In this paper we investigate the existence of bases of Φ over F which are both self-complementary and normal (in short, SCN bases). The main practical significance of SCN bases is in combining the proven potential of both self-complementary and normal bases as frameworks for fast multiplication in finite fields. MacWilliams and Sloane ([1, Chap. 4, Thm. 26]) have shown that Φ has an SCN basis over F if $q = 2$ and n is odd. Imamura [2] conjectured that Φ has an SCN basis over F if and only if n is odd or $n \equiv 2 \pmod{4}$ and q is even. The nonexistence of SCN bases when q is odd and n is even is a direct consequence of a result by Seroussi and Lempel [3], who proved this to be the only case in which there exists no self-complementary basis. Recently, Imamura and Morii [4] proved the nonexistence of SCN bases when q is even and $n \equiv 0 \pmod{4}$.

In this paper we prove in full the existence part of Imamura's conjecture. All proofs are constructive and can be readily employed to obtain an SCN basis of Φ over F whenever n is odd or $n \equiv 2 \pmod{4}$ and q is even. The required background material is presented in § 2. The main results and the associated constructions are presented in § 3 for odd n , and in § 4 for even n .

2. Background. In the sequel we shall make free use of some fundamental properties of finite fields (see [1, Chap. 4]; [5, Chap. 2]). In particular, we shall use the following properties of the trace operator:

$$\begin{aligned} \text{tr}(\alpha) &\in F, & \alpha &\in \Phi, \\ \text{tr}(\alpha^q) &= \text{tr}(\alpha), & \alpha &\in \Phi, \\ \text{tr}(a \cdot \alpha + b \cdot \beta) &= a \cdot \text{tr}(\alpha) + b \cdot \text{tr}(\beta), & \alpha, \beta &\in \Phi, \quad a, b \in F. \end{aligned}$$

* Received by the editors December 15, 1986; accepted for publication (in revised form) October 26, 1987.

† Department of Computer Science, Technion-Israel Institute of Technology, Haifa 32000, Israel.

For a matrix $M = (M_{ij})$, $M' = (M_{ji})$ denotes the transpose of M and $\text{tr}(M)$ means the matrix with $\text{tr}(M_{ij})$ as its (i, j) th element.

Let $\Gamma = \{\gamma_1, \gamma_2, \dots, \gamma_n\}$ be a basis of Φ over F and let $\underline{\gamma}$ denote the associated (row) vector $(\gamma_1 \gamma_2 \dots \gamma_n)$. Then, Γ is self-complementary if and only if $(\underline{\gamma}' \underline{\gamma}) = I_n$, the unit matrix of order n . In general, for every vector $\underline{\alpha}$ with entries from Φ , the elements of $\text{tr}(\underline{\alpha}' \underline{\alpha})$ are all in F . We also have the following.

LEMMA 1 ([5, Thm. 2.37]). *Given a vector $\underline{\alpha} = (\alpha_1 \alpha_2 \dots \alpha_n)$, $\alpha_i \in \Phi$, the matrix $\text{tr}(\underline{\alpha}' \underline{\alpha})$ is nonsingular if and only if $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ is a basis of Φ over F .*

Let $S = (S_{ij})$, $0 \leq i, j < n$, denote the $n \times n$ cyclic permutation matrix, where

$$S_{ij} = \delta_{i, j-1} = \begin{cases} 1 & \text{if } i \equiv j-1 \pmod{n}, \\ 0 & \text{otherwise.} \end{cases}$$

Clearly, $S^n = I_n$ and $S^{-1} = S^{n-1} = S'$. The polynomial matrix

$$C = \sum_{i=0}^{n-1} c_i S^i$$

is called a *circulant* matrix. It is easy to see that the first row of $C = (C_{ij})$ is given by $(c_0 c_1 \dots c_{n-1})$ and that

$$C_{ij} = C_{i+1, j+1} = c_{j-i}, \quad 0 \leq i, j < n,$$

where all indices are taken mod- n .

LEMMA 2 ([6, p. 124]). *The sum and the product of two circulant matrices, and the inverse of a circulant matrix, are circulant.*

From their polynomial form, it should also be clear that multiplication of circulant matrices is commutative.

LEMMA 3 ([5, Thm. 2.35]). *Every finite extension Φ of a finite field F has a normal basis over F .*

An element $\alpha \in \Phi$ such that the set $\{\alpha^{q^i}\}_{i=0}^{n-1}$ forms a basis of Φ over F will be referred to as a *normal basis generator* for Φ/F .

3. Self-complementary normal bases: odd n . Throughout this section $\Phi = \text{GF}(q^n)$ is a finite extension of odd degree n of the finite field $F = \text{GF}(q)$.

THEOREM 1. *Every odd extension $\Phi = \text{GF}(q^n)$ of any finite field $F = \text{GF}(q)$ has an SCN basis over F .*

Proof. Let $\alpha \in \Phi$ be a normal basis generator for Φ/F and let $\underline{\alpha} = (\alpha \alpha^q \dots \alpha^{q^{n-1}})$. Consider the circulant matrix $D = \sum_{k=0}^{n-1} \alpha^{q^k} S^k$. We have

$$\begin{aligned} A \triangleq DD' &= \sum_{k=0}^{n-1} \alpha^{q^k} S^k \sum_{j=0}^{n-1} \alpha^{q^j} S^{-j} \\ &= \sum_{k=0}^{n-1} \sum_{j=0}^{n-1} \alpha^{q^j(q^k-1)} S^{k-j} \\ &= \sum_{i=0}^{n-1} S^i \sum_{j=0}^{n-1} \alpha^{q^j(q^i+1)} \\ &= \text{tr}(\underline{\alpha}' \underline{\alpha}). \end{aligned}$$

We also have

$$\begin{aligned}
 R \triangleq D^2 &= \sum_{k=0}^{n-1} \sum_{j=0}^{n-1} \alpha^{q^k + q^j} S^{k+j} \\
 &= \sum_{i=0}^{n-1} S^i \sum_{j=0}^{n-1} \alpha^{q^{i-j} + q^j} \\
 &= \sum_{i=0}^{n-1} R_{0,i} S^i.
 \end{aligned}$$

With indices taken mod- n , we obtain

$$\begin{aligned}
 R_{0,i+2} &= \sum_{j=0}^{n-1} \alpha^{q^{i+2-j} + q^j} \\
 &= \sum_{j=0}^{n-1} \alpha^{q(q^{i-(j-1)} + q^{j-1})} \\
 &= \sum_{j=0}^{n-1} \alpha^{q(q^{i-j} + q^j)} = (R_{0,i})^q.
 \end{aligned}$$

Define $\omega \triangleq R_{0,0} = \sum_{j=0}^{n-1} \alpha^{q^{-j} + q^j}$, and $m = \frac{1}{2}(n+1)$. Then, we have

$$R_{0,i} = \omega^{q^{mi}}, \quad 0 \leq i < n,$$

and

$$\begin{aligned}
 RR' &= \sum_{i=0}^{n-1} \omega^{q^{mi}} S^i \sum_{k=0}^{n-1} \omega^{q^{mk}} S^{-k} \\
 &= \sum_{j=0}^{n-1} S^j \sum_{i=0}^{n-1} \omega^{q^{mi}(1+q^{-mj})}.
 \end{aligned}$$

Since m is relatively prime to n , we can write

$$RR' = \sum_{j=0}^{n-1} S^j \sum_{i=0}^{n-1} \omega^{q^{i(1+q^{-mj})}} = \text{tr}(\underline{\omega}' \underline{\omega}),$$

where $\underline{\omega} = (\omega \omega^{q^{-m}} \omega^{q^{-2m}} \dots \omega^{q^{-(n-1)m}})$. Since multiplication of circulant matrices is commutative, it follows that

$$A^2 = (DD')^2 = RR' = \text{tr}(\underline{\omega}' \underline{\omega}).$$

By Lemma 1, A is nonsingular. Since A is also a symmetric matrix with entries in F , we obtain

$$\text{tr}(A^{-1} \underline{\omega}' \underline{\omega} A^{-1}) = A^{-1} \text{tr}(\underline{\omega}' \underline{\omega}) A^{-1} = I_n.$$

By Lemma 1, again, the components γ_i , $0 \leq i < n$, of $\underline{\gamma} = \underline{\omega} A^{-1}$ form a basis Γ of Φ over F . Clearly, Γ is self-complementary. We claim that Γ is, actually, an SCN basis. To prove this it suffices to show that γ_0 is a normal basis generator for Φ/F . This, in turn, will follow if we show that for each $0 \leq j < n$, $\gamma_j = \gamma_0^{q^{km}}$ for some k . To this end, let $A^{-1} = B = (B_{ij})$. By Lemma 2, B is a circulant matrix so that $B_{i+1,j+1} = B_{i,j}$ for all i and j .

Thus, with indices taken mod- n , we have

$$\begin{aligned}
 \gamma_j &= \sum_{i=0}^{n-1} B_{ij} \omega^{q^{-im}} = \sum_{i=0}^{n-1} B_{i+1, j+1} \omega^{q^{-im}} \\
 &= \sum_{i=0}^{n-1} B_{i+1, j+1} \omega^{q^{-(i+1)m+m}} = \sum_{i=0}^{n-1} B_{i, j+1} \omega^{q^{-im+m}} \\
 &= \left(\sum_{i=0}^{n-1} B_{i, j+1} \omega^{q^{-im}} \right)^{q^m} \\
 &= \gamma_{j+1}^{q^m} = \gamma_{j+2}^{q^{2m}} = \cdots = \gamma_0^{q^{(n-j)m}}.
 \end{aligned}$$

This holds for each $0 \leq j < n$ and completes the proof. $\square$

The construction steps required to obtain an SCN basis of Φ over F , as suggested by the proof of Theorem 1, are summarized below.

(S1) Find a normal basis generator $\alpha \in \Phi$ for Φ/F .

(S2) Compute the matrix $A = \text{tr}(\underline{\alpha}' \underline{\alpha})$, where $\underline{\alpha} = (\alpha \alpha^q \cdots \alpha^{q^{n-1}})$.

(S3) Compute $B = A^{-1}$. Since A is a polynomial $a(S)$ in S , B can be readily obtained using Euclid's algorithm for the greatest common divisor of $a(x)$ and $x^n - 1$.

(S4) Compute $\gamma = \omega B$, where $\omega_i = \omega^{q^{-im}}$, $0 \leq i < n$, $m = \frac{1}{2}(n+1)$, and $\omega = \sum_{j=0}^{n-1} \alpha^{q^{-j}+q^j}$. $\Gamma = \{\gamma_i\}_{i=0}^{n-1}$ is an SCN basis of Φ over F .

4. Self-complementary normal bases: even n . Throughout this section, $\Phi = \text{GF}(q^n)$ is a finite extension of degree $n = 2m$, m odd, of the finite field $\text{GF}(q)$, q even.

THEOREM 2. *For every positive odd integer m and every $q = 2^k$, $k = 1, 2, \dots$, the field $\Phi = \text{GF}(q^{2m})$ has an SCN basis over $F = \text{GF}(q)$.*

The proof of Theorem 2 is presented in a series of lemmas, where we also deal with the fields $\Phi_1 = \text{GF}(q^m)$ and $F_2 = \text{GF}(q^2)$. The trace in a field X of an element α from an extension Y of X will be denoted by $\text{tr}(\alpha, Y/X)$. That is, if $X = \text{GF}(r)$ and $Y = \text{GF}(r^k)$ then

$$\text{tr}(\alpha, Y/X) = \sum_{i=0}^{k-1} \alpha^{r^i}.$$

LEMMA 4. $F_2 = \text{GF}(q^2)$ has an SCN basis over $F = \text{GF}(q)$, q even.

Proof. Let ω be any element from F_2 with $\text{tr}(\omega, F_2/F) = 1$. (It is well known that such an element always exists.) Since $\text{tr}(\omega, F_2/F) \neq 0$ implies $\omega \notin F$ it follows that ω is a normal basis generator for F_2/F . We have

$$\text{tr}(\omega^{2q}, F_2/F) = \text{tr}(\omega^2, F_2/F) = [\text{tr}(\omega, F_2/F)]^2 = 1$$

and, since $\omega^q \cdot \omega = \omega^{q+1} \in F$, we also have

$$\text{tr}(\omega^{q+1}, F_2/F) = 0.$$

Hence, $\{\omega, \omega^q\}$ is an SCN basis of F_2 over F . $\square$

LEMMA 5. Let $\Gamma = \{\gamma_1, \gamma_2, \dots, \gamma_m\}$ be a basis of $\Phi_1 = \text{GF}(q^m)$, m odd, over $F = \text{GF}(q)$. Then, Γ is also a basis of $\Phi = \text{GF}(q^{2m})$ over $F_2 = \text{GF}(q^2)$.

Proof. The matrix

$$M = \begin{bmatrix} \gamma_1 & \gamma_1^q \cdots \gamma_1^{q^{m-1}} \\ \gamma_2 & \gamma_2^q \cdots \gamma_2^{q^{m-1}} \\ \vdots & \vdots \\ \gamma_m & \gamma_m^q \cdots \gamma_m^{q^{m-1}} \end{bmatrix}$$

has a nonzero determinant (see [1, Chap. 4. Lemma 18], or [5, Cor. 2.38]). Therefore, the rows of M are linearly independent over F as well as over F_2 . Since m is odd, we can write

$$M_{ij} = \gamma_i^{q^j} = \gamma_i^{(q^2)^{k(j)}}, \quad 1 \leq i \leq m, \quad 0 \leq j < m,$$

where

$$k(j) = \begin{cases} \frac{1}{2}j & \text{if } j \text{ is even,} \\ \frac{1}{2}(j+m) & \text{if } j \text{ is odd.} \end{cases}$$

Hence, with $a_i \in F_2$, $1 \leq i \leq m$, we obtain for all $0 \leq j < m$,

$$\sum_{i=1}^m a_i \gamma_i^{q^j} = \sum_{i=1}^m a_i \gamma_i^{q^{2k(j)}} = \left[\sum_{i=1}^m a_i \gamma_i \right]^{q^{2k(j)}}.$$

This implies the independence of the elements of Γ over F_2 and, since Γ spans q^{2m} distinct elements when the coefficients are taken from F_2 , Γ is a basis of Φ over F_2 . $\square$

LEMMA 6. *Let $\{\alpha_1, \alpha_2, \dots, \alpha_m\}$ be a basis of Φ_1 over F and let $\{\omega_1, \omega_2\}$ be a basis of F_2 over F . Then, $\{\omega_i \alpha_j\}$, $i = 1, 2, j = 1, 2, \dots, m$ is a basis of Φ over F .*

Proof. Suppose $a_i, b_i \in F$, $1 \leq i \leq m$ are such that

$$0 = \sum_{i=1}^m (a_i \omega_1 \alpha_i + b_i \omega_2 \alpha_i) = \omega_1 \sum_{i=1}^m \alpha_i (a_i + b_i \omega_2 \omega_1^{-1}).$$

Then, since $(a_i + b_i \omega_2 \omega_1^{-1}) \in F_2$, $1 \leq i \leq m$, and since, by Lemma 5, the elements of $\{\alpha_1, \alpha_2, \dots, \alpha_m\}$ are independent over F_2 , we must have

$$a_i + b_i \omega_2 \omega_1^{-1} = 0, \quad 1 \leq i \leq m.$$

This, of course, implies $a_i = b_i = 0$ for all $1 \leq i \leq m$. $\square$

LEMMA 7. *Let α be a normal basis generator for Φ_1/F and let $\{\omega, \omega^q\}$ be a normal basis of F_2 over F . Then, $\omega \alpha$ is a normal basis generator for Φ/F .*

Proof. Since $\alpha^{q^m} = \alpha$, $\omega^{q^2} = \omega$, and m is odd, we have

$$\Gamma \triangleq \{(\omega \alpha)^{q^j}\}_{j=0}^{2m-1} = \{\omega \alpha^{q^j}\}_{j=0}^{m-1} \cup \{\omega^q \alpha^{q^j}\}_{j=0}^{m-1}.$$

Hence, by Lemma 6, Γ is a basis of Φ over F and, thus, it is a normal basis of Φ over F . $\square$

LEMMA 8. *Let $\{\alpha_1, \alpha_2, \dots, \alpha_m\}$ be a self-complementary basis of Φ_1 over F and let $\{\omega_1, \omega_2\}$ be a self-complementary basis of F_2 over F . Then, $\Gamma = \{\omega_i \alpha_j\}$, $i = 1, 2, j = 1, 2, \dots, m$ is a self-complementary basis of Φ over F .*

Proof. First, we claim that $\{\alpha_j\}$, $j = 1, 2, \dots, m$ is also a self-complementary basis of Φ over F_2 . Since, as in the proof of Lemma 5, for every $\beta \in \Phi_1$, we can write

$$\{\beta^{q^j}\}_{j=0}^{m-1} = \{\beta^{q^{2j}}\}_{j=0}^{m-1}.$$

Hence, for all $1 \leq i, j \leq m$,

$$\text{tr}(\alpha_i \alpha_j, \Phi/F_2) = \text{tr}(\alpha_i \alpha_j, \Phi_1/F) = \delta_{i,j} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

Now, it is easy to verify that for every $\gamma \in \Phi$,

$$\text{tr}(\gamma, \Phi/F) = \text{tr}(\gamma, \Phi/F_2) + [\text{tr}(\gamma, \Phi/F_2)]^q.$$

Therefore, for all $i, r \in \{1, 2\}$ and all $j, s \in \{1, 2, \dots, m\}$, we have

$$\begin{aligned}
 \text{tr}(\omega_i \alpha_j \cdot \omega_r \alpha_s, \Phi/F) &= \omega_i \omega_r \text{tr}(\alpha_j \alpha_s, \Phi/F_2) + [\omega_i \omega_r \text{tr}(\alpha_j \alpha_s, \Phi/F_2)]^q \\
 &= \omega_i \omega_r \delta_{j,s} + \omega_i^q \omega_r^q \delta_{j,s} \\
 &= \delta_{j,s} \text{tr}(\omega_i \omega_r, F_2/F) \\
 &= \delta_{j,s} \delta_{i,r}.
 \end{aligned}
 \quad \square$$

Proof of Theorem 2. By Theorem 1, there exists an SCN basis $\{\alpha_1, \alpha_2, \dots, \alpha_m\}$ of Φ_1 over F . By Lemma 4, F_2 has an SCN basis $\{\omega_1, \omega_2\}$ of F_2 over F . By Lemmas 7 and 8, $\Gamma = \{\omega_i \alpha_j\}$, $i = 1, 2, j = 1, 2, \dots, m$, is an SCN basis of Φ over F . $\square$

Remark. The proof of Lemma 5 can be readily extended to show that if Γ is a basis of $\text{GF}(q^k)$ over $\text{GF}(q)$ and t is relatively prime to k , then Γ is also a basis of $\text{GF}(q^{tk})$ over $\text{GF}(q^t)$. Consequently, the proofs of Lemmas 6–8 can be extended accordingly to obtain more general results. Lemma 4, however, cannot be extended in a similar fashion.

The construction steps suggested by the proof of Theorem 2 for the case where q is even and $n = 2m$, m odd, are as follows.

(1) Apply steps (S1)–(S4) of § 3 to obtain an SCN basis $\{\alpha_1, \alpha_2, \dots, \alpha_m\}$ of $\text{GF}(q^m)$ over $\text{GF}(q)$.

(2) Find an element $\omega_1 \in \text{GF}(q^2)$ with $\text{tr}(\omega_1, \text{GF}(q^2)/\text{GF}(q)) = 1$, and set $\omega_2 = \omega_1^q$.

(3) $\Gamma = \{\omega_i \alpha_j\}$, $i = 1, 2, j = 1, 2, \dots, m$ is an SCN of $\text{GF}(q^{2m})$ over $\text{GF}(q)$.

REFERENCES

- [1] F. J. MACWILLIAMS AND N. J. A. SLOANE, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam, 1977.
- [2] K. IMAMURA, *On self-complementary bases of $\text{GF}(q^n)$ over $\text{GF}(q)$* , Trans. IECE Japan (Section E) E66, 12 (1983), pp. 717–721.
- [3] G. SEROUSSI AND A. LEMPEL, *Factorization of symmetric matrices and trace-orthogonal bases in finite fields*, SIAM J. Comput., 9 (1980), pp. 758–767.
- [4] K. IMAMURA AND M. MORII, *Two classes of finite fields which have no self-complementary normal bases*, IEEE International Symposium on Information Theory, Brighton, England, June 1985.
- [5] R. LIDL AND H. NIEDERREITER, *Finite Fields*, Addison-Wesley, Reading, MA, 1983.
- [6] A. C. AITKEN, *Determinants and Matrices*, 9th ed., Oliver & Boyd, London, 1962.

ON THE OPTIMAL STRONGLY CONNECTED ORIENTATIONS OF CITY STREET GRAPHS I: LARGE GRIDS*

FRED S. ROBERTS† AND YONGHUA XU‡

Abstract. The problem of finding strongly connected orientations (one-way street assignments) for graphs which arise from city streets is studied. Specifically, the grid graphs consisting of $n_1 + 1$ east-west avenues and $n_2 + 1$ north-south streets, for n_1, n_2 sufficiently large, are studied. In general, it is difficult to find strongly connected orientations of graphs which are optimal according to any of a variety of criteria. However, for the grid graphs in question, optimal strongly connected orientations according to several important criteria are described. The results are surprising in that they improve significantly on the solution usually used in practice, namely: alternation of east and west orientations and north and south orientations.

Key words. graph theory, strongly connected, orientation, traffic, transportation, grids, optimization

AMS(MOS) subject classification. 05C99

1. Background. It has often been argued that making streets one-way would move traffic more efficiently. If all streets are to be made one-way, a one-way street assignment can be looked at as an *orientation* O of an undirected graph G , an assignment of a direction to each edge of G .¹ We seek an orientation O which is *strongly connected*, in the sense that in the digraph O , each vertex can reach each other vertex by a path. Robbins [4] gave criteria for a graph to have a strongly connected orientation, and Roberts [5], Boesch and Tindell [1], and Chung et al. [2] have given efficient algorithms for finding strongly connected orientations. However, as Roberts [5] points out, not every strongly connected orientation is very good. He also observes that there are various possible definitions of an efficient or an optimal strongly connected orientation. Chvátal and Thomasson [3] show that under a reasonable definition of optimality, the problem of finding the optimal strongly connected orientation of a graph is *NP-hard*. However, it is reasonable to restrict attention to the problem of finding the optimal strongly connected orientations of special graphs which arise in practice. In this paper, we consider this problem for the grid graphs or city street graphs which arise in cities with $n_1 + 1$ east-west avenues and $n_2 + 1$ north-south streets. We solve this problem under a reasonable definition of optimal for n_1, n_2 both sufficiently large. In subsequent papers [7]–[9], we solve it for n_1 or n_2 small. The results are surprising in that they improve significantly on the solution usually used in practice, namely: alternation of east and west orientations and north and south orientations.

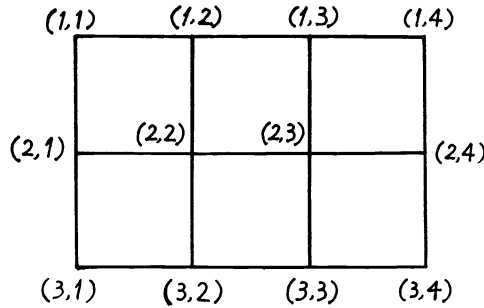
2. City street graphs and criteria for optimality. For $n_1 \geq 1, n_2 \geq 1$, let $G = G_{n_1 \times n_2} = (V, E)$ be the $n_1 \times n_2$ *grid graph* or *city street graph* defined as follows. G has vertices $v_{ij}, i = 1, 2, \dots, n_1 + 1, j = 1, 2, \dots, n_2 + 1$, and an edge $\{v_{pq}, v_{rs}\}$ if and only if $\{|p - r|, |q - s|\} = \{0, 1\}$. The graph $G_{n_1 \times n_2}$ can of course be drawn in the plane with vertex v_{ij} located at point $(j, n_1 + 2 - i)$, i.e., in the i th row and j th column of the grid. The edges are vertical or horizontal lines between neighboring grid points (i, j) . Figure 1 shows $G_{2 \times 3}$. Here we introduce some terminology about $G_{n_1 \times n_2}$ which will be useful later. All edges on the boundary of G when G is drawn in the plane are called

* Received by the editors May 12, 1986; accepted for publication (in revised form) June 1, 1987. This work was supported by the U.S. Air Force Office of Scientific Research under grant AFOSR-0271 to Rutgers University.

† Department of Mathematics and Center for Operations Research, Rutgers University, New Brunswick, New Jersey 08903.

‡ Taiyuan Heavy Machinery Institute, Taiyuan, Shanxi Province, People's Republic of China.

¹ Unless otherwise noted, we use the graph-theoretical notation and terminology of Roberts [5] or [6].

FIG. 1. $G_{2 \times 3}$.

boundary edges of G , all other edges are called *internal edges*. The four corner points are denoted by

$$w_1 = v_{11}, \quad w_2 = v_{1(n_2+1)}, \quad w_3 = v_{(n_1+1)1}, \quad w_4 = v_{(n_1+1)(n_2+1)}.$$

The grid is broken into four quadrants R_i , $i = 1, 2, 3, 4$, with R_i being the quadrant containing w_i . To be precise, suppose that

$$k_j = \left\lceil \frac{n_j + 1}{2} \right\rceil, \quad j = 1, 2,$$

and let k_j^* be k_j if n_j is odd and $k_j + 1$ if n_j is even. Then

$$R_1 = \{v_{ij} : 1 \leq i \leq k_1^*, 1 \leq j \leq k_2^*\},$$

$$R_2 = \{v_{ij} : 1 \leq i \leq k_1^*, k_2 + 1 \leq j \leq n_2 + 1\},$$

$$R_3 = \{v_{ij} : k_1 + 1 \leq i \leq n_1 + 1, 1 \leq j \leq k_2^*\},$$

$$R_4 = \{v_{ij} : k_1 + 1 \leq i \leq n_1 + 1, k_2 + 1 \leq j \leq n_2 + 1\}.$$

Note that if n_1 is even, then R_1 and R_3 overlap and R_2 and R_4 overlap; and if n_2 is even, then R_1 and R_2 overlap and R_3 and R_4 overlap. $G_{n_1 \times n_2}$, of course, always has a strongly connected orientation. Let $O_{n_1 \times n_2} = O$ be such an orientation. We shall also use the notation O for the digraph resulting from applying the orientation O to G .

Let $d(u, v)$ denote the length of the shortest chain in G from u to v , and let $\vec{d}(u, v)$ denote the length of the shortest (directed) path from u to v in O . (Path in this paper will always mean *directed path*.) A number of possible criteria for optimality of O are discussed in Roberts [5], [6]. We shall concentrate on minimizing one of the following functions:

$$D(O) = \max \{\vec{d}(u, v) : u, v \in V\},$$

$$L(O) = \sum_{u \in V} \max \{\vec{d}(u, x) : x \in V\},$$

$$M(O) = \sum_{u \in V} [\max \{\vec{d}(u, x) : x \in V\} - \max \{d(u, x) : x \in V\}].$$

Minimizing L is equivalent to minimizing M since $L(O) - M(O)$ is constant over all O . Among other possible functions to minimize are the following functions and their equivalents:

$$A(O) = \sum_{u, v \in V} \vec{d}(u, v),$$

$$N(O) = \max \{\vec{d}(u, v) - d(u, v) : u, v \in V\},$$

$$P(O) = \sum_{u, v \in V} [\vec{d}(u, v) - d(u, v)].$$

Assuming n_1 and n_2 are both sufficiently large, we shall find orientations O of $G_{n_1 \times n_2}$ which minimize both D and M (and hence L). The orientations improve significantly on the “alternating” solution commonly used, at least if M is taken into consideration. However, they are not as good under other criteria.

3. The fundamental lemmas. The parameters $D(O)$ and $M(O)$ are not so easy to compute. We begin by proving some lemmas which will allow us to give lower bounds on $M(O)$. The lemmas will be used frequently later. First, we introduce some notation:

$$\begin{aligned} d(u) &= \max \{d(u, x): x \in V\}, \\ \bar{d}_O(u) &= \bar{d}(u) = \max \{\bar{d}(u, x): x \in V\}, \\ m_O(u) &= m(u) = \bar{d}(u) - d(u). \end{aligned}$$

Then

$$M(O) = \sum_{u \in V} m(u).$$

Most of our early work will be aimed at obtaining bounds on $m(u)$.

If x maximizes $d(u, x)$, then x will be called a *value vertex* of u . Clearly, $x \in \{w_1, w_2, w_3, w_4\}$. In what follows, $f(u)$ will denote any value vertex of u . Also, in what follows, $p[u, v]$ will denote some path from u to v , and $\bar{p}[u, v]$ will denote a shortest path from u to v . If p is a path, $|p|$ will denote the length of the path, i.e., its number of arcs.

LEMMA 1. *If $p = p[u, w_i]$ and $d_{ui} = d(u, w_i)$, then $|p| - d_{ui}$ is even.*

Proof. The proof is by induction on $|p|$. If $|p| = 1$, then clearly $|p| = d_{ui}$. Arguing by induction, suppose the first step of p is to v . Let p' be p from v to w_i . Then $|p| - |p'| = 1$ and $|d_{ui} - d_{vi}| = 1$. $\square$

Note that if

$$(1) \quad |p[u, v]| = d(u, v),$$

then $p[u, v]$ is a shortest path from u to v . Any path $p[u, v]$ satisfying (1) is called a *limiting path* from u to v . If there is a limiting path from u to v , we shall write $LP[u, v, T]$. Otherwise, we write $LP[u, v, F]$. Note that if u has two incoming arcs pointing away from the value vertex $f(u)$, then $LP[u, f(u), F]$. (Pointing away from w_1 means pointing down or right, and similarly for other w_i .)

LEMMA 2. *Suppose that w_i is a value vertex of u . If $LP[u, w_i, F]$ (in particular if u has two incoming arcs pointed away from w_i), then $m(u) \geq 2$.*

Proof. For any $\bar{p} = \bar{p}[u, w_i]$, Lemma 1 and the hypothesis imply that

$$|\bar{p}| - d_{ui} \geq 2.$$

But $d_{ui} = d(u)$, so

$$m(u) = \bar{d}(u) - d(u) \geq |\bar{p}| - d(u) = |\bar{p}| - d_{ui} \geq 2. \quad \square$$

Vertex u_1 of Fig. 2 below illustrates Lemma 2. The value vertex is w_4 , and u_1 has two incoming arcs pointed away from w_4 .

If n_1 is odd (if n_2 is odd), v_{ij} is called a *horizontally (vertically) central vertex* if $i = k_1$ or $k_1 + 1$ (if $j = k_2$ or $k_2 + 1$). If u is a horizontally (vertically) central vertex and $w_i = f(u)$, then the corresponding *relevant vertex* $g(u)$ ($h(u)$) is w_s , the opposite corner point on the same vertical (horizontal) line as w_i . Thus, f , g , and h are given in Table 1. If u has two value vertices, it could of course have two relevant vertices.

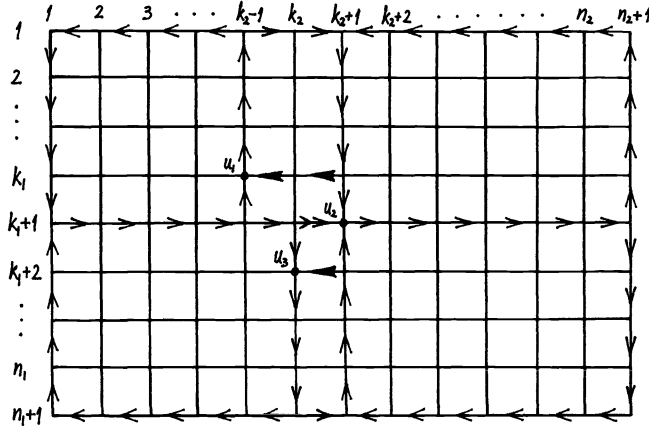


FIG. 2. O_{EE} of $G_{8 \times 12}$. All arcs shown are defining arcs except those shown with double arrows. Arcs shown with dark heads are defining internal arcs. All unoriented edges agree with their corresponding boundary edges. (See the definition of “corresponding” in § 4.) Vertices u_1, u_2, u_3 of Table 2 are shown.

LEMMA 3. Suppose that u is a central vertex and that w_t is a relevant vertex of u . If $LP[u, w_t, F]$ (in particular if u has two incoming arcs pointed away from w_t), then $m(u) \geq 1$.

Proof. As in the proof of Lemma 2, for any $\bar{p} = \bar{p}[u, w_t]$,

$$|\bar{p}| - d_{ut} \geq 2.$$

Moreover, since u is central and w_t is relevant to u , $d_{ut} = d(u) - 1$. Thus,

$$m(u) = \bar{d}(u) - d(u) \geq |\bar{p}| - d(u) = |\bar{p}| - d_{ut} - 1 \geq 1. \quad \square$$

Lemma 3 is illustrated by vertex u_7 of Fig. 3. This is a vertically central vertex; w_4 is one value vertex corresponding to u_7 , and w_3 is therefore a relevant vertex. By convention in Fig. 3, the one edge adjacent to u_7 with no orientation is oriented into u_7 . Hence, u_7 has two incoming arcs pointed away from w_3 . It follows that $m(u_7) \geq 1$.

LEMMA 4. Suppose that $u, v \notin R_t$ and that $p_1 = p_1[u, w_t]$ and $p_2 = p_2[v, w_t]$ are limiting paths to w_t from u and v , respectively. Suppose that $h_t = h_t(u, v)$ is the first common vertex of p_1 and p_2 going in the directions from u and from v . If h_t belongs to R_t , then $m(h_t) \geq 2$.

Proof. First, note that h_t exists since w_t belongs to both paths. Second, note that p_1 and p_2 enter h_t from different arcs. Third, since p_1 and p_2 are limiting paths, both these arcs must point toward w_t . Note that since $h_t \in R_t$, w_{5-t} is a value vertex of h_t . Finally, since h_t has two incoming arcs pointing away from w_{5-t} , Lemma 2 implies $m(h_t) \geq 2$. $\square$

LEMMA 5. Suppose that u and x are in the same quadrant R_t , that x is at least as close to the boundary of the grid as u on both dimensions, and that $LP[x, u, T]$. Then $m(x) \leq m(u)$.

TABLE 1

Value vertex	$f(u)$	w_1	w_2	w_3	w_4
Relevant vertex for u horizontally central	$g(u)$	w_3	w_4	w_1	w_2
Relevant vertex for u vertically central	$h(u)$	w_2	w_1	w_4	w_3

and

$$\{v_{rj_1}, v_{(r+1)j_1}\}, \{v_{rj_2}, v_{(r+1)j_2}\}, \quad r = i_1, i_1 + 1, \dots, i_2 - 1.$$

We now proceed to define orientations which will be seen to be optimal.

The orientation O_{EE} . Suppose that n_1, n_2 are even, $n_1 \geq 4, n_2 \geq 6$. We define the orientation O_{EE} of $G_{n_1 \times n_2}$ as follows. First, orient the circuits

$$[1(k_1 + 1); 1(k_2 - 1)] \quad \text{and} \quad [1(k_1 + 1); (k_2 + 1)(n_2 + 1)]$$

in a counterclockwise sense and orient the circuits

$$[(k_1 + 1)(n_1 + 1); 1k_2] \quad \text{and} \quad [(k_1 + 1)(n_1 + 1); (k_2 + 1)(n_2 + 1)]$$

in a clockwise sense. Second, include the following boundary arcs:

$$(v_{1(k_2 - 1)}, v_{1k_2}), \quad (v_{1k_2}, v_{1(k_2 + 1)}), \quad (v_{(n_1 + 1)k_2}, v_{(n_1 + 1)(k_2 + 1)}).$$

Third, include the following internal arcs:

$$(v_{k_1k_2}, v_{k_1(k_2 - 1)}), \quad (v_{k_1(k_2 + 1)}, v_{k_1k_2}), \quad (v_{(k_1 + 2)(k_2 + 1)}, v_{(k_1 + 2)k_2}).$$

Orient all remaining edges by the convention that internal edges agree with their corresponding boundary edges. The orientation O_{EE} for $G_{8 \times 12}$ is shown in Fig. 2.

The orientation O_{EO} . Suppose that n_1 is even, n_2 is odd, $n_1 \geq 4, n_2 \geq 5$. We define the orientation O_{EO} of $G_{n_1 \times n_2}$ as follows. First, orient the circuits

$$[1(k_1 + 1); 1k_2] \quad \text{and} \quad [1(k_1 + 1); (k_2 + 1)(n_2 + 1)]$$

in a counterclockwise sense and orient the circuits

$$[(k_1 + 1)(n_1 + 1); 1(k_2 - 1)] \quad \text{and} \quad [(k_1 + 1)(n_1 + 1); (k_2 + 1)(n_2 + 1)]$$

in a clockwise sense. Second, include the following boundary arcs:

$$(v_{1k_2}, v_{1(k_2 + 1)}), \quad (v_{(n_1 + 1)(k_2 - 1)}, v_{(n_1 + 1)k_2}), \quad (v_{(n_1 + 1)(k_2 + 1)}, v_{(n_1 + 1)k_2}).$$

Third, include the following internal arcs:

$$(v_{(k_1 + 1)k_2}, v_{(k_1 + 1)(k_2 + 1)}), \quad (v_{k_1(k_2 + 1)}, v_{k_1k_2}), \quad (v_{(k_1 + 2)k_2}, v_{(k_1 + 2)(k_2 - 1)}).$$

Orient all remaining edges by the convention that internal edges agree with their corresponding boundary edges. The orientation O_{EO} for $G_{8 \times 11}$ is shown in Fig. 3.

The orientation O_{OO} . Suppose that n_1, n_2 are odd, $n_1 \geq 5, n_2 \geq 7$. We define the orientation O_{OO} of $G_{n_1 \times n_2}$ as follows. First, orient the circuits

$$[k_1(k_1 + 1); 1(n_2 + 1)], \quad [1(k_1 - 1); (k_2 + 1)(n_2 + 1)], \quad \text{and} \quad [(k_1 + 2)(n_1 + 1); 1k_2]$$

in a counterclockwise sense and orient the circuits

$$[(k_1 - 1)k_1; 1(n_2 + 1)], [(k_1 + 1)(k_1 + 2); 1(n_2 + 1)],$$

$$[(k_1 + 1)(n_1 + 1); (k_2 + 1)(n_2 + 1)] \quad \text{and} \quad [1k_1; 1k_2]$$

in a clockwise sense. Second, orient the following boundary edges arbitrarily:

$$\{v_{1k_2}, v_{1(k_2 + 1)}\}, \quad \{v_{(n_1 + 1)k_2}, v_{(n_1 + 1)(k_2 + 1)}\}.$$

Third, include the following internal arcs:

$$(v_{k_1(k_2 + 2)}, v_{(k_1 - 1)(k_2 + 2)}), \quad (v_{(k_1 + 1)k_2}, v_{k_1k_2}), \\ (v_{k_1(k_2 + 1)}, v_{(k_1 + 1)(k_2 + 1)}), \quad (v_{(k_1 + 1)(k_2 - 1)}, v_{(k_1 + 2)(k_2 - 1)}).$$

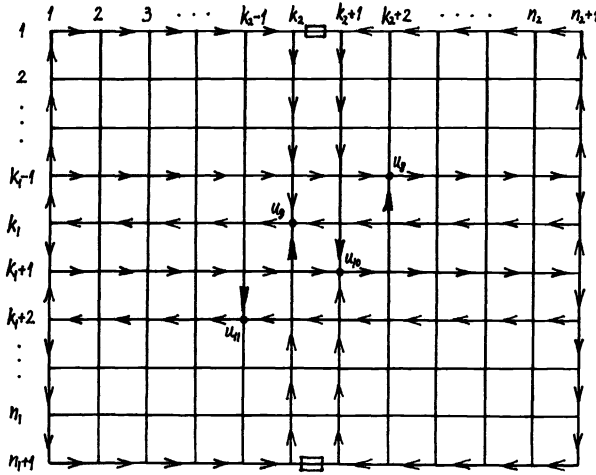


FIG. 4. O_{OO} of $G_{9 \times 11}$. All arcs shown are defining arcs. Arcs shown with dark heads are defining internal arcs. All unoriented edges agree with their corresponding boundary edges. (See the definition of "corresponding" in § 4.) Boundary edges with boxes ($\square$) are oriented arbitrarily. Vertices u_8, u_9, u_{10}, u_{11} of Table 2 are shown.

Orient all remaining edges by the convention that internal edges agree with their corresponding boundary edges. The orientation O_{OO} for $G_{9 \times 11}$ is shown in Fig. 4.

LEMMA 6. (a) O_{EE}, O_{EO} and O_{OO} are all strongly connected.

(b) $M(O_{EE}) = 6, M(O_{EO}) = 7, M(O_{OO}) = 8$.

(c) $D(O_{EE}) = D(O_{EO}) = D(O_{OO}) = n_1 + n_2$.

Proof. Let O be O_{EE}, O_{EO} or O_{OO} —whichever is appropriate. The lemma can be verified directly for $G_{4 \times 6}, G_{4 \times 5}$ and $G_{5 \times 7}$. Moreover, it can be verified that for $G_{4 \times 6}, G_{4 \times 5}$ and $G_{5 \times 7}$, Table 2 shows all vertices u with $m(u) > 0$ in O , and correctly shows all values $m(u)$. (The lower bounds on the m -values shown come directly from Lemmas 2 and 3.) We shall now show by induction that the three statements of the lemma hold and that Table 2 gives the correct vertices u with $m(u) > 0$ and the correct values $m(u)$.² We assume the lemma for $G_{n_1 \times n_2}$ and show it for $G_{n_1 \times (n_2+2)}$ and $G_{(n_1+2) \times n_2}$. We present the details of the argument only for $G_{n_1 \times (n_2+2)}$. Let O be O_{EE}, O_{EO} or O_{OO} , (whichever is appropriate) for $G_{n_1 \times (n_2+2)}$ and let O' be O_{EE}, O_{EO} or O_{OO} for $G_{n_1 \times n_2}$. Starting with O , delete the first and last columns. Then the resulting digraph is O' . By inductive assumption, O' is strongly connected. That O is strongly connected follows trivially (Figs. 2–4). Let $m'(u)$ denote the m -value in O' and $m(u)$ denote the m -value in O . Let d' and $\bar{d}'$ denote distances in $G' = G_{n_1 \times n_2}$ and O' , and let d and $\bar{d}$ denote distances in $G = G_{n_1 \times (n_2+2)}$ and O . We show that for all $u \in G'$, $m(u) \leq m'(u)$. In particular, we show that for all $u \in G'$,

$$(2) \quad \bar{d}(u) - d(u) \leq m'(u).$$

Let $\bar{d}(u) = \bar{d}(u, x)$ for x in G . Note that $d(u) = d'(u) + 1$. If $x \in G'$, then

$$\bar{d}(u) = \bar{d}(u, x) \leq \bar{d}'(u, x) \leq \bar{d}'(u)$$

and so

$$m(u) = \bar{d}(u) - d(u) \leq \bar{d}'(u) - (d'(u) + 1) \leq m'(u).$$

² The reader who wishes to skip the rest of the proof, which is not particularly interesting, can simply note that the key point is to verify the claim about Table 2. Parts (b) and (c) follow readily from this and part (a) is straightforward. Unfortunately, we do not know of either a simpler or a more interesting proof of the claim about Table 2.

TABLE 2

<i>EE case.</i> Vertices u with $m(u) > 0$	$u_1 = v_{k_1(k_2-1)}, \quad u_2 = v_{(k_1+1)(k_2+1)}, \quad u_3 = v_{(k_1+2)k_2}$			
$m(u)$	2	2	2	
<i>EO case.</i> Vertices u with $m(u) > 0$	$u_4 = v_{k_1k_2}, \quad u_5 = v_{(k_1+1)(k_2+1)}, \quad u_6 = v_{(k_1+2)(k_2-1)}, \quad u_7 = v_{(k_1+1)k_2}$			
$m(u)$	2	2	2	1
<i>OO case.</i> Vertices u with $m(u) > 0$	$u_8 = v_{(k_1-1)(k_2+2)}, \quad u_9 = v_{k_1k_2}, \quad u_{10} = v_{(k_1+1)(k_2+1)}, \quad u_{11} = v_{(k_1+2)(k_2-1)}$			
$m(u)$	2	2	2	2

If $x \in G - G'$, we shall show that

$$(3) \quad \bar{d}(u) \leq \bar{d}'(u) + 1,$$

and so

$$m(u) = \bar{d}(u) - d(u) \leq (\bar{d}'(u) + 1) - (d'(u) + 1) = m'(u).$$

To demonstrate (3), suppose first that boundary vertex x has an incoming arc from its neighboring internal vertex $\bar{x}$. Then by first following a shortest path in G' from u to $\bar{x}$, one sees that

$$(4) \quad \bar{d}(u, x) \leq \bar{d}'(u, \bar{x}) + 1.$$

Now $\bar{d}(u) = \bar{d}(u, x)$ and $\bar{d}'(u, \bar{x}) \leq \bar{d}'(u)$, so (4) implies (3). Thus, assume that $(x, \bar{x}) \in O$.

We consider several cases. First, consider O_{EE} and O_{EO} . Suppose that $x = v_{i(n_2+1)}$. Then $(x, \bar{x}) \in O$ implies that $i \neq k_1 + 1$. If $u = v_{pj}$, $j \neq n_2$, then in G and in G' the shortest paths from u to x and u to $\bar{x}$ must both go through $v_{(k_1+1)(k_2+1)}$. From this observation, (4) follows easily, and (4) implies (3). If $u = v_{pn_2}$, then $(x, \bar{x}) \in O$ implies $p \neq k_1 + 1$. Let $z = v_{p(n_2-1)}$. By what we have just proven,

$$(5) \quad \bar{d}(z, x) \leq \bar{d}'(z, \bar{x}) + 1.$$

By inductive assumption, $m'(z) = 0$ and so $\bar{d}'(z) = d'(z)$. Thus,

$$\begin{aligned}
 \bar{d}(u) &= \bar{d}(u, x) \\
 &\leq \bar{d}(z, x) + 1 \\
 &\leq \bar{d}'(z, \bar{x}) + 2 \\
 &\leq \bar{d}'(z) + 2 \\
 &= d'(z) + 2 \\
 &= (d'(u) - 1) + 2 \\
 &= d'(u) + 1 \\
 &= \bar{d}'(u) + 1,
 \end{aligned}$$

so (3) follows. Next, suppose that $x = v_{i1}$. Then $(x, \bar{x}) \in O$ implies that $i = k_1 + 1$. Suppose that $u = v_{pj}$. If $p \neq k_1 + 1$, then (4) follows immediately, and (4) implies (3). If $p = k_1 + 1$ and $j > 2$, then consider O_{EE} . In G and in G' , shortest paths from u to x and u to $\bar{x}$ must both go through $v_{k_1(k_2-1)}$ if $j \leq k_2 - 1$, or both go through $v_{(k_1+2)k_2}$ if $j = k_2$. If $j > k_2$, we can assume these paths both go through $v_{(k_1+2)k_2}$. In either case, (4) follows easily; hence, (3) holds. The argument is similar for O_{EO} . Finally, if $p = k_1 + 1$ and $j = 2$, let $z = v_{(k_1+1)3}$. Then by an argument similar to that above, (5) follows, and (3) follows from (5), as before. This concludes the argument for O_{EE} and O_{EO} .

Next, consider O_{OO} . By symmetry, we may assume that $x = v_{i(n_2+1)}$. Then since $(x, \bar{x}) \in O$, we have $i \neq k_1 - 1$, $i \neq k_1 + 1$. Suppose that $u = v_{pj}$. Suppose that $i < k_1 - 1$ and suppose first that $j \neq n_2$, $p \neq k_1 - 1$. Then in G and in G' the shortest paths from u to x and u to $\bar{x}$ must both pass through $v_{(k_1-1)(k_2+2)}$. From this observation (4) follows easily, and (4) implies (3). If $j \neq n_2$, $p = k_1 - 1$, then (4) follows directly; hence, (3) also holds. If $j = n_2$, let $z = v_{p(n_2-1)}$. Then in G and in G' , the shortest paths from z to x and z to $\bar{x}$ must both pass through $v_{(k_1-1)(k_2+2)}$. This implies (5). Then as above, (5) implies (3). We have concluded the argument if $i < k_1 - 1$. The argument if $i > k_1 + 1$ is identical, using $v_{(k_1+1)(k_2+1)}$. Finally, suppose that $i = k_1$. Then the argument is again the same, except we use *either* $v_{(k_1-1)(k_2+2)}$ *or* $v_{(k_1+1)(k_2+1)}$ and we do not have to test the case $j \neq n_2$, $p = k_1 - 1$ separately.

Thus, (2) has been established for all $u \in G'$. We next verify Table 2. By (2) and inductive hypothesis, $m(u) = 0$ for $u \neq u_i$, $u \in G'$. Moreover, $m(u_i) \leq m'(u_i) = 2$, $i \neq 7$. By Lemma 2, $m(u_i) \geq 2$ for $i \neq 7$. Thus, $m(u_i) = 2$, $i \neq 7$. Finally, $m(u_7) \leq m'(u_7) = 1$ and by Lemma 3, $m(u_7) \geq 1$, so $m(u_7) = 1$. To verify the rest of Table 2, it remains to show that $m(x) = 0$ for $x \in G - G'$. But this follows directly from the part of Table 2 we have already verified. Specifically, in O_{EE} we obtain the following results, using Lemma 5:

$$\begin{aligned} m(v_{i1}) &\leq m(v_{(k_1+1)k_2}) = 0 \quad \text{for all } i, \\ m(v_{i(n_2+1)}) &\leq m(v_{k_1(k_2+1)}) = 0, \quad i \leq k_1, \\ m(v_{i(n_2+1)}) &\leq m(v_{(k_1+2)(k_2+1)}) = 0, \quad i \geq k_1 + 2. \end{aligned}$$

Finally, we show that

$$(6) \quad m(v_{(k_1+1)(n_2+1)}) = 0.$$

Suppose that $x = v_{(k_1+1)(n_2+1)}$ and $\bar{x} = v_{(k_1+1)n_2}$. Suppose that $\bar{d}(x) = \bar{d}(x, y)$. Then clearly $y \neq v_{i(n_2+1)}$, for any i , since $\bar{d}(x) > \bar{d}(x, v_{i(n_2+1)})$, for all i . Moreover, we shall show that

$$(7) \quad \bar{d}(x, y) \leq \bar{d}(\bar{x}, y) + 1.$$

To see why, let p be a shortest path from $\bar{x}$ to y and let u be the vertex following $\bar{x}$ on this path. Note that since $y \neq v_{i(n_2+1)}$, $u = v_{rn_2}$, $r = k_1$ or $k_1 + 2$. Now let path p' go from x to $v_{r(n_2+1)}$ to v_{rn_2} and then follow path p . Now $|p'| = |p| + 1$. Next, note that

$$(8) \quad d(x) = d(\bar{x}) + 1.$$

Also, since $m(\bar{x}) = 0$,

$$(9) \quad \bar{d}(\bar{x}) = d(\bar{x}).$$

By (7), (8) and (9),

$$\bar{d}(x) = \bar{d}(x, y) \leq \bar{d}(\bar{x}, y) + 1 \leq \bar{d}(\bar{x}) + 1 = d(\bar{x}) + 1 = d(x),$$

so $m(x) = 0$ follows. This proves (6).

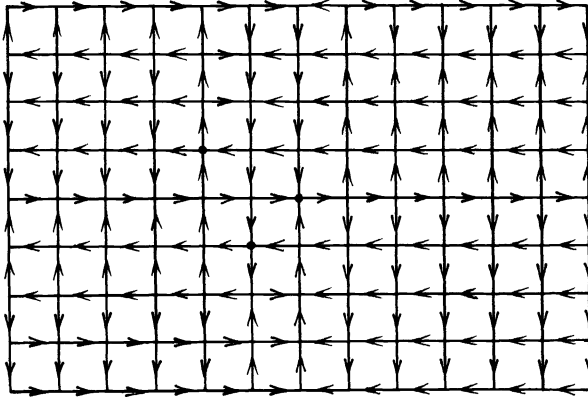


FIG. 5. Another orientation of $G_{8 \times 12}$ that satisfies the conditions of Lemma 6 is shown.

Similarly in O_{EO} , we use Lemma 5 to obtain

$$m(v_{i1}) \leq m(v_{(k_1+1)(k_2-1)}) = 0 \quad \text{for all } i,$$

$$m(v_{i(n_2+1)}) \leq m(v_{k_1(k_2+1)}) = 0, \quad i \leq k_1,$$

$$m(v_{i(n_2+1)}) \leq m(v_{(k_1+2)(k_2+1)}) = 0, \quad i \geq k_1 + 2.$$

Then, by an argument similar to that used in proving (6), we can show that

$$m(v_{(k_1+1)(n_2+1)}) = 0.$$

Finally, in O_{OO} , we use Lemma 5 to obtain

$$m(v_{i1}) \leq m(v_{(k_1-1)k_2}) = 0, \quad i \leq k_1 - 1,$$

$$m(v_{i1}) \leq m(v_{(k_1+1)k_2}) = 0, \quad i \geq k_1 + 1,$$

$$m(v_{i(n_2+1)}) \leq m(v_{k_1(k_2+1)}) = 0, \quad i \leq k_1,$$

$$m(v_{i(n_2+1)}) \leq m(v_{(k_1+2)(k_2+1)}) = 0, \quad i \geq k_1 + 2.$$

We then use an argument similar to that used in proving (6) to show that

$$m(v_{k_11}) = 0, \quad m(v_{(k_1+1)(n_2+1)}) = 0.$$

This completes the verification of Table 2 for G .

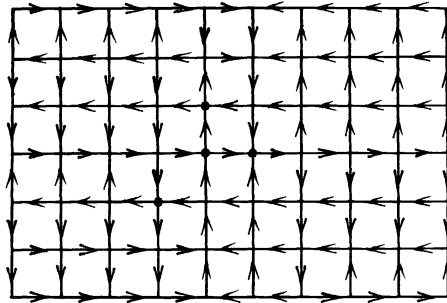


FIG. 6. Another orientation of $G_{6 \times 9}$ that satisfies the conditions of Lemma 6 is shown.

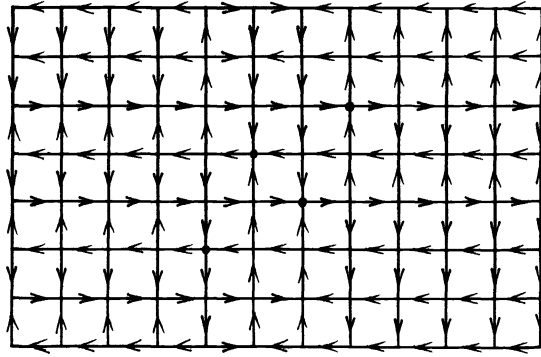


FIG. 7. Another orientation of $G_{7 \times 11}$ that satisfies the conditions of Lemma 6 is shown.

Table 2 now implies (b) of Lemma 6. Finally it is easy to derive (c) from Table 2. $\square$

We close this section by noting that the optimal orientations are not unique. There are other (nonisomorphic) orientations of $G_{n_1 \times n_2}$ which satisfy the conditions of Lemma 6. Figures 5–7 give examples of orientations O with the same $D(O)$ and $M(O)$ as those defined in this section.

5. The optimality of O_{EE} . In this section we shall study orientations of $G_{n_1 \times n_2}$, n_1, n_2 even, and we shall use the notation defined in Fig. 8.

LEMMA 7. Assume that $n_1 \geq 4$, $n_2 \geq 4$, n_1, n_2 even, and $(c_0, c_1) \in O$. Then either $M(O) \geq 6$ or there are $h_1, h_3 \neq c_0$, with $h_1 \neq h_3$, $h_t \in R_t$ and $m(h_t) \geq 2$, $t = 1, 3$. In the latter case $h_1 = h_1(s_i, y_j)$ or $h_1(z_i, y_j)$, for $(i, j) = (1, 1), (0, 1)$, or $(0, 2)$, and $h_3 = h_3(r_i, z_j)$ or $h_3(y_i, z_j)$, for $(i, j) = (1, 1), (0, 1)$, or $(0, 2)$, where $h_i(x, y)$ is defined in Lemma 4.

Proof. Assume that $m(c_1) \leq 1$. By Lemma 2 LP $[c_1, w_t, T]$, $t = 1, 3$. Thus $(c_1, y_1) \in O$ and $(c_1, z_1) \in O$. It follows that LP $[y_1, w_1, T]$ for the limiting path from c_1 to w_1 must go via y_1 . Similarly, LP $[z_1, w_3, T]$.

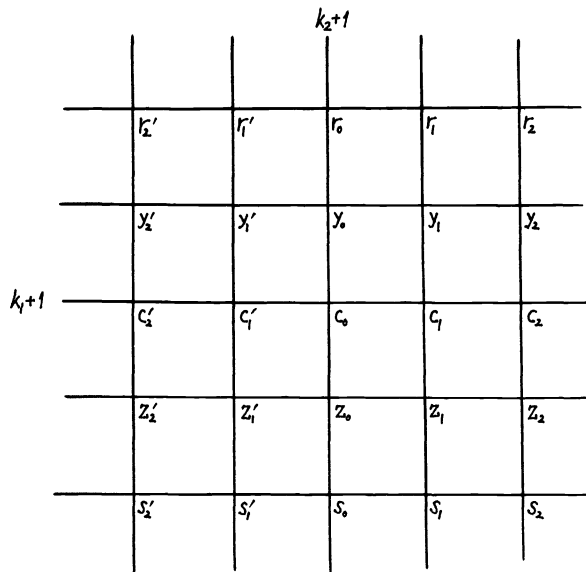


FIG. 8. Notation for $G_{n_1 \times n_2}$, n_1, n_2 even.

If LP $[r_1, w_3, T]$, let $h_3 = h_3(r_1, z_1)$ of Lemma 4. Note that $r_1, z_1 \notin R_3$. By location of r_1 and z_1 , h_3 must be in R_3 , and h_3 could not be c_0 or in R_1 . Hence, by Lemma 4, $m(h_3) \geq 2$. Similarly, LP $[y_1, w_3, T]$ implies there is $h_3(y_1, z_1) = h_3 \in R_3$, $h_3 \neq c_0$, $h_3 \notin R_1$, with $m(h_3) \geq 2$. If LP $[r_1, w_3, F]$ and LP $[y_1, w_3, F]$, Lemma 2 implies that $m(r_1) \geq 2$ and $m(y_1) \geq 2$. Similarly, by considering whether LP $[s_1, w_1, T]$ and whether LP $[z_1, w_1, T]$, we show that either there is $h_1 = h_1(s_1, y_1)$ or $h_1(z_1, y_1)$ with $h_1 \neq c_0$, $h_1 \notin R_3$, and $h_1 \in R_1$, with $m(h_1) \geq 2$, or $m(z_1) \geq 2$ and $m(s_1) \geq 2$. In summary, one of the following holds:

- (a) There are h_1, h_3 different from c_0 with $h_1 \in R_1 - R_3$, $h_3 \in R_3 - R_1$, and $m(h_i) \geq 2$, $t = 1, 3$.
- (b) There is $h_1 \in R_1$ with $m(h_1) \geq 2$, and also $m(r_1) \geq 2$ and $m(y_1) \geq 2$.
- (c) There is $h_3 \in R_3$ with $m(h_3) \geq 2$, and also $m(s_1) \geq 2$ and $m(z_1) \geq 2$.
- (d) $m(u) \geq 2$, $u = r_1, y_1, s_1, z_1$.

In case (a), the lemma follows. In cases (b), (c) and (d), $M(O) \geq 6$.

Next, assume that $m(c_1) \geq 2$. Now, either $m(c_2) \leq 1$ or $m(c_2) \geq 2$. In the former case, Lemma 2 implies that LP $[c_2, w_t, T]$, $t = 1, 3$. We can now argue that either LP $[y_1, w_1, T]$ or LP $[y_2, w_1, T]$, because the limiting path from c_2 to w_1 goes through y_1 or y_2 . Suppose LP $[y_j, w_1, T]$ where j is 1 or 2. Similarly, LP $[z_k, w_3, T]$, where k is 1 or 2. Then, as above, LP $[y_0, w_3, T]$ or LP $[r_0, w_3, T]$ implies that there is $h_3 \in R_3 - R_1$, $h_3 \neq c_0$, with $m(h_3) \geq 2$. Also LP $[y_0, w_3, F]$ or LP $[r_0, w_3, F]$ implies that $m(y_0) \geq 2$ and $m(r_0) \geq 2$. Similarly, either there is $h_1 \in R_1 - R_3$, $h_1 \neq c_0$, with $m(h_1) \geq 2$ or $m(z_0) \geq 2$ and $m(s_0) \geq 2$. We conclude that $M(O) \geq 6$. For $m(u) \geq 2$ for $u = h_1, h_3, c_1$, or for $u = h_3, z_0, s_0$, or for $u = h_1, y_0, r_0$, or for $u = z_0, s_0, r_0$.

It is left to argue in the case when $m(c_1) \geq 2$ and $m(c_2) \geq 2$. If $m(c'_q) \geq 2$ for some $q = 1, 2, \dots, k_2$, $M(O) \geq 6$ follows. Thus, suppose that $m(c'_q) \leq 1$ for all $q = 1, 2, \dots, k_2$. Assume first that for some $q = 1, 2, \dots, k_2$, $(c'_{q-1}, c'_q) \in O$, where $c'_0 = c_0$. Since $m(c'_q) \leq 1$, Lemma 2 implies that LP $[c'_q, w_4, T]$. If LP $[y_0, w_4, T]$, find $h_4 = h_4(c'_q, y_0)$ of Lemma 4. Clearly h_4 is in R_4 . Hence, by Lemma 4, $m(h_4) \geq 2$. Since $(c'_{q-1}, c'_q) \in O$, h_4 could not be c_1 or c_2 . Hence, $M(O) \geq 6$. If LP $[y_0, w_4, F]$, then by Lemma 2, $m(y_0) \geq 2$ and $M(O) \geq 6$ follows. It is left to consider the possibility that $(c'_q, c'_{q-1}) \in O$, $q = 1, 2, \dots, k_2$. (Again, $c'_0 = c_0$.) We may also assume that $m(c_0) \leq 1$, $m(y_0) \leq 1$, $m(z_0) \leq 1$, else $M(O) \geq 6$. Since $m(c_0) \leq 1$ and $(c'_1, c_0) \in O$, Lemma 2 implies $(c_0, y_0) \in O$ and $(c_0, z_0) \in O$. Also by Lemma 2 LP $[y_0, w_3, T]$ and LP $[z_0, w_1, T]$. Since the limiting paths p_1 and p_2 , respectively, both cross the line $i = k_1 + 1$, there are c'_{q_i} in p_i , $i = 1, 2$. Since $(c_0, y_0) \in O$ and $(c_0, z_0) \in O$, $q_i \geq 1$, $i = 1, 2$. Moreover, q_1 and q_2 are unique since $(c'_q, c'_{q-1}) \in O$, $q = 1, 2, \dots, k_2$. Moreover, $q_1 \neq q_2$ because the edge heading up from c'_{q_i} cannot be oriented in two directions. Without loss of generality let $q_1 > q_2$. Suppose that the vertex c'_{q_2} is $v_{(k_1+1)j}$. We show $m(v_{k_1j}) \geq 2$. For the path p_1 must still be going left when it hits the j th vertical line, so the arc $(v_{k_1(j+1)}, v_{k_1j})$ is in O . Moreover, the path p_2 must go up from $v_{(k_1+1)j}$ to v_{k_1j} , so $(v_{(k_1+1)j}, v_{k_1j}) \in O$. Thus by Lemma 2, $m(v_{k_1j}) \geq 2$. $M(O) \geq 6$ follows.

The last statement of the lemma follows by analysis of the proof. $\square$

THEOREM 1. Suppose that n_1, n_2 are even, $n_1 \geq 4$, $n_2 \geq 6$. Then O_{EE} minimizes $D(O)$, $L(O)$, and $M(O)$ over all strongly connected orientations O of $G_{n_1 \times n_2}$.

Proof. Clearly, $D(O) \geq d(w_1) = n_1 + n_2 = D(O_{EE})$, by Lemma 6. Moreover, as we have observed, $L(O)$ is minimized if and only if $M(O)$ is minimized. By Lemma 6, it suffices to show that for any O , $M(O) \geq 6$. By symmetry, we may assume that $(c_0, c_1) \in O$. By symmetry and strong connectedness, we may limit the possible orientations of edges incident to c_0 to those shown in Fig. 9. If case (1) or case (2) of Fig. 9 occurs, Lemma 2 implies that $m(c_0) \geq 2$. Then Lemma 7 implies that $M(O) \geq 6$. If case (3)

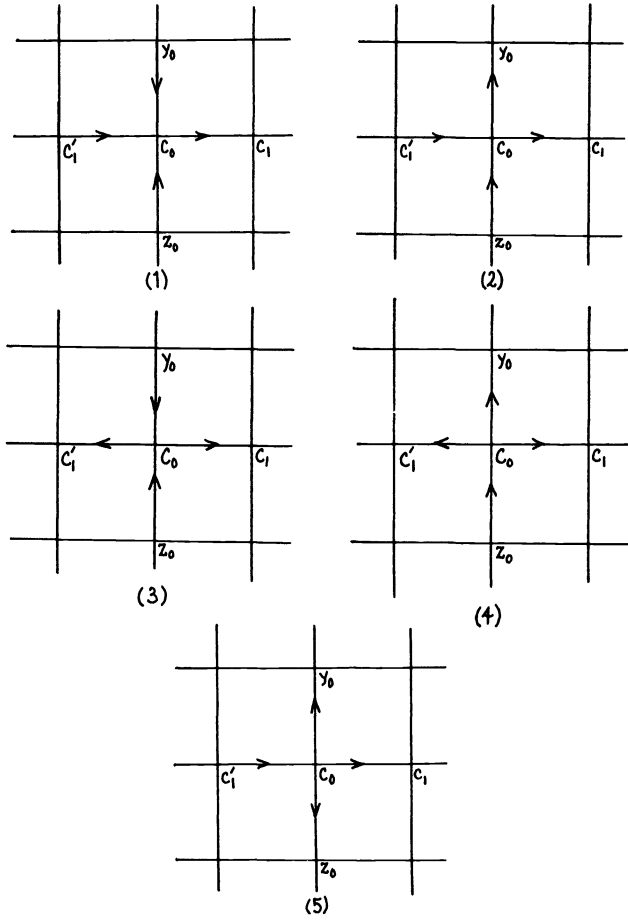


FIG. 9. The possible orientations of edges incident to c_0 in $G_{n_1 \times n_2}$, n_1, n_2 even.

occurs and $M(O) < 6$, Lemma 7 implies there are h_1, h_3 with $h_1 \neq h_3$, $h_i \in R_i$, $m(h_i) \geq 2$, $i = 1, 3$. Since $(z_0, c_0) \in O$ and $h_3 = h_3(r_i, z_j)$ or $h_3(y_i, z_j)$ for $(i, j) = (1, 1), (0, 1)$ or $(0, 2)$, and $h_3 \in R_3$, it follows that $h_3 \notin R_4$. Similarly, $(y_0, c_0) \in O$ implies that $h_1 \notin R_2$. Applying a symmetric version of Lemma 7 to the assumption $(c_0, c'_1) \in O$, we find that $M(O) < 6$ implies there are h_2, h_4 with $h_2 \neq h_4$, $h_i \in R_i$, $m(h_i) \geq 2$, $i = 2, 4$. Since $(y_0, c_0) \in O$ and $(z_0, c_0) \in O$, $h_2 \notin R_1$ and $h_4 \notin R_3$. Thus h_1, h_2, h_3, h_4 are all different, and $M(O) \geq 8$.

Finally, suppose case (4) or case (5) of Fig. 9 occurs. If $m(c_0) \geq 2$ or $m(c_1) \geq 2$, $M(O) \geq 6$ follows by Lemma 7. Suppose $m(c_0) \leq 1$ and $m(c_1) \leq 1$. By Lemma 2, $(c_1, y_1) \in O$ and $(c_1, z_1) \in O$. Now if $M(O) < 6$, pick h_1, h_3 as in Lemma 7. Consider case (5). Note that since $h_3 = h_3(r_i, z_j)$ or $h_3(y_i, z_j)$, $(i, j) = (1, 1), (0, 1)$, or $(0, 2)$, and since $(c_0, y_0) \in O$, and $(c_1, y_1) \in O$, h_3 could not be z_0 or z_1 . Moreover, h_1 is in R_1 and so could not be z_0 or z_1 . If $(z_1, z_0) \in O$, then Lemma 2 implies that $m(z_0) \geq 2$, so $M(O) \geq 6$. If $(z_0, z_1) \in O$, then Lemma 2 implies that $m(z_1) \geq 2$, so again $M(O) \geq 6$. Next consider case (4) of Fig. 9. As in case (3), find $h_2 \neq h_4$, $h_2 \in R_2$, $h_4 \in R_4$, $m(h_2) \geq 2$, $m(h_4) \geq 2$. Since $h_3 = h_3(r_i, z_j)$ or $h_3(y_i, z_j)$, $(i, j) = (1, 1), (0, 1)$, or $(0, 2)$, and since $(c_0, y_0) \in O$ and $(c_1, y_1) \in O$, h_3 is not in R_2 or R_4 . Thus, $h_3 \neq h_2, h_4$. We conclude that $M(O) \geq 6$. $\square$

6. The optimality of O_{00} . In this section we shall study orientations of $G_{n_1 \times n_2}$, n_1, n_2 odd, and we shall use the notation defined in Fig. 10.

LEMMA 8. Assume that $n_1 \geq 5$, $n_2 \geq 7$, n_1, n_2 odd. Then for every $t \in \{1, 2, 3, 4\}$, at least one of the following conclusions (1)_t–(7)_t must hold:

- (1)_t $m(c_t) \geq 2$.
- (2)_t $m(h_{5-t}) \geq 2$ for some $h_{5-t} \in R_{5-t}$, $h_{5-t} \neq c_{5-t}$.
- (3)_t $m(h_t) \geq 2$ for some $h_t \in R_t$, $h_t \neq c_t$.
- (4)_t If $t \in \{1, 4\}$, $m(c'_{(5-t)1}) \geq 2$; if $t \in \{2, 3\}$, $m(c''_{(5-t)1}) \geq 2$.
- (5)_t If $t \in \{1, 4\}$, $m(c''_{(5-t)1}) \geq 2$; if $t \in \{2, 3\}$, $m(c'_{(5-t)1}) \geq 2$.
- (6)_t If $t \in \{1, 4\}$, $m(c'_{q1}) \geq 1$ and $m(c'_{q2}) \geq 1$; if $t \in \{2, 3\}$, $m(c''_{q1}) \geq 1$ and $m(c''_{q2}) \geq 1$; where q is defined from t by the permutation (1342).
- (7)_t If $t \in \{1, 4\}$, $m(c'_{q1}) \geq 1$ and $m(c'_{q2}) \geq 1$; if $t \in \{2, 3\}$, $m(c'_{q1}) \geq 1$ and $m(c'_{q2}) \geq 1$; where q is defined from t by the permutation (1243).

Proof. We argue for $t = 1$. The other cases are symmetric. Suppose conclusion (1)₁ fails. Then $m(c_1) \leq 1$ and by Lemma 2, LP $[c_1, w_4, T]$. It follows that $(c_1, c_2) \in O$ or $(c_1, c_3) \in O$. We consider three cases:

- (i) $(c_1, c_2) \in O$, $(c_1, c_3) \in O$,
- (ii) $(c_1, c_2) \in O$, $(c_1, c_3) \notin O$,
- (iii) $(c_1, c_2) \notin O$, $(c_1, c_3) \in O$.

Case (i). Now either

- (a) LP $[c_2, w_4, T]$, or
- (b) LP $[c_3, w_4, T]$,

because the limiting path from c_1 to w_4 first goes to c_2 or to c_3 . We now consider subcases depending on which of (a) and (b) hold.

Subcase (iA): (a) and (b) both hold. Let $h_4 = h_4(c_2, c_3)$. Clearly, $h_4 \in R_4$ and so, by Lemma 4, $m(h_4) \geq 2$. If $h_4 \neq c_4$, conclusion (2)₁ follows. If $h_4 = c_4$, then

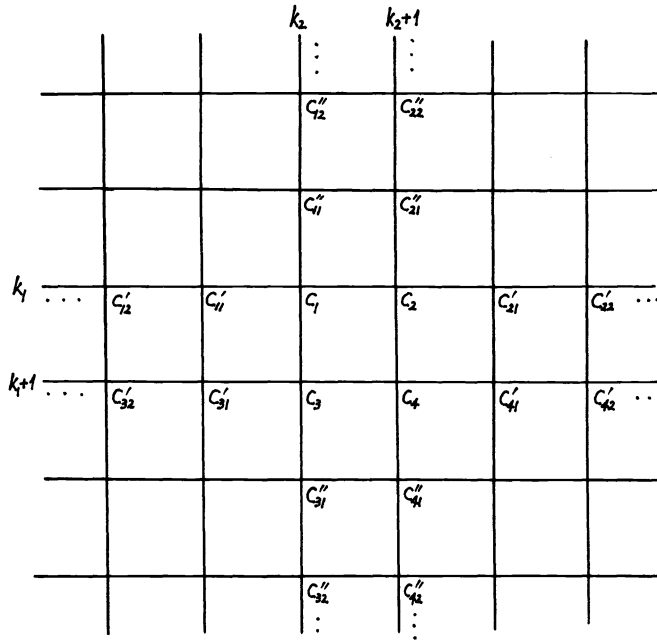


FIG. 10. Notation for $G_{n_1 \times n_2}$, n_1, n_2 odd.

$(c_3, c_4) \in O$ and $(c_2, c_4) \in O$. If LP $[c'_{41}, w_1, F]$, Lemma 2 implies conclusion $(4)_1$. If LP $[c''_{41}, w_1, F]$, Lemma 2 implies $(5)_1$. Thus, we can assume LP $[c'_{41}, w_1, T]$ and LP $[c''_{41}, w_1, T]$. Let $h_1 = h_1(c'_{41}, c''_{41})$. Since $(c_3, c_4) \in O$ and $(c_2, c_4) \in O$, $h_1 \in R_1$, so Lemma 4 implies $m(h_1) \geq 2$. Also, $(c_1, c_2) \in O$ and $(c_1, c_3) \in O$ imply $h_1 \neq c_1$. Thus conclusion $(3)_1$ holds.

Subcase (iB): (a) holds and (b) fails. If LP $[c'_{31}, w_4, T]$ or LP $[c'_{32}, w_4, T]$, let $h_4 = h_4(c_2, c'_{31})$ or $h_4(c_2, c'_{32})$. The vertices c_3 and c_4 cannot be on a limiting path from c'_{31} or c'_{32} to w_4 , since (b) fails. Hence $h_4 \in R_4$ and $h_4 \neq c_4$. By Lemma 4, $m(h_4) \geq 2$. Thus, conclusion $(2)_1$ holds. If LP $[c'_{31}, w_4, F]$ and LP $[c'_{32}, w_4, F]$, then by Lemma 3, $m(c'_{31}) \geq 1$ and $m(c'_{32}) \geq 1$, so conclusion $(6)_1$ follows.

Subcase (iC): (a) fails and (b) holds. A symmetric argument shows that LP $[c''_{21}, w_4, T]$ or LP $[c''_{22}, w_4, T]$ leads to conclusion $(2)_1$ and LP $[c''_{21}, w_4, F]$ and LP $[c''_{22}, w_4, F]$ leads to conclusion $(7)_1$.

Case (ii). Note that LP $[c_1, w_4, T]$ implies LP $[c_2, w_4, T]$. Suppose LP $[c'_{31}, w_4, T]$ or LP $[c'_{32}, w_4, T]$ and let h_4 be defined as in Subcase (iB). Now, clearly, $h_4 \in R_4$. Thus $m(h_4) \geq 2$ by Lemma 4. If $h_4 \neq c_4$, conclusion $(2)_1$ holds. If $h_4 = c_4$, then $(c_3, c_4) \in O$ and $(c_2, c_4) \in O$. Now, exactly as in the proof of (iA), LP $[c'_{41}, w_1, F]$ implies conclusion $(4)_1$, LP $[c''_{41}, w_1, F]$ implies $(5)_1$, and LP $[c'_{41}, w_1, T]$ and LP $[c''_{41}, w_1, T]$ together imply conclusion $(3)_1$. (In the latter argument, one cannot use $(c_1, c_3) \in O$ to prove $h_1 \neq c_1$. However, $(c_1, c_2) \in O$, $(c_2, c_4) \in O$ and $(c_3, c_4) \in O$ suffice to show that $h_1 \neq c_1$.) Finally, suppose LP $[c'_{31}, w_4, F]$ and LP $[c'_{32}, w_4, F]$. Then, as in the proof of Subcase (iB), conclusion $(6)_1$ follows.

Case (iii). This is symmetric. LP $[c_1, w_4, T]$ implies LP $[c_3, w_4, T]$. If LP $[c'_{21}, w_4, T]$ or LP $[c'_{22}, w_4, T]$, then let $h_4 = h_4(c_3, c'_{21})$ or $h_4(c_3, c'_{22})$. If $h_4 \neq c_4$, one concludes $(2)_1$ and if $h_4 = c_4$, one concludes $(4)_1$, $(5)_1$ or $(3)_1$. Finally, if LP $[c'_{21}, w_4, F]$ and LP $[c'_{22}, w_4, F]$, one concludes $(7)_1$. $\square$

LEMMA 9. For fixed t , the vertices in conclusions $(1)_t$ – $(7)_t$ of Lemma 8 are all different. Moreover, if vertex u appears in conclusion $(r)_t$ and vertex v in conclusion $(s)_t$, $t_1 \neq t_2$, then $u \neq v$ except in the following situations:

- (a) u is in $(2)_{t_1}$, v is in $(3)_{t_2}$, where $t_1 + t_2 = 5$;
- (b) u is in $(5)_{t_1}$, v is in $(6)_{t_2}$, where $(t_1, t_2) \in \{(1, 3), (3, 4), (4, 2), (2, 1)\}$;
- (c) u is in $(4)_{t_1}$, v is in $(7)_{t_2}$, where $(t_1, t_2) \in \{(1, 2), (2, 4), (4, 3), (3, 1)\}$.

Proof. The proof is straightforward.

LEMMA 10. Let (t, g) be any of the ordered pairs $(1, 2)$, $(2, 4)$, $(4, 3)$, or $(3, 1)$. Then:

- (a) If (c_t, c_g) , (c_g, c_{5-t}) , (c_t, c_{5-g}) , and (c_{5-g}, c_{5-t}) are in O , then one of conclusions $(1)_t$, $(3)_t$, $(4)_t$, $(5)_t$ of Lemma 8 is satisfied.
- (b) If (c_{5-g}, c_t) and (c_g, c_t) are in O , then $(1)_t$ is satisfied.
- (c) If (c_{5-t}, c_{5-g}) is in O or (c_{5-t}, c_g) is in O , then one of $(1)_t$, $(2)_t$, $(6)_t$, $(7)_t$ is satisfied.
- (d) If (c_{5-g}, c_t) is in O , then one of $(1)_t$, $(2)_t$, $(3)_t$, $(4)_t$, $(5)_t$, $(6)_t$ is satisfied.
- (e) If (c_{5-g}, c_t) and (c_{5-t}, c_{5-g}) are in O or if (c_{5-g}, c_t) and (c_{5-t}, c_g) are in O , then one of $(1)_t$, $(2)_t$, $(6)_t$ is satisfied.
- (f) If (c_g, c_t) is in O , then one of $(1)_t$, $(2)_t$, $(3)_t$, $(4)_t$, $(5)_t$, $(7)_t$ is satisfied.
- (g) If (c_g, c_t) and (c_{5-t}, c_g) are in O , or if (c_g, c_t) and (c_{5-t}, c_{5-g}) are in O , then one of $(1)_t$, $(2)_t$, $(7)_t$ is satisfied.

Proof. We assume $t = 1$ and hence $g = 2$. All other cases are similar.

(a) Suppose that $(1)_1$ fails. If LP $[c'_{41}, w_1, T]$ and LP $[c''_{41}, w_1, T]$, let $h_1 = h_1(c'_{41}, c''_{41})$. Now h_1 must be in R_1 and could not be c_1 , by the assumptions of (a). Hence, by Lemma 4, $(3)_1$ holds. If LP $[c'_{41}, w_1, F]$, then $m(c'_{41}) \geq 2$ by Lemma 2, and so $(4)_1$ holds. Similarly, if LP $[c''_{41}, w_1, F]$, then $(5)_1$ holds.

(b) Here, Lemma 2 implies $(1)_1$.

(c) Suppose that $(c_4, c_3) \in O$ or $(c_4, c_2) \in O$. Suppose that $(1)_1$ fails. We follow the argument in the proof of Lemma 8. In Subcase (iA) of the proof, $h_4 \neq c_4$ must hold. Hence $(2)_1$ holds. In Subcase (iB), $(2)_1$ or $(6)_1$ holds. In Subcase (iC), $(2)_1$ or $(7)_1$ holds. In Case (ii) in the first part of the argument, $h_4 \neq c_4$ must hold, so $(2)_1$ follows. In the second part $(6)_1$ follows. In Case (iii), in the first part of the argument, $h_4 \neq c_4$ must hold, so $(2)_1$ follows. In the second part of the argument, $(7)_1$ follows.

(d) The only places in the proof of Lemma 8 in which conclusion $(7)_1$ arises are in Subcase (iC) and Case (iii). But in Subcase (iC) and Case (iii), it is assumed that $(c_1, c_3) \in O$, which contradicts our hypothesis $(c_3, c_1) \in O$. Thus, $(1)_1$, $(2)_1$, $(3)_1$, $(4)_1$, $(5)_1$ or $(6)_1$.

(e) By parts (c) and (d).

(f) The only places in the proof of Lemma 8 in which conclusion $(6)_1$ arises are in Subcase (iB) and Case (ii). But in Subcase (iB) and Case (ii), it is assumed that $(c_1, c_2) \in O$, which contradicts our hypothesis $(c_2, c_1) \in O$.

(g) By parts (c) and (f). $\square$

THEOREM 2. Suppose that n_1, n_2 are odd, $n_1 \geq 5, n_2 \geq 5$. Then O_{OO} minimizes $D(O), L(O)$, and $M(O)$ over all strongly connected orientations O of $G_{n_1 \times n_2}$.

Proof. The proof for $D(O)$ is as in Theorem 1. Also, as in the proof of Theorem 1, it suffices, by Lemma 6, to show that for any $O, M(O) \geq 8$. By symmetry, we may assume that O restricted to $\{c_1, c_2, c_3, c_4\}$ is one of the digraphs of Fig. 11.

By Lemma 8, for every $t \in \{1, 2, 3, 4\}$, one of conclusions $(1)_t$ – $(7)_t$ must hold. Using Lemma 10, we can determine in each of cases I–IV of Fig. 11 a set of conclusions $(i)_t$ which is guaranteed to hold. The results are shown in Table 3, which also indicates which part of Lemma 10 justifies the conclusion.

From Table 3, we know that in case I, $(1)_1, (2)_1$ or $(6)_1$ holds, and also $(1)_2, (2)_2$ or $(6)_2$, and also $(1)_3, (2)_3$ or $(6)_3$ and also $(1)_4, (2)_4$ or $(6)_4$. By Lemma 9, $M(O) \geq 8$.

The argument in cases III and IV is similar. In case II, the same argument gives us $M(O) \geq 8$ unless $(5)_3$ holds when $t = 3$ and $(6)_4$ holds when $t = 4$. In this situation, $m(c'_{21}) \geq 2$ and $m(c'_{21}) \geq 1$, and $m(c'_{22}) \geq 1$, and we can only conclude $M(O) \geq 7$. If $LP[c'_{21}, w_3, F]$, then $m(c''_{21}) \geq 2$. Since $m(c'_{21}) \geq 2$ and $m(c'_{21}) \geq 2$ and $(1)_1, (2)_1$ or

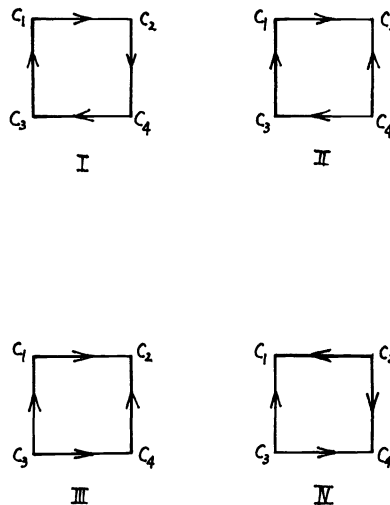


FIG. 11. The possible orientations of $G_{n_1 \times n_2}$, n_1, n_2 odd, when restricted to $\{c_1, c_2, c_3, c_4\}$.

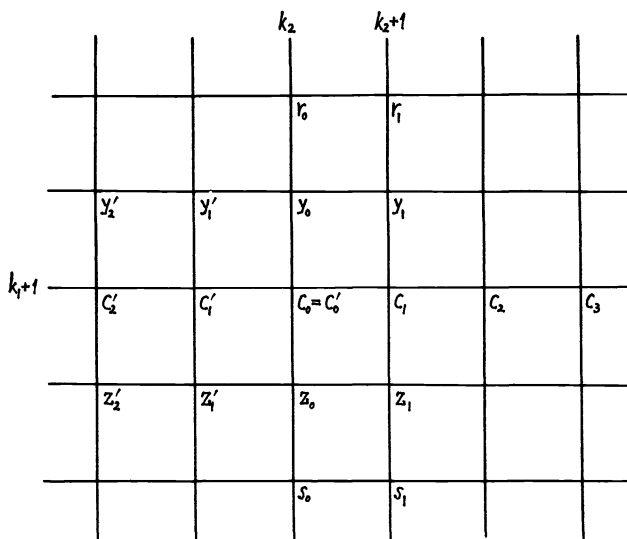
TABLE 3

Conclusions from Lemma 8 which are guaranteed to hold in cases I–IV of Fig. 11, with justification from Lemma 10 shown.

Case	$t = 1$	$t = 2$	$t = 3$	$t = 4$
I	$(1)_1, (2)_1$ or $(6)_1$ by (e) of Lemma 10	$(1)_2, (2)_2$ or $(6)_2$ by (e) of Lemma 10	$(1)_3, (2)_3$ or $(6)_3$ by (e) of Lemma 10	$(1)_4, (2)_4$ or $(6)_4$ by (e) of Lemma 10
II	$(1)_1, (2)_1$ or $(6)_1$ by (e) of Lemma 10	$(1)_2$ by (b) of Lemma 10	$(1)_3, (2)_3, (3)_3,$ $(4)_3, (5)_3$ or $(6)_3$ by (d) of Lemma 10	$(1)_4, (2)_4, (6)_4$ or $(7)_4$ by (c) of Lemma 10
III	$(1)_1, (2)_1$ or $(6)_1$ by (e) of Lemma 10	$(1)_2$ by (b) of Lemma 10	$(1)_3, (3)_3, (4)_3$ or $(5)_3$ by (a) of Lemma 10	$(1)_4, (2)_4$ or $(7)_4$ by (g) of Lemma 10
IV	$(1)_1$ by (b) of Lemma 10	$(1)_2, (2)_2, (6)_2$ or $(7)_2$ by (c) of Lemma 10	$(1)_3, (2)_3, (6)_3$ or $(7)_3$ by (c) of Lemma 10	$(1)_4$ by (b) of Lemma 10

$(6)_1$ and $(1)_2$, $M(O) \geq 8$, because c'_{21} and c'_{21} do not appear in $(1)_1, (2)_1, (6)_1$ or $(1)_2$. Similarly, if $\text{LP}[c'_{22}, w_3, F]$, then $m(c'_{22}) \geq 2$ and $M(O) \geq 8$. If $\text{LP}[c'_{21}, w_3, T]$ and $\text{LP}[c'_{22}, w_3, T]$, let $h_3 = h_3(c'_{21}, c'_{22})$. Since we are in case II, $h_3 \in R_3 - R_4$. By Lemma 4, $m(h_3) \geq 2$. Now $m(c'_{21}) \geq 2$ and $m(h_3) \geq 2$ and, by $(1)_2$, $m(c_2) \geq 2$. If $(1)_1$ or $(2)_1$ hold, then $M(O) \geq 8$ follows since $h_3 \neq h_4$. If $(6)_1$ holds, then either $h_3 \neq c'_{31}$ or $h_3 \neq c'_{32}$, say the former. Now $m(c'_{31}) \geq 1$ and also $m(c'_{22}) \geq 1$. This gives us $M(O) \geq 8$. $\square$

7. The optimality of O_{EO} . In this section, we shall study orientations of $G_{n_1 \times n_2}$, n_1 even, n_2 odd, and we shall use the notation defined in Fig. 12.

FIG. 12. Notation for $G_{n_1 \times n_2}$, n_1 even, n_2 odd.

LEMMA 11. Assume that $n_1 \geq 4$, $n_2 \geq 5$, n_1 even, n_2 odd. Suppose that $q \in \{0, 1, \dots, k_2 - 1\}$ and $(c'_q, c'_{q+1}) \in O$. Then either $m(c'_{q+1}) \geq 2$ or in $R_2 - R_4 \cup R_4 - R_2$ there exist at least two vertices whose total m -value is at least 4.

Proof. Suppose that $m(c'_{q+1}) \leq 1$. By Lemma 2, $LP[c'_{q+1}, w_j, T]$ for $j = 2, 4$. If $LP[y_1, w_4, F]$ and $LP[r_1, w_4, F]$, then Lemma 3 implies that $m(y_1) \geq 1$ and $m(r_1) \geq 1$. If $LP[y_1, w_4, T]$ or $LP[r_1, w_4, T]$, then let $h_4 = h_4(y_1, c'_{q+1})$ or $h_4(r_1, c'_{q+1})$. Note that h_4 is in R_4 and not in R_2 , since $(c'_q, c'_{q+1}) \in O$. By Lemma 4, $m(h_4) \geq 2$. Similarly, $LP[z_1, w_2, F]$ and $LP[s_1, w_2, F]$ imply that $m(z_1) \geq 1$ and $m(s_1) \geq 1$, while $LP[z_1, w_2, T]$ or $LP[s_1, w_2, T]$ implies that there is h_2 in R_2 , not in R_4 , with $m(h_2) \geq 2$. Note that $h_2 \neq h_4$ because $h_4 \notin R_2$. The lemma follows by using y_1, r_1, z_1, s_1 or y_1, r_1, h_2 or z_1, s_1, h_4 or h_2, h_4 . $\square$

THEOREM 3. Suppose that $n_1 \geq 4$, $n_2 \geq 5$, n_1 even, n_2 odd. Then O_{EO} minimizes $D(O)$, $L(O)$ and $M(O)$ over all strongly connected orientations O of $G_{n_1 \times n_2}$.

Proof. The proof for $D(O)$ is as in Theorem 1. Also, as in the proof of Theorem 1, it suffices, by Lemma 6, to show that for any O , $M(O) \geq 7$. Without loss of generality, we assume that $(c_0, c_1) \in O$. We argue by cases.

Case 1. $m(c_i) \geq 2$, $i = 1, 2, 3$. (This makes sense since $n_2 \geq 5$.) We may assume that $(c'_{q+1}, c'_q) \in O$, $q = 0, 1, \dots, k_2$, for otherwise $M(O) \geq 8$ by Lemma 11 and the hypothesis of Case 1. Now if $LP[y_0, w_3, F]$ or $LP[z_0, w_1, F]$, then by Lemma 3, $m(y_0) \geq 1$ or $m(z_0) \geq 1$. Thus $M(O) \geq 7$. Suppose $LP[y_0, w_3, T]$ and $LP[z_0, w_1, T]$. If $(y_0, c_0) \in O$, then $LP[c_0, w_1, F]$ because $(c'_1, c_0) \in O$. By Lemma 3, $m(c_0) \geq 1$, so $M(O) \geq 7$. Similarly, if $(z_0, c_0) \in O$, $m(c_0) \geq 1$ and $M(O) \geq 7$. So, we assume that $(c_0, y_0) \in O$ and $(c_0, z_0) \in O$. Since $(c'_{q+1}, c'_q) \in O$, $q = 0, 1, \dots, k_2$, limiting paths from y_0 to w_3 and z_0 to w_1 must meet in either some y'_p or some z'_q , say y'_p without loss of generality. Then $LP[y'_p, w_4, F]$ since arcs (y'_{p-1}, y'_p) and (c'_p, y'_p) are in O . By Lemma 2, $m(y'_p) \geq 2$. Hence, $M(O) \geq 8$.

Case 2. $m(c_1) \leq 1$. By Lemma 2, $LP[c_1, w_t, T]$, $t = 1, 3$. Since $(c_0, c_1) \in O$, we must have $(c_1, y_1) \in O$ and $(c_1, z_1) \in O$. Suppose $LP[y_1, w_3, T]$ or $LP[r_1, w_3, T]$. Let $h_3 = h_3(c_1, y_1)$ or $h_3(c_1, r_1)$. Because $(c_1, y_1) \in O$ and $(c_0, c_1) \in O$, h_3 is in R_3 and not in R_1 . By Lemma 4, $m(h_3) \geq 2$. Suppose $LP[y_1, w_3, F]$ and $LP[r_1, w_3, F]$. Then $m(y_1) \geq 2$ and $m(r_1) \geq 2$, by Lemma 2. By symmetric argument $LP[z_1, w_1, T]$ or $LP[s_1, w_1, T]$ implies there is $h_1 = h_1(c_1, z_1)$ or $h_1(c_1, s_1)$ in R_1 not in R_3 , with $m(h_1) \geq 2$, and $LP[z_1, w_1, F]$ and $LP[s_1, w_1, F]$ imply that $m(z_1) \geq 2$ and $m(s_1) \geq 2$. There are therefore four subcases:

- (a) $m(y_1) \geq 2$, $m(r_1) \geq 2$, $m(z_1) \geq 2$, $m(s_1) \geq 2$;
- (b) $m(h_3) \geq 2$, $m(z_1) \geq 2$, $m(s_1) \geq 2$;
- (c) $m(h_1) \geq 2$, $m(y_1) \geq 2$, $m(r_1) \geq 2$;
- (d) $m(h_3) \geq 2$, $m(h_1) \geq 2$.

In subcase (a), $M(O) \geq 8$ and we are done.

In subcase (b), suppose $m(c'_2) \geq 2$. Since $h_3 \notin R_1$, $h_3 \neq c'_2$. Hence, $M(O) \geq 8$ and we are done. So we assume $m(c'_2) \leq 1$. Then $LP[c'_2, w_4, T]$ by Lemma 2. If $LP[y_1, w_4, F]$, Lemma 3 implies that $m(y_1) \geq 1$, so $M(O) \geq 7$ follows. If $LP[y_1, w_4, T]$, let $h_4 = h_4(c'_2, y_1)$. Then $h_4 \in R_4$ so Lemma 4 implies that $m(h_4) \geq 2$. Since $(c_1, y_1) \in O$, $h_4 \neq z_1, s_1$. Since $R_3 \cap R_4 = \emptyset$, $h_3 \neq h_4$. Thus, $M(O) \geq 8$ follows.

Subcase (c) is analogous to subcase (b), using c'_2, w_2 and z_1 .

Turning to subcase (d), if we apply the argument of subcase (b), we find that $m(c'_2) \leq 1$ implies $m(y_1) \geq 1$ or that there is $h_4 \in R_4$ with $m(h_4) \geq 2$. A similar argument shows that $m(c'_2) \leq 1$ implies $m(r_1) \geq 1$ or that there is $h_4 \in R_4$ with $m(h_4) \geq 2$. Thus, one of the following holds:

- (d1) $m(c'_2) \geq 2$,
- (d2) $m(y_1) \geq 1, m(r_1) \geq 1$,
- (d3) $m(h_4) \geq 2$, for some $h_4 \in R_4$.

Note that as argued in subcase (b), $h_3 \neq c'_2$, $h_4 \neq z_1$ and $h_3 \neq h_4$. If $(y_0, y_1) \in O$, then since $(c_1, y_1) \in O$, $m(y_1) \geq 2$ by Lemma 2. Then $M(O) \geq 7$ in (d1), (d2) and (d3). If $(z_0, z_1) \in O$, then since $(c_1, z_1) \in O$, $m(z_1) \geq 2$ by Lemma 2. Then $M(O) \geq 8$ in (d1), (d2), and (d3). Finally, suppose that $(y_1, y_0) \in O$ and $(z_1, z_0) \in O$. If $(c_0, c'_1) \in O$, then by Lemma 11 either $m(c'_1) \geq 2$ or there are at least two vertices in $R_2 - R_4 \cup R_4 - R_2$ whose total m -value is at least four. In both cases, since $m(h_1) \geq 2$ and $m(h_3) \geq 2$, $M(O) \geq 8$ in (d1), (d2) and (d3). Hence, assume that $(c'_1, c_0) \in O$. If $(y_0, c_0) \in O$, then by Lemma 3, $m(c_0) \geq 1$. Since $c_0 \neq h_4$ because $c_0 \notin R_4$ and $c_0 \neq h_1, h_3$ since $h_1 \notin R_3$ and $h_3 \notin R_1$, we have $M(O) \geq 7$ in (d1), (d2) and (d3). Finally, if $(c_0, y_0) \in O$, then since $(y_1, y_0) \in O$, $m(y_0) \geq 2$ by Lemma 2. If $y_0 \neq h_1$, then $M(O) \geq 8$ in (d1), (d2) and (d3). Suppose $y_0 = h_1$. We know that $(c_1, y_1) \in O$ and $(c_1, z_1) \in O$. Consider the limiting path from s_1 or z_1 to w_1 , which goes through $h_1 = y_0$ by definition of h_1 . Since $(c_1, z_1) \in O$ this path must go from z_0 to c_0 , i.e., $(z_0, c_0) \in O$. We have assumed $(c'_1, c_0) \in O$. Thus $m(c_0) \geq 1$ by Lemma 3. Therefore, $M(O) \geq 7$ in (d1), (d2) and (d3).

Case 3. $m(c_1) \geq 2$ and $m(c_i) \leq 1$, $i = 2$ or $i = 3$. By Lemma 2 LP $[c_i, w_t, T]$, for $t = 1, 3$. There are three subcases:

- (a) $(c_1, y_1) \in O$,
- (b) $(c_1, z_1) \in O$,
- (c) $(y_1, c_1) \in O$ and $(z_1, c_1) \in O$.

Consider first subcase (a). If LP $[y_1, w_3, T]$ or LP $[r_1, w_3, T]$, let $h_3 = h_3(y_1, c_i)$ or $h_3(r_1, c_i)$. Since $(c_1, y_1) \in O$, h_3 is in R_3 . Since $(c_0, c_1) \in O$, $h_3 \notin R_1$. By Lemma 4, $m(h_3) \geq 2$. If LP $[y_1, w_3, F]$ and LP $[r_1, w_3, F]$, then by Lemma 2, $m(y_1) \geq 2$ and $m(r_1) \geq 2$. Similarly, LP $[z_0, w_1, T]$ or LP $[s_0, w_1, T]$ implies there is $h_1 = h_1(z_0, c_i)$ or $h_1(s_0, c_i)$ with h_1 in R_1 , h_1 not in R_3 , and $m(h_1) \geq 2$. The argument uses Lemma 4 and $(c_0, c_1) \in O$. Also, LP $[z_0, w_1, F]$ and LP $[s_0, w_1, F]$ implies $m(z_0) \geq 1$ and $m(s_0) \geq 1$, by Lemma 3. Thus we have one of the following:

- (a1) $m(h_3) \geq 2, m(h_1) \geq 2$;
- (a2) $m(h_3) \geq 2, m(z_0) \geq 1, m(s_0) \geq 1$;
- (a3) $m(y_1) \geq 2, m(r_1) \geq 2, m(h_1) \geq 2$;
- (a4) $m(y_1) \geq 2, m(r_1) \geq 2, m(z_0) \geq 1, m(s_0) \geq 1$.

Since $m(c_1) \geq 2$, we have $M(O) \geq 8$ in (a3) and (a4). We can assume $m(c'_1) \leq 1$, for otherwise $M(O) \geq 8$ in (a1) and (a2). By Lemma 2, LP $[c'_1, w_4, T]$ holds. If LP $[y_1, w_4, T]$ or LP $[r_1, w_4, T]$, let $h_4 = h_4(y_1, c'_1)$ or $h_4(r_1, c'_1)$. Since $(c_1, y_1) \in O$, $h_4 \neq c_1$ and $h_4 \in R_4$. Thus Lemma 4 implies $m(h_4) \geq 2$. Hence, since $m(c_1) \geq 2$, $M(O) \geq 8$ in (a1) and (a2). Finally, if LP $[y_1, w_4, F]$ and LP $[r_1, w_4, F]$, then by Lemma 3, $m(y_1) \geq 1$ and $m(r_1) \geq 1$. Since $m(c_1) \geq 2$, $M(O) \geq 8$ in (a1) and (a2).

In subcase (b), $M(O) \geq 8$ follows by a symmetric argument.

In subcase (c), note that strong connectedness implies that $(c_1, c_2) \in O$, else c_1 has no outgoing arcs. Then, the argument is the same as that in subcase (a) except for two changes. First, to show $h_3 \in R_3$, $h_3 \notin R_1$, one uses that $(c_1, c_2) \in O$ and $(z_1, c_1) \in O$. Second, it is possible that $h_4 = c_1$. Assume this is the case. Then arc (c'_1, c_0) is in O . If LP $[y_1, w_3, F]$, then by Lemma 2, $m(y_1) \geq 2$ and this plus $m(c_1) \geq 2$ gives $M(O) \geq 8$ in (a1) and (a2). Thus we may assume LP $[y_1, w_3, T]$ and so $(y_1, y_0) \in O$. Next, by Lemma 3, LP $[c_0, w_1, F]$ implies that $m(c_0) \geq 1$. This plus $m(c_1) \geq 2$ gives $M(O) \geq 7$ in (a1) and (a2). Thus we may assume LP $[c_0, w_1, T]$ and so, since $(c'_1, c_0) \in O$, we have $(c_0, y_0) \in O$. Similarly, we show that $(z_1, z_0) \in O$ and $(c_0, z_0) \in O$. Now $(c_0, y_0) \in O$ and

$(y_1, y_0) \in O$ imply that $m(y_0) \geq 2$. Since $(c_0, z_0) \in O$, y_0 could not be h_1 . Hence, $m(y_0) \geq 2$ and $m(c_1) \geq 2$ gives us $M(O) \geq 8$ in (a1) and (a2). $\square$

8. The orientation which alternates one-way streets. A one-way street assignment of $G_{n_1 \times n_2}$ which is in widespread use is the one which alternates the direction of north-south (vertical) streets and also of east-west (horizontal) streets. In this section, we compare this assignment to those we have studied, namely O_{EE} , O_{EO} and O_{OO} .

To be precise, let us obtain the *basic alternating orientation* by orienting all odd-numbered rows from right to left, and all even-numbered rows from left to right, all odd-numbered columns from top to bottom, and all even-numbered columns from bottom to top. If n_1 and n_2 are both odd, we call the resulting digraph O_{OO}^* . See Fig. 13. Unfortunately, unless n_1 and n_2 are both odd, the basic alternating orientation is not strongly connected. The simplest change to achieve strong connectedness is to change the orientation on two edges. If n_1 and n_2 are even, we change the orientation on edges $\{v_{(n_1+1)1}, v_{(n_1+1)2}\}$ and $\{v_{1(n_2+1)}, v_{2(n_2+1)}\}$, and obtain the strongly connected orientation O_{EE}^* . See Fig. 13. If n_1 is even and n_2 is odd, we change the orientation on edges $\{v_{(n_1+1)1}, v_{(n_1+1)2}\}$ and $\{v_{n_1(n_2+1)}, v_{(n_1+1)(n_2+1)}\}$, and obtain the strongly connected orientation O_{EO}^* . See Fig. 13.

Actually, there is a slightly more complicated change in the basic alternating orientation which seems to improve on O_{EE}^* , O_{EO}^* and O_{OO}^* by any of our criteria. Namely, we alternate boundary edges. Specifically, define O_{EE}^{**} from O_{EE}^* by changing the orientations of the following boundary edges of O_{EE}^* :

$$\begin{aligned} \{v_{1j}, v_{1(j+1)}\}, & \quad j = 2, 4, \dots, n_2 - 2, \\ \{v_{(n_1+1)j}, v_{(n_1+1)(j+1)}\}, & \quad j = 1, 3, \dots, n_2 - 1, \\ \{v_{i1}, v_{(i+1)1}\}, & \quad i = 2, 4, \dots, n_1, \\ \{v_{i(n_2+1)}, v_{(i+1)(n_2+1)}\}, & \quad i = 3, 5, \dots, n_1 - 1. \end{aligned}$$

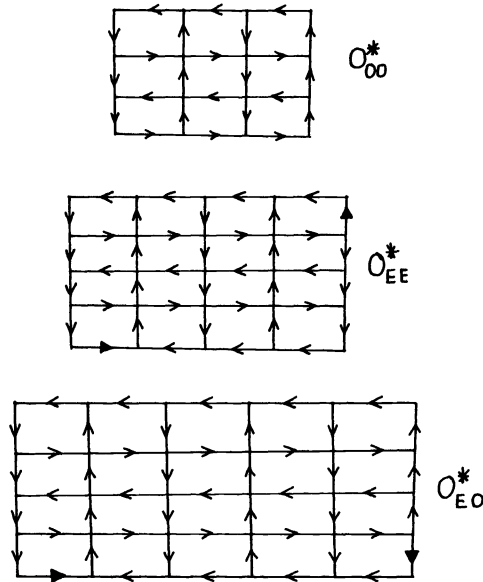


FIG. 13. O_{OO}^* for $n_1 = 3, n_2 = 3$, O_{EE}^* for $n_1 = 4, n_2 = 4$, O_{EO}^* for $n_1 = 4, n_2 = 5$. Dark arrows are changed from the basic alternating solution.

See Fig. 14. Similarly, define O_{EO}^{**} from O_{EO}^* by changing the orientations of the following boundary edges of O_{EO}^* :

$$\begin{aligned} \{v_{1j}, v_{1(j+1)}\}, & \quad j = 2, 4, \dots, n_2 - 1, \\ \{v_{(n_1+1)j}, v_{(n_1+1)(j+1)}\}, & \quad j = 1, 3, \dots, n_2 - 2, \\ \{v_{i1}, v_{(i+1)1}\}, & \quad i = 2, 4, \dots, n_1, \\ \{v_{i(n_2+1)}, v_{(i+1)(n_2+1)}\}, & \quad i = 2, 4, \dots, n_1 - 2. \end{aligned}$$

See Fig. 14. Finally, define O_{OO}^{**} from O_{OO}^* by changing the orientations of the following boundary edges of O_{OO}^* :

$$\begin{aligned} \{v_{ij}, v_{i(j+1)}\}, & \quad i = 1 \text{ or } n_1 + 1, \quad j = 2, 4, \dots, n_2 - 1, \\ \{v_{ij}, v_{(i+1)j}\}, & \quad i = 2, 4, \dots, n_1 - 1, \quad j = 1 \text{ or } n_2 + 1. \end{aligned}$$

See Fig. 14.

THEOREM 4.

- (a) $M(O_{EE}^*) \geq \frac{n_1 n_2}{2} + n_1 + n_2$;
- (b) $M(O_{EO}^*) \geq \frac{n_1 n_2}{2} + n_1 + n_2$;
- (c) $M(O_{OO}^*) \geq \frac{(n_1 + 1)(n_2 + 1)}{2} + \frac{n_1 + 1}{2} + \frac{n_2 + 1}{2} - 1$;
- (d) For $O^* = O_{EE}^*, O_{EO}^*$, or O_{OO}^* , $D(O^*) \geq n_1 + n_2 + 1$.

Proof. Let I_1 consist of all vertices v_{ij} in R_1 with i odd and j even, I_2 of all vertices in R_2 with i even and j even, I_3 of all vertices in R_3 with i odd and j odd and I_4 of all

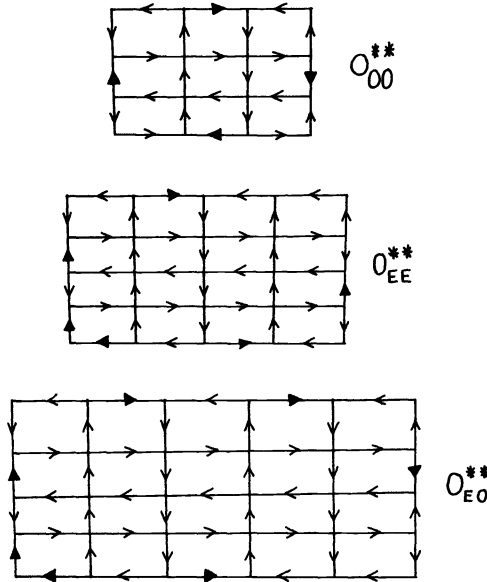


FIG. 14. O_{OO}^{**} for $n_1 = 3, n_2 = 3$, O_{EE}^{**} for $n_1 = 4, n_2 = 4$, O_{EO}^{**} for $n_1 = 4, n_2 = 5$. Dark arrows are changed from O_{OO}^* , O_{EE}^* or O_{EO}^* .

vertices in R_4 with i even and j odd. Then let I'_i consist of all vertices in I_i not in any other quadrant R_j , $j \neq i$. By Lemma 2, $m(u_i) \geq 2$ for u_i in I'_i except for $u_3 = v_{(n_1+1)1}$.

A simple counting argument shows that there are $k_1 k_2$ vertices in $X = I'_1 \cup I'_2 \cup I'_3 \cup I'_4$. If n_1 is even, one of these vertices is $v_{(n_1+1)1}$. Thus, we have that

$$(10) \quad \sum \{m(u): u \in X\} \geq 2(k_1 k_2 - 1) = 2k_1 k_2 - 2, \quad n_1 \text{ even},$$

and

$$(11) \quad \sum \{m(u): u \in X\} \geq 2k_1 k_2, \quad n_1 \text{ odd}.$$

Suppose n_1 and n_2 are both even. Then there are $k_1 + k_2 + 1$ vertices in

$$Y = (I_1 - I'_1) \cup (I_2 - I'_2) \cup (I_3 - I'_3) \cup (I_4 - I'_4),$$

namely, every vertex $v_{(k_1+1)j}$, $j \leq k_2 + 1$ if $(k_1 + 1)$ is odd, or $j \geq k_2 + 1$ if $(k_1 + 1)$ is even, and every vertex $v_{i(k_2+1)}$, $i \geq k_1 + 1$, if $(k_2 + 1)$ is odd, or $i \leq k_1 + 1$ if $(k_2 + 1)$ is even. By Lemma 2, $m(u) \geq 2$ for all u in Y . Thus,

$$(12) \quad \sum \{m(u): u \in Y\} \geq 2(k_1 + k_2 + 1) = 2k_1 + 2k_2 + 2.$$

By (10) and (12),

$$\begin{aligned} M(O_{EE}^*) &\geq 2k_1 k_2 - 2 + 2k_1 + 2k_2 + 2 \\ &= 2k_1 k_2 + 2k_1 + 2k_2 \\ &= \frac{n_1 n_2}{2} + n_1 + n_2. \end{aligned}$$

This proves (a). Next, since by Lemma 2 $m(v_{n_1(n_2+1)}) \geq 2$, $\bar{d}(v_{n_1(n_2+1)}, v_{11}) \geq n_1 + n_2 + 1$, so $D(O_{EE}^*) \geq n_1 + n_2 + 1$, which proves part of (d).

Next, suppose n_1 is even and n_2 is odd. Then there are k_2 vertices in Y , namely all vertices $v_{(k_1+1)j}$, $j \leq k_2$ if $(k_1 + 1)$ is odd, or $j \geq k_2 + 1$ if $(k_1 + 1)$ is even. Now $m(u) \geq 2$ for all u in Y , by Lemma 2. Thus,

$$(13) \quad \sum \{m(u): u \in Y\} \geq 2k_2.$$

Next, note that by Lemma 3, if k_2 is even, then $m(v_{ik_2}) \geq 1$ if i is even, $2 \leq i \leq k_1 + 1$, and $m(v_{i(k_2+1)}) \geq 1$ if i is odd, $k_1 + 1 \leq i \leq n_1 + 1$. Let Z be the set of all such v_{ik_2} and $v_{i(k_2+1)}$. Note that Z is disjoint from X and from Y . Also, there are $k_1 + 1$ vertices in Z . Hence,

$$(14) \quad \sum \{m(u): u \in Z\} \geq k_1 + 1.$$

By (10), (13) and (14),

$$\begin{aligned} M(O_{EE}^*) &\geq 2k_1 k_2 - 2 + 2k_2 + k_1 + 1 \\ &= \frac{n_1 n_2}{2} + n_1 + n_2, \end{aligned}$$

using $n_1 = 2k_1$, $n_2 = 2k_2 - 1$. Thus, (b) follows if k_2 is even. If k_2 is odd, then by Lemma 3, $m(v_{ik_2}) \geq 1$ for i even, $k_1 + 1 \leq i \leq n_1$, and $m(v_{i(k_2+1)}) \geq 1$ for i odd, $1 \leq i \leq k_1 + 1$. Let Z be the set of all such v_{ik_2} and $v_{i(k_2+1)}$. As above, Z is disjoint from X and from Y and has $k_1 + 1$ vertices. Hence, (14) follows and, as before, (b) follows. Next, note that by Lemma 2, $m(v_{2(n_2+1)}) \geq 2$, so $D(O_{EO}^*) \geq n_1 + n_2 + 1$ follows, proving another part of (d).

Finally, suppose n_1 and n_2 are odd. Let

$$a(k_1, k_2) = \begin{cases} k_1 + 1 & \text{if } k_1, k_2 \text{ odd,} \\ k_1 - 1 & \text{if } k_1 \text{ odd, } k_2 \text{ even,} \\ k_1 & \text{otherwise} \end{cases}$$

and

$$b(k_1, k_2) = \begin{cases} k_2 + 1 & \text{if } k_1, k_2 \text{ odd,} \\ k_2 - 1 & \text{if } k_1 \text{ even, } k_2 \text{ odd,} \\ k_2 & \text{otherwise.} \end{cases}$$

If k_2 is even, then by Lemma 3, $m(v_{ik_2}) \geq 1$ if i is even, $2 \leq i \leq k_1$, and $m(v_{i(k_2+1)}) \geq 1$ if i is odd, $k_1 + 1 \leq i \leq n_1$. Let Z be the set of such v_{ik_2} and $v_{i(k_2+1)}$. Then Z is disjoint from X and has $a(k_1, k_2)$ vertices. It follows that

$$(15) \quad \sum \{m(u) : u \in Z\} \geq a(k_1, k_2).$$

If k_2 is odd, then by Lemma 3, $m(v_{ik_2}) \geq 1$ if i is even, $k_1 + 1 \leq i \leq n_1 + 1$, and $m(v_{i(k_2+1)}) \geq 1$ if i is odd, $1 \leq i \leq k_1$. Let Z be the set of such v_{ik_2} and $v_{i(k_2+1)}$. Then, again, Z is disjoint from X and has $a(k_1, k_2)$ vertices, so (15) follows.

Next, if k_1 is even, then $m(v_{(k_1+1)j}) \geq 1$ if j is even, $2 \leq j \leq k_2$, and $m(v_{k_1j}) \geq 1$ if j is odd, $k_2 + 1 \leq j \leq n_2$. Let W be the set of all such $v_{(k_1+1)j}$ and v_{k_1j} . Then W is disjoint from X and Z and has $b(k_1, k_2)$ vertices. Hence,

$$(16) \quad \sum \{m(u) : u \in W\} \geq b(k_1, k_2).$$

If k_1 is odd, then $m(v_{(k_1+1)j}) \geq 1$ if j is even, $k_2 + 1 \leq j \leq n_2 + 1$, and $m(v_{k_1j}) \geq 1$ if j is odd, $1 \leq j \leq k_2$. Let W be the set of all such $v_{(k_1+1)j}$ and v_{k_1j} . Then W is disjoint from X and Z and has $b(k_1, k_2)$ vertices, so again (16) follows.

Now (11), (15) and (16) give us

$$\begin{aligned} M(O_{OO}^*) &\geq 2k_1k_2 + k_1 + k_2 - 1 \\ &= \frac{(n_1+1)(n_2+1)}{2} + \frac{n_1+1}{2} + \frac{n_2+1}{2} - 1, \end{aligned}$$

which proves (c). Finally, since by Lemma 2, $m(v_{n_11}) \geq 2$, $D(O_{OO}^*) \geq n_1 + n_2 + 1$, which completes the proof of (d). $\square$

Theorem 4 and Lemma 6 allow a comparison of O_{EE} to O_{EE}^* , O_{EO} to O_{EO}^* , and O_{OO} to O_{OO}^* . If M (or L) is to be minimized, O_{EE} , O_{EO} and O_{OO} are far superior to their corresponding alternating orientations. Even if D is to be minimized, O_{EE} , O_{EO} and O_{OO} are superior. However, these orientations are not superior on other criteria which have been suggested in the literature. For instance, consider

$$A(O) = \sum_{u,v} \bar{d}(u, v).$$

If $A(O)$ is to be minimized, the situation changes. For $A(O_{EE}^*)$ is less than $A(O_{EE})$, and similarly for OO and EO . For example, for $G_{6 \times 10}$, $A(O_{EE}^*) = 41,416$ and $A(O_{EE}) = 45,602$, and for $G_{5 \times 7}$, $A(O_{OO}^*) = 13,048$ and $A(O_{OO}) = 13,404$. Thus, whether or not the orientations introduced in this paper are preferred to the alternating orientations depends on the criterion of optimality adopted. We should note that the discrepancy between $A(O_{EE}^*)$ and $A(O_{EE})$ is not that great when averaged over the number of vertex pairs $p = |V|^2 - |V|$. For $A(O_{EE}^*)/p = 7.1$, while $A(O_{EE})/p = 7.8$, so there is a difference

of less than one in the average distance traveled. Similarly, $A(O_{\partial\partial}^*)/p = 5.8$ while $A(O_{\partial\partial})/p = 5.9$.

Next we turn to the orientations O_{EE}^* , O_{EO}^* and $O_{\partial\partial}^*$.

THEOREM 5.

- (a) $M(O_{EE}^*) \geq \frac{n_1 n_2}{2}$;
- (b) $M(O_{EO}^*) \geq \frac{n_1 n_2}{2}$;
- (c) $M(O_{\partial\partial}^*) \geq \frac{(n_1 + 1)(n_2 + 1)}{2} - \frac{n_1 + 1}{2} - \frac{n_2 + 1}{2}$;
- (d) For $O^{**} = O_{EE}^*$, O_{EO}^* , or $O_{\partial\partial}^*$, $D(O^{**}) \geq n_1 + n_2$.

The proof of Theorem 5 is similar to that of Theorem 4 and is omitted. Note that $A(O^{**})$ is less than $A(O^*)$. For example, for $G_{6 \times 10}$, $A(O_{EE}^*) = 40,352$, and for $G_{5 \times 7}$, $A(O_{\partial\partial}^*) = 12,500$.

9. Closing comments. In closing, we note that the optimal orientations have a weakness not captured by any of the criteria for optimality mentioned in § 2 or previously considered in the literature. Namely, they each have a vertex u with two one-way streets heading into u from opposite directions. Consider $u = u_2$ in O_{EE} , $u = u_5$ in O_{EO} , and $u = u_{10}$ in $O_{\partial\partial}$. Such a vertex might cause traffic havoc, and a criterion for optimality might want to add a “penalty” for having one. Whether or not O_{EE} , O_{EO} and $O_{\partial\partial}$ remain optimal will depend on the precise nature of the penalty.

The problems we have considered have analogues for other criteria of optimality. To the authors’ knowledge, no results have been obtained for optimal orientations for grid graphs under any other criteria.

The problems we have discussed are also of interest for other graphs. Of particular interest should be the graphs which result from annular cities, with circular beltways and spokes heading outward from the center.

REFERENCES

- [1] F. BOESCH AND R. TINDELL, *Robbins’ theorem for mixed graphs*, Amer. Math. Monthly, 87 (1980), pp. 716–719.
- [2] F. R. K. CHUNG, M. R. GAREY, AND R. E. TARJAN, *Strongly connected orientations of mixed multigraphs*, Networks, 15 (1985), pp. 477–484.
- [3] V. CHVÁTAL AND C. THOMASSON, *Distances in orientations of graphs*, J. Comb. Theory B, 24 (1978), pp. 61–75.
- [4] H. E. ROBBINS, *A theorem on graphs, with an application to a problem of traffic control*, Amer. Math. Monthly, 46 (1939), pp. 281–283.
- [5] F. S. ROBERTS, *Discrete Mathematical Models, with Applications to Social, Biological, and Environmental Problems*, Prentice-Hall, Englewood Cliffs, NJ, 1976.
- [6] ———, *Graph Theory and its Applications to Problems of Society*, CBMS-NSF Regional Conference Series in Mathematics 29, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1978.
- [7] F. S. ROBERTS AND Y. XU, *On the optimal strongly connected orientations of city street graphs II: Two east-west avenues or north-south streets*, Report RRR 9-86, Rutgers Center for Operations Research, Rutgers University, New Brunswick, NJ, May 1986, to appear.
- [8] ———, *On the optimal strongly connected orientations of city street graphs III: Three east-west avenues or north-south streets*, Report RRR 21-86, Rutgers Center for Operations Research, Rutgers University, New Brunswick, NJ, September 1986.
- [9] ———, *On the optimal strongly connected orientations of city street graphs IV: Four east-west avenues or north-south streets*, in preparation.

AN OPTIMALITY CRITERION FOR GRAPH EMBEDDINGS OF METRICS*

J. M. S. SIMÕES-PEREIRA†

Abstract. A criterion for optimality of graph realizations of distance matrices is presented. In order to illustrate the criterion's potential, it is used to prove the optimality of several well-known graph realizations and a few new ones.

Key words. distance matrices, graph realizations

AMS(MOS) subject classification. 05C

1. Introduction. Interest in graph embeddings of metrics or, as applied mathematicians prefer to say, graph realizations of distance matrices, has been increasing steadily. The literature on the subject is widespread, mainly because the topic relates to many different areas of pure and applied mathematics as the sample of papers we quote [1]–[38] clearly shows.

This paper presents a criterion for optimality of graph realizations. The criterion was developed to study matrices obtained from tree-realizable matrices by continuously decreasing selected sets of distances. Although successful in this case [32] and in another particular problem studied in [34], as with any criterion, no claim is made that it works for each and every graph. Two additional facts stress its usefulness: first, even for very small graphs, establishing optimality has always been a lengthy and difficult process; second, finding optimal realizations will likely remain very difficult since it has just been proven [1] that the problem is NP-complete. General criteria for optimality will therefore be most welcome.

In what follows: D or $\langle \{1, 2, \dots, n\} \rangle$ is a distance matrix of order n ; G is a graph whose edges are assigned lengths; d_{ij} is an entry of D ; $d(u, v)$ is the distance between vertices u and v in G ; and $p(u, v)$ is a shortest path in G linking u and v . A realization G of D is optimal if it has minimum total length among all realizations of D . Recall that graph realizations have main vertices whose mutual distances are the entries of D and auxiliary vertices. The main vertices of a graph realization G will be denoted by the respective indices of D ; no ambiguity will arise.

2. Main results. We start with a few results on the maximum length that two shortest paths may share in an optimal realization. (Obviously, maximum shared length equal to 0 means edge disjoint paths.)

LEMMA 2.1. *In any realization G of D , let v, u, w be distinct main vertices and let $L = (d_{uv} + d_{vw} - d_{uw})/2$. Then: (i) no point of a path $p(u, w)$ may be at a distance from v smaller than L ; (ii) no two paths $p(v, u)$ and $p(v, w)$ may share one point z with $d(v, z) > L$.*

Proof. To prove (i), suppose that z is on $p(u, w)$ and $d(v, z) = L' < L$. Obviously, $d(u, z) + d(z, w) = d_{uw}$. Further, $d_{uv} \leq d(u, z) + d(z, v)$ and $d_{vw} \leq d(w, z) + d(z, v)$. It follows that $d_{uv} + d_{vw} - d_{uw} \leq 2d(z, v) = 2L' < 2L$, a contradiction.

* Received by the editors May 1, 1987; accepted for publication June 5, 1987. This work was supported by the National Science Foundation under grant DMS-8401686 and by the Professional Staff Congress/City University of New York research award program.

† Hunter College and Graduate School, The City University of New York, New York, New York 10021, and Departamento de Matemática, Universidade de Coimbra, 3000-Coimbra, Portugal.

To prove (ii), suppose that $p(v, u)$ and $p(v, w)$ share one point z with $d(v, z) = L' > L$. Obviously, $d(u, z) < d_{uv} - L$ and $d(w, z) < d_{wv} - L$. Hence $d_{uw} < d_{uv} + d_{wv} - 2L$, a contradiction.

Remark to Lemma 2.1. Under the conditions of the lemma, (i) implies that each path $p(v, u)$ or $p(v, w)$ has a subpath of length at least L disjoint from each path $p(u, w)$; furthermore, this subpath of $p(v, u)$ or $p(v, w)$ which does not overlap $p(u, w)$ is necessarily the beginning, starting from v , of $p(v, u)$ or $p(v, w)$, respectively. Similarly, (ii) implies that no two paths $p(v, u)$ and $p(v, w)$ may share edges of total length greater than L , and if they do have overlapping edges they must be close to v : a common subpath of length L can only be what we call a terminal subpath, that is, a subpath that starts at v .

Now let $S_1 = d_{uv} + d_{wy}$, $S_2 = d_{uw} + d_{vy}$ and $S_3 = d_{uy} + d_{vw}$ be the Buneman sums [6] of the matrix $\langle \{u, v, w, y\} \rangle$, which is any principal submatrix of D .

LEMMA 2.2. *In any realization G of D , the following holds: (i) if $S_1 < \max \{S_2, S_3\}$, then any two paths $p(u, v)$ and $p(w, y)$ are always disjoint; (ii) if $S_1 \geq \max \{S_2, S_3\}$ and $H = (S_1 - \min \{S_2, S_3\})/2$, then no two paths $p(u, v)$ and $p(w, y)$ may share two points z' and z'' with $d(z', z'') > H$.*

Proof. (i) It was proven by Lemma 3.1 of [20].

(ii) Let z' and z'' be the first and last vertices, respectively, which $p(u, v)$ shares with $p(w, y)$ when scanned from u to v . If $d(z', z'') = H' > H$, then, obviously,

$$d(u, z') + d(z'', v) = d_{uv} - H' < d_{uv} - H.$$

Furthermore, either

$$d(w, z') + d(z'', y) < d_{wy} - H$$

or

$$d(y, z') + d(z'', w) < d_{wy} - H.$$

It follows that either

$$S_2 \leq d(u, z') + d(z'', v) + d(w, z') + d(z'', y) < d_{uv} - H + d_{wy} - H$$

or

$$S_3 \leq d(u, z') + d(z'', v) + d(y, z') + d(z'', w) < d_{uv} - H + d_{wy} - H,$$

which means that $S_1 - \min \{S_2, S_3\} > 2H$, a contradiction.

Remark to Lemma 2.2. The subpath $p(z', z'')$ shared by $p(u, v)$ and $p(w, y)$ is obviously nonterminal. If its length is maximal and $\langle \{u, v, w, y\} \rangle$ is tree-realizable, then its location within $p(u, v)$ (and within $p(w, y)$) is the same in any realization G of D . To see this, assume, without loss of generality, that $S_1 = S_3 > S_2$; since, by Lemma 2.1 (i), $d(u, z') \geq L_1 = (d_{uy} + d_{uw} - d_{wy})/2$, $d(z'', v) \geq L_2 = (d_{vy} + d_{vw} - d_{wy})/2$ and, therefore, $d(z', z'') = d(u, v) - d(u, z') - d(z'', v) \leq (S_1 - S_3 + S_1 - S_2)/2 = H$, we conclude that $d(z', z'') = H$ implies $d(u, z') = L_1$ and $d(z'', v) = L_2$. This is similar for $p(w, y)$.

If, however, $\langle \{u, v, w, y\} \rangle$ is nontree-realizable, then we can only say that $p(u, v)$ has two terminal subpaths disjoint from $p(w, y)$ and with total length at least $L_1 + L_2$ where $L_1 + L_2 + H < d_{uv}$.

LEMMA 2.3. *Let S' and S'' be Buneman sums of $\langle \{u, v, w, y\} \rangle$, with $S' < S''$. In any realization G of D , a path realizing one of the distances of S'' has a subpath of length at least $(S'' - S')/2$ disjoint from the (set-theoretic) union of any two paths realizing the distances of S' .*

Proof. With no loss of generality, suppose $S_1 < S_3$. By Lemma 2.2(i), $p(u, v)$ and $p(w, y)$ are disjoint. When going, say, from u to y along $p(u, y)$, let u' be the last point

common to $p(u, y)$ and $p(u, v)$; by the definitions and Lemma 2.2(i), u' precedes, in $p(u, y)$, the point y' which is the first point common to $p(u, y)$ and $p(w, y)$.

Suppose that $d(u', y') < (S_3 - S_1)/2$. Obviously,

$$d(u, u') + d(u', y') + d(y', y) = d_{uy}.$$

On the other hand,

$$d_{vw} \leq d(v, u') + d(u', y') + d(y', w).$$

Now from

$$S_1 = d_{uv} + d_{wy} = d(u, u') + d(u', v) + d(w, y') + d(y', y)$$

and

$$\begin{aligned} S_3 &= d_{uy} + d_{vw} \leq d(u, u') + d(u', y') + d(y', y) + d(y', w) \\ &= S_1 + 2d(u', y') \end{aligned}$$

we obtain

$$S_3 - S_1 \leq 2d(u', y') < S_3 - S_1,$$

a contradiction that proves the lemma.

Remark to Lemma 2.3. When $\langle \{u, v, w, y\} \rangle$ is tree-realizable, if $p(u', y')$ has minimal length, then its location within $p(u, y)$ is the same in any realization G of D . Note that by Lemma 2.1(ii), $d(u, u') \leq L_1 = (d_{uy} + d_{uv} - d_{yv})/2$ and $d(y, y') \leq L_2 = (d_{yu} + d_{yw} - d_{uw})/2$; hence

$$d(u', y') = d_{uy} - d(u, u') - d(y, y') \geq d_{uy} - L_1 - L_2 = (S_2 - S_1)/2 = (S_3 - S_1)/2.$$

It follows that if $d(u', y') = (S_3 - S_1)/2$, then $d(u, u') = L_1$ and $d(y, y') = L_2$.

The preceding lemmas are very useful for proving optimality of graph realizations. Used separately or jointly, they easily lead to optimality criteria essentially based on our ability to find, in a given realization G of D , a collection P of shortest paths that contain all edges of G and overlap as much as is allowed by the lemmas and their remarks.

One such criterion has just been used to study optimal realizations of matrices arising from tree-realizable matrices by continuously decreasing selected sets of distances [32]; it may be formulated as follows and involves only Lemmas 2.1 and 2.2.

OPTIMALITY CRITERION. Let G be a realization of D and Π be a set of pairs of main vertices of G , each pair joined in G by one single shortest path; use P to denote this set of paths and suppose that they contain all edges of G . Consider each pair of paths in P . Suppose that:

Case (Z) When $p \equiv p(u, v)$ and $p' \equiv p(w, v)$ (or when $p \equiv p(u, v)$ and $p' \equiv p(w, y)$) are edge disjoint in G , the distances among u, v, w , and y satisfy the hypothesis of Lemma 2.1 with $L = 0$ (or of Lemma 2.2(i) or 2.2(ii) with $H = 0$). Furthermore, for pairs of paths that overlap in G , suppose that we always have one of the following cases:

Case (A) When the paths are $p \equiv p(u, v)$ and $p' \equiv p(w, v)$, they share the one and only subpath of maximum length as given by Lemma 2.1 and its remark.

Case (B) When the paths are $p \equiv p(u, v)$ and $p' \equiv p(w, y)$, when $\langle \{u, v, w, y\} \rangle$ is tree-realizable and when the distances among u, v, w , and y satisfy the hypothesis of Lemma 2.2(ii) with $H > 0$, p and p' share the one and only subpath of length H as given by Lemma 2.2(ii) and its remark.

Then G is optimal.

Proof. Obviously, the paths in P account for the total length of G , and in any other realization G' of D we must have at least one shortest path joining each vertex pair in

II; moreover, the length of each shortest path in G is equal to the length of its corresponding shortest path in G' .

Let us consider the pairs in Π successively, say $\pi_1, \pi_2, \dots, \pi_t$, and denote by $p_1, p_2, \dots, p_t$ the respective paths in P . For $i = 1$, let $r_i(G) = 0$ and, for $i > 1$, let $r_i(G)$ be the total length of the edges of G which lie in both p_i and the union of all p_k with $k < i$. Let $r_i(G')$ be similarly defined. Since the total length of any realization G' of D must be at least $\sum_{i=1}^t (|p_i| - r_i(G'))$, where $|p_i|$ is the length of p_i , to prove that G is optimal it is enough to show that for every i , $r_i(G) \geq r_i(G')$. We claim that this is true. In fact, if $p_i \equiv p(u, v)$ and $p_k \equiv p(v, w)$ are edge disjoint in G , then, by our assumptions, the distances among u, v , and w satisfy the hypothesis of Lemma 2.1 with $L = 0$; hence p_i and p_k are disjoint in any G' . Similarly, if $p_i \equiv p(u, v)$ and $p_k \equiv p(w, y)$ are edge disjoint in G , then the distances among u, v, w and y satisfy the hypothesis of Lemma 2.2(i) or 2.2(ii) with $H = 0$; hence p_i and p_k are disjoint in any G' . In these cases the contributions to r_i are 0 in G and any G' . Let $b_{i,1}$ and $b_{i,2}$ be the lengths of the two longest terminal subpaths of p_i which, by Lemma 2.1 and its remark, p_i may share with those $p_k, k < i$, having one terminal vertex in common with p_i . These subpaths contribute $b_{i,1} + b_{i,2}$ to r_i and, by the hypothesis of Case (A), this contribution is maximal in G among all realizations. Finally, let $p_{i,1}, p_{i,2}, \dots, p_{i,s}$ be the uniquely located maximum length subpaths which $p_i \equiv p(u, v)$ may share with each one of the, say, $s (s < i)$ paths p_k , each linking a different vertex pair $\{w, y\}$, such that the distances among u, v, w , and y satisfy the hypothesis of Lemma 2.2(ii) with $H > 0$. By the hypothesis of Case (B), the contribution of these shared subpaths to r_i is maximal in G , since the uniqueness of the location where the subpaths may occur within p_i and their maximum length in G preclude any possibility of having a larger total length for their union. This proves the claim and the criterion.

An extension of this criterion accepts paths $p \equiv p(u, v)$ and $p' \equiv p(w, y)$ with $\langle \{u, v, w, y\} \rangle$ nontree-realizable and the distances among u, v, w , and y satisfying the hypothesis of Lemma 2.2(ii) with $H > 0$. To guarantee optimality of G , the maximum length subpath that p shares with p' must be edge disjoint from all other subpaths shared by p and other paths in P . The reason is that the location of such a subpath is not unique within p . Another extension or variant uses Lemma 2.3 in a role similar to that of Lemma 2.2.

3. Optimality of some well-known and some new realizations. A long proof that the graph in Fig. 3.1 is optimal was given in [35]; now, optimality follows easily from our criterion: we take the pairs $\{1, 2\}, \{3, 4\}, \{2, 3\}, \{1, 4\}$. To see the full details, note that, in any realization, $p(3, 4)$ is disjoint from $p(1, 2)$; then $p(2, 3)$, by Case (A), may share at most e with $p(1, 2)$ and f with $p(3, 4)$. Finally, $p(1, 4)$, by Case (A), may share at most c with $p(1, 2)$ and at most g with $p(3, 4)$ and, by Case (Z), is disjoint from $p(2, 3)$. When we use T to denote the total length of a realization then: the existence of

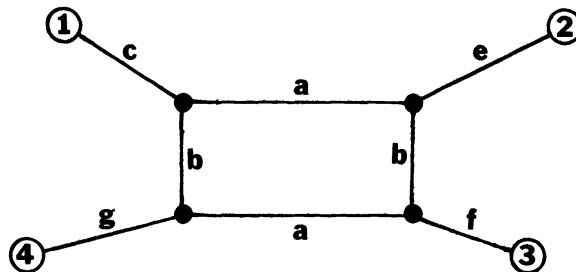


FIG. 3.1

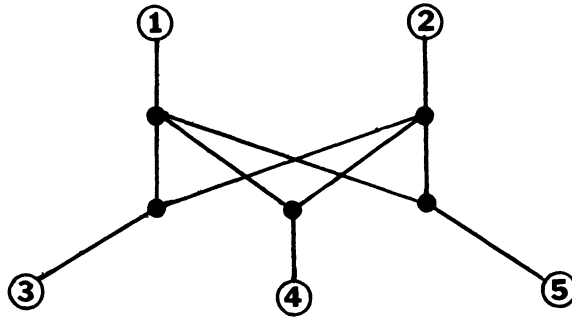


FIG. 3.2

$p(1, 2)$ in any G' implies that $T(G') \geq d_{12}$; $p(3, 4)$ implies that $T(G') \geq d_{12} + d_{34}$; $p(2, 3)$ implies that $T(G') \geq d_{12} + d_{34} + d_{23} - e - f = d_{12} + d_{34} + b$; finally, $p(1, 4)$ implies that $T(G') \geq d_{12} + d_{34} + b + d_{14} - c - g = T(G)$.

Tree realizations are known to be optimal [16]; to apply our criterion, let Π be the set of vertex pairs $\{i, j\}$ where i is pendant and fixed and j runs over all pendant vertices distinct from i , and use Case (A).

The graphs in Figs. 3.2, 3.3, and 3.4 are optimal realizations. For the first, with all nonpendant edges of the same length, use the pairs $\{1, 3\}$, $\{1, 4\}$, $\{1, 5\}$, $\{2, 3\}$, $\{2, 4\}$, $\{2, 5\}$. For the second, which is a compact graph with the minimum number (2) of nonbasic distances [35], use the pairs $\{1, 2\}$, $\{1, 5\}$, $\{1, 4\}$, $\{3, 2\}$, $\{3, 5\}$, $\{3, 4\}$. For the third (with all edges of the same length) use the pairs $\{1, 8\}$, $\{2, 7\}$, $\{3, 6\}$, $\{4, 5\}$, $\{2, 3\}$, $\{4, 1\}$, $\{5, 8\}$, $\{6, 7\}$ and the first extension of the criterion.

A long proof that the graph G in Fig. 3.5, where $-\frac{1}{2} \leq e \leq \frac{1}{2}$, is optimal is given in [1], where G was introduced to show that a distance matrix may have a continuum of optimal realizations. In the following proof we apply the second extension of our criterion when discussing the last path. For clarity, we specify the lemmas we use in each step. Consider the pairs $\{1, 2\}$, $\{2, 3\}$, $\{3, 4\}$, $\{4, 5\}$, $\{5, 6\}$, $\{6, 1\}$, $\{1, 4\}$. For the first six pairs, Case (Z) applies; any optimal realization G' therefore contains the pairwise disjoint paths of the polygon whose vertices are 1, 2, 3, 4, 5, 6 and whose length is 8. Consider now the pair $\{1, 4\}$. In G' , by Lemma 2.1, the length of a subpath shared by $p(1, 4)$ with $p(1, 2)$ or $p(4, 5)$ is at most $\frac{1}{2}$, and that of a subpath shared by $p(1, 4)$ with $p(1, 6)$ or $p(3, 4)$ is at most $\frac{3}{2}$. By Lemma 2.2(i), $p(1, 4)$ is disjoint from $p(2, 3)$ and $p(5, 6)$. Since $d_{14} = 3$, $p(1, 4)$ must have subpaths of total length at least 1 disjoint from the union of $p(1, 2)$ and $p(3, 4)$ or $p(1, 6)$ and $p(4, 5)$, and of total length at least 2 disjoint from the union of $p(1, 2)$ and $p(4, 5)$; by their very definitions, $p(1, 4)$ cannot share subpaths with both $p(1, 2)$ and $p(1, 6)$ or with both $p(3, 4)$ and $p(4, 5)$; by Lemma 2.3, it has a subpath

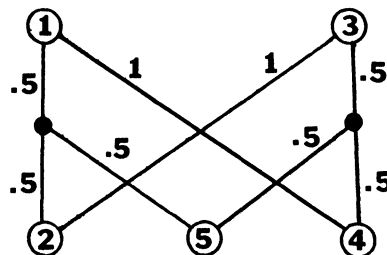


FIG. 3.3

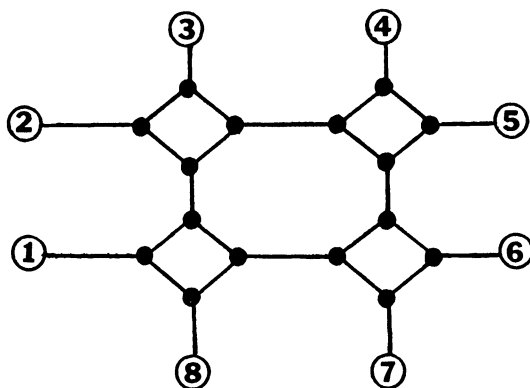


FIG. 3.4

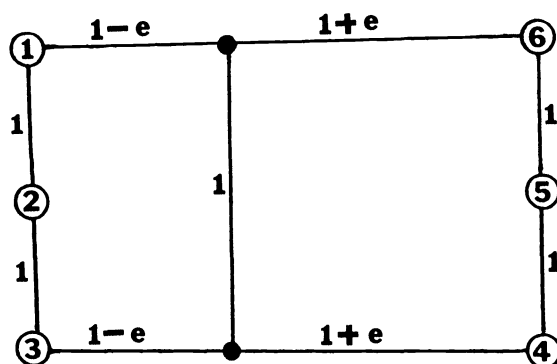


FIG. 3.5

of length at least 1 disjoint from the union of $p(1, 6)$ and $p(3, 4)$. It follows that $p(1, 4)$ adds at least 1 unit to the total length of the above mentioned polygon of G' , which proves the optimality of G .

REFERENCES

- [1] I. ALTHÖFER, *On optimal realizations of finite metric spaces by graphs*, Discrete and Computational Geometry, 3 (1988), pp. 103–122.
- [2] H.-J. BANDELT, *Ein Axiomensystem für Baum-Algebren*, Časopis Pěst. Mat., 108 (1983), pp. 353–355.
- [3] H.-J. BANDELT AND J. HEDLÍKOVÁ, *Median algebras*, Discrete Math., 45 (1983), pp. 1–30.
- [4] H.-J. BANDELT AND H. M. MULDER, *Distance-hereditary graphs*, J. Combin. Theory Ser. B, 41 (1986), pp. 182–208.
- [5] F. T. BOESCH, *Properties of the distance matrix of a tree*, Quart. Appl. Math., 26 (1968–69), pp. 607–609.
- [6] P. BUNEMAN, *A note on the metric properties of trees*, J. Combin. Theory Ser. B, 17 (1974), pp. 48–50.
- [7] S. CHAIKEN, A. K. DEWDNEY, AND P. J. SLATER, *An optimal diagonal tree code*, SIAM J. Algebraic Discrete Methods, 4 (1983), pp. 42–49.
- [8] J. A. CUNNINGHAM, *Free trees and bidirectional trees as representations of psychological distance*, J. Math. Psych., 17 (1978), pp. 165–188.
- [9] A. K. DEWDNEY, *Diagonal tree codes*, Inform. and Control, 40 (1979), pp. 234–239.
- [10] J. K. DOYLE AND J. E. GRAVER, *Mean distance for shapes*, J. Graph Theory, 6 (1982), pp. 453–471.
- [11] A. W. M. DRESS, *Trees, tight extensions of metric spaces and the cohomological dimension of certain groups—a note on combinatorial properties of metric spaces*, Adv. in Math., 53 (1984), pp. 321–402.
- [12] M. EDELBERG, M. R. GAREY, AND R. L. GRAHAM, *On the distance matrix of a tree*, Discrete Math., 14 (1976), pp. 23–29.

- [13] R. L. GRAHAM, A. J. HOFFMAN, AND H. HOSOYA, *On the distance matrix of a directed graph*, J. Graph Theory, 1 (1977), pp. 85–88.
- [14] R. L. GRAHAM AND L. LOVASZ, *Distance matrix polynomials of trees*, Adv. in Math., 29 (1978), pp. 60–88.
- [15] R. L. GRAHAM AND P. M. WINKLER, *On isometric embeddings of graphs*, Trans. Amer. Math. Soc., 288 (1985), pp. 527–536; Corrigendum, 294 (1986), p. 379.
- [16] S. L. HAKIMI AND S. S. YAU, *Distance matrix of a graph and its realizability*, Quart. Appl. Math., 22 (1964–65), pp. 305–317.
- [17] E. HOWE, C. R. JOHNSON, AND J. LAWRENCE, *The structure of distances in networks*, Networks, 16 (1986), pp. 87–106.
- [18] W. IMRICH, *On metric properties of tree like spaces*, in Contributions to Graph Theory and Its Applications, Intern. Colloq. Oberhof 1977, Technische Hochschule Ilmenau, Ilmenau, 1977, pp. 129–156.
- [19] W. IMRICH AND G. SCHWARZ, *Trees and length functions in groups*, Ann. Disc. Math., 17 (1983), pp. 347–359.
- [20] W. IMRICH, J. M. S. SIMÕES-PEREIRA, AND C. M. ZAMFIRESCU, *On optimal embeddings of metrics in graphs*, J. Combin. Theory Ser. B, 36 (1984), pp. 1–15.
- [21] W. IMRICH AND E. STOTZKII, *The optimal embeddings of metrics into graphs*, Siberian Math. J., 13 (1972), pp. 558–565. (In Russian.)
- [22] N. IWAHORI, *On some algorithm of the number of branch points and their degrees starting from the distance matrix of a given tree*, Sci. Papers College Arts Sci. Univ. Tokyo, 33 (1983), pp. 63–81.
- [23] R. A. MELTER AND I. TOMESCU, *Isometric embeddability for graphs*, Ars Combin., 12 (1981), pp. 111–115.
- [24] A. NEUMAIER, *Distances, graphs and designs*, European J. Combin., 1 (1980), pp. 163–174.
- [25] ———, *Distance matrices and n -dimensional designs*, European J. Combin., 2 (1981), pp. 165–172.
- [26] A. N. PATRINOS AND S. L. HAKIMI, *The distance matrix of a graph and its tree realization*, Quart. Appl. Math., 30 (1972–73), pp. 255–269.
- [27] D. SANKOFF AND R. J. CEDERGREN, *Simultaneous comparisons of three or more sequences related by a tree*, in Time Warps, String Edits and Macromolecules: The Theory and Practice of Sequence Comparison, D. Sankoff and J. B. Kruskal, eds., Addison-Wesley, Reading, MA, 1983, pp. 253–263.
- [28] J. M. S. SIMÕES-PEREIRA, *Some results on the tree realization of a distance matrix*, in Théorie des Graphes—Journées Internationales d'Etude, Rome, 1966, Dunod Éditeur, Paris 1967, pp. 383–388.
- [29] ———, *A note on the tree realizability of a distance matrix*, J. Combin. Theory, 6 (1969), pp. 303–310.
- [30] ———, *A note on optimal and suboptimal digraph realizations of quasidistance matrices*, SIAM J. Algebraic Discrete Methods, 5 (1984), pp. 117–132.
- [31] ———, *A note on distance matrices with unicyclic graph realizations*, Discrete Math., 65 (1987), pp. 277–287.
- [32] ———, *Underlying graph and total length of optimal realizations of variable distance matrices*, Graphs and Combinatorics, 3 (1987), pp. 383–393.
- [33] ———, *A note on convexity and submatrices of distance matrices*, Linear and Multilinear Algebra, 20 (1987), pp. 363–366.
- [34] ———, *An optimality criterion and the total length of the graph realization of a distance matrix*, Proc. 3rd International Conference on Combinatorics, Ann. New York Acad. Sci., to appear.
- [35] J. M. S. SIMÕES-PEREIRA AND C. M. ZAMFIRESCU, *Submatrices of nontree-realizable distance matrices*, Linear Algebra Appl., 44 (1982), pp. 1–17.
- [36] E. A. SMOLENSKII, *A method for the linear recording of graphs*, Zh. Vychisl. Mat. i Mat. Fiz., 2 (1962), pp. 371–372. (In Russian.) USSR Comput. Math. and Math. Phys., 2 (1963), pp. 396–397. (In English.)
- [37] S. V. YUSMANOV, *Reconstruction of an arbitrary graph from certain subsets of lines and columns of its distance matrix*, Akad. Nauk SSSR Doklady, 259 (1981), pp. 49–52. (In Russian.)
- [38] K. A. ZARETZKII, *Constructing a tree on the basis of a set of distances between the hanging vertices*, Uspekhi Mat. Nauk, 20 (1965), pp. 90–92. (In Russian.)

A COMBINATORIAL APPROACH TO THRESHOLD SCHEMES*

D. R. STINSON† AND S. A. VANSTONE‡

Abstract. We investigate the combinatorial properties of threshold schemes. Informally, a (t, w) -threshold scheme is a way of distributing partial information (shadows) to w participants, so that any t of them can easily calculate a key, but no subset of fewer than t participants can determine the key. Our interest is in *perfect* threshold schemes: no subset of fewer than t participants can determine any partial information regarding the key. We give a combinatorial characterization of a certain type of perfect threshold scheme. We also investigate the maximum number of keys that a perfect (t, w) -threshold scheme can incorporate, as a function of t , w , and the total number of possible shadows, v . This maximum can be attained when there is a Steiner system $S(t, w, v)$ that can be partitioned into Steiner systems $S(t-1, w, v)$. Using known constructions for such Steiner systems, we present two new classes of perfect threshold schemes, and discuss their implementation.

Key words. threshold scheme, secret sharing, combinatorial design, Steiner system

AMS(MOS) subject classifications. 05B05, 68R05, 94A60

1. Introduction. Let X be a set of v elements (which we refer to as *shadows*), and let K be a set of m elements (called *keys*). A (t, w) -threshold scheme is a pair $(\mathcal{B}, \phi)$, where $\mathcal{B}$ is a set of b (distinct) w -subsets of X (*blocks*), and $\phi: \mathcal{B} \rightarrow K$, such that the following properties are satisfied:

- (1) Any t shadows determine at most one key (i.e., for every t -subset S of X , $|\{\phi(B): S \subseteq B \in \mathcal{B}\}| = 0$ or 1).
- (2) Any set of fewer than t shadows that occur in a block do *not* determine a unique key (i.e., for every t' -subset S of X , where $t' < t$, $|\{\phi(B): S \subseteq B \in \mathcal{B}\}| \neq 1$).

The idea behind threshold schemes is that we wish to give partial information (shadows) to w people, so that any t of them can determine the key, but no group of fewer than t can do so. Suppose we want to send shadows corresponding to key k . We do this by choosing a block B such that $\phi(B) = k$. Then, we give each of the w participants a different shadow in B .

For example, the key could be the combination of a safe, and we might desire that two of three specified people be required in order to determine this combination. This would correspond to a threshold scheme with $w = 3$ and $t = 2$.

Threshold schemes were first described by Shamir [9] and Blakely [5] in 1979. Since then, many constructions have been given for threshold schemes. Most of the constructions are linear algebraic in nature (see, for example, Kothari [8]). Recently, Beutelspacher [4] has given some constructions for threshold schemes using finite geometries. The purpose of this paper is to investigate the properties of threshold schemes from a combinatorial viewpoint. This more general approach enables us to give some new constructions for threshold schemes based on combinatorial designs. In particular, we give constructions for $(3, 3)$ - and $(3, 4)$ -threshold schemes that handle more keys than do previously known schemes.

Property (2) in the definition of threshold schemes states that t participants are required in order to determine the key k . However, it is possible that a group of t' ($< t$)

* Received by the editors July 6, 1987; accepted for publication (in revised form) December 10, 1987. A preliminary version of this paper appeared in *Advances in Cryptology—Proceedings of Crypto 87*, Lecture Notes in Computer Science, Vol. 293, Springer-Verlag, 1988.

† Department of Computer Science, University of Manitoba, Winnipeg, Manitoba, Canada R3T 2N2.

‡ Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada, N2L 3G1.

participants may be able to obtain some partial information about the key. For example, they may be able to rule out certain elements of $\mathbf{K}$. Ideally, we would like to have threshold schemes in which no partial information can be obtained by fewer than t participants.

To make these ideas precise, we discuss the idea of security for threshold schemes (see, for example, [6]). First, we introduce some probability distributions. We assume that we are given a specified probability distribution on the key space $\mathbf{K}$. For every key k , we then choose a probability distribution on the blocks in $\phi^{-1}(k)$. Together, these determine a probability distribution on $\mathcal{B}$.

Now, suppose a block B has been chosen, and the shadows distributed to the participants. Suppose a subset of the participants have been given the shadows in the set $S \subseteq B$. These participants can compute the conditional probability distribution on $\mathbf{K}$, given that they possess the shadows in S , as follows:

$$\begin{aligned} p(k \mid S) &= p(S \mid k) \cdot p(k) / p(S) \\ &= \frac{\sum_{\{B \in \phi^{-1}(k): S \subseteq B\}} p(B \mid k) \cdot p(k)}{\sum_{k' \in \mathbf{K}} \sum_{\{B \in \phi^{-1}(k'): S \subseteq B\}} p(B \mid k') \cdot p(k')}. \end{aligned}$$

If $p(k) \neq p(k \mid S)$ for some key k , then these participants have obtained some (partial) information regarding the actual key that was sent. This motivates the following rigorous definition of security in a threshold scheme. Given a (t, w) -threshold scheme, and given $t' < t$, we say that the threshold scheme is *perfectly t' -secure* if for every subset $S \subseteq \mathbf{X}$ of cardinality t' that occurs as a subset of at least one block, and for every key k , we have that $p(k \mid S) = p(k)$. (Similar definitions for security of computations with encrypted data are given in [1].)

We will refer to a threshold scheme as *regular* if b/m blocks correspond to each possible key and, for every key k , each block in $\phi^{-1}(k)$ is chosen with equal probability m/b . Given $S \subseteq \mathbf{X}$ and $k \in \mathbf{K}$, define $\lambda(S, k) = |\{B \in \phi^{-1}(k): S \subseteq B\}|$. It follows that, in a regular threshold scheme, we have

$$\begin{aligned} p(k \mid S) &= \frac{\sum_{\{B \in \phi^{-1}(k): S \subseteq B\}} p(k)}{\sum_{k' \in \mathbf{K}} \sum_{\{B \in \phi^{-1}(k'): S \subseteq B\}} p(k')} \\ &= \frac{p(k) \cdot \lambda(S, k)}{\sum_{k' \in \mathbf{K}} p(k') \cdot \lambda(S, k')}. \end{aligned}$$

In a regular scheme, $p(k \mid S) = p(k)$ if and only if

$$\lambda(S, k) = \sum_{k' \in \mathbf{K}} p(k') \cdot \lambda(S, k').$$

Thus, a regular threshold scheme is perfectly t' -secure if and only if, for every $S \subseteq \mathbf{X}$ of cardinality t' , $\lambda(S, k)$ is independent of the key k . Equivalently, we have the following lemma.

LEMMA 1.1. *A regular (t, w) -threshold scheme $(\mathcal{B}, \phi)$ is perfectly t' -secure if and only if the following property holds. For every $S \subseteq \mathbf{X}$ of cardinality t' , there exists a nonnegative integer $\lambda(S)$, such that, for every key k , we have*

$$|\{B \in \phi^{-1}(k): S \subseteq B\}| = \lambda(S).$$

The following result is now an immediate consequence.

LEMMA 1.2. *If a regular threshold scheme is perfectly t' -secure, then it is perfectly t'' -secure for all t'' , $1 \leq t'' \leq t'$.*

Proof. Given a subset T , where $|T| = t''$, we have

$$\lambda(T) = \sum_{\{S: |S| = t' \text{ and } T \subseteq S\}} \lambda(S) / \binom{w-t''}{t'-t''}. \quad \square$$

2. A combinatorial characterization of perfect threshold schemes. Our main interest is in regular (t, w) -threshold schemes that are perfectly $(t-1)$ -secure. We refer to such a threshold scheme as *perfect*. Next, we give a characterization of perfect threshold schemes in terms of the blocks corresponding to each key.

Let $\mathcal{A}$ be a set of w -subsets (*blocks*) of a v -set X . We refer to $(X, \mathcal{A})$ as a w -uniform hypergraph, and we say that v is the number of *points* in the hypergraph. Given any integer $t' \leq w$, define a multi-set $\mathcal{A}(t') = \bigcup_{A \in \mathcal{A}} \{S: |S| = t', S \subseteq A\}$. Note that $\mathcal{A}(t')$ can contain “repeated” t' -subsets. The *multiplicity* of a t' -subset is the number of times it occurs in $\mathcal{A}(t')$. We also say that $\mathcal{A}(t')$ is the multi-set of *induced* t' -subsets of $\mathcal{A}$.

Two w -uniform hypergraphs $(X, \mathcal{A}_1)$ and $(X, \mathcal{A}_2)$ are defined to be t -compatible if the following two properties are satisfied:

- (1) $\mathcal{A}_1(t-1) = \mathcal{A}_2(t-1)$, and
- (2) $\mathcal{A}_1(t) \cap \mathcal{A}_2(t) = \emptyset$.

The following result characterizes perfect (t, w) -threshold schemes in terms of t -compatible w -uniform hypergraphs.

THEOREM 2.1. *There exists a perfect (t, w) -threshold scheme having v shadows and m keys if and only if there exist m mutually t -compatible w -uniform hypergraphs on v points.*

One way to approach the construction of a perfect (t, w) -threshold scheme having v shadows is to start with a multi-set $\mathcal{S}$ of $(t-1)$ -subsets of a v -set, and attempt to find t -compatible w -uniform hypergraphs $\mathcal{A}_1, \dots, \mathcal{A}_m$ such that $\mathcal{A}_i(t-1) = \mathcal{S}$, $1 \leq i \leq m$ (that is, so that $\mathcal{S}$ is the multi-set of induced $(t-1)$ -subsets of each $\mathcal{A}_i$). Given t, w, v , and $\mathcal{S}$, we would want to find the maximum number of such hypergraphs (equal to the maximum number of keys in the resulting threshold system). We denote this number by $m(t, w, v, \mathcal{S})$. Also, denote

$$m(t, w, v) = \max \{m(t, w, v, \mathcal{S}): \mathcal{S} \text{ is a multiset of } (t-1)\text{-subsets of a } v\text{-set}\}.$$

As we would suspect, determining the numbers $m(t, w, v, \mathcal{S})$ is very difficult. Holyer [7] has proved that the question “can a graph G be edge-decomposed into triangles?” is NP-complete. This question can be rephrased as “is $m(3, 3, v, E(G)) \geq 1$?”, where G has v vertices and $E(G)$ denotes the set of edges of G . Hence, determining the numbers $m(t, w, v, \mathcal{S})$ is NP-hard.

We can, however, give an upper bound on $m(t, w, v, \mathcal{S})$. This bound depends on the largest multiplicity of any t' -subset in $\mathcal{S}$.

THEOREM 2.2. *Let λ be the largest multiplicity of any t' -subset in $\mathcal{S}$. Let u denote the smallest positive integer such that*

$$\binom{u}{w-t+1} \geq \lambda.$$

Then $m(t, w, v, \mathcal{S}) \leq (v-t+1)/u$.

Proof. Let $\mathcal{A}$ be a w -uniform hypergraph on v points such that $\mathcal{A}(t-1) = \mathcal{S}$. Let $S \in \mathcal{S}$ have multiplicity λ . The λ blocks in $\mathcal{A}$ that contain S must all be distinct. Hence, together they contain at least u different elements.

Now, suppose we have m t -compatible hypergraphs. From each of the m hypergraphs, we obtain a set of u elements as described above. These m sets must be disjoint, since

otherwise the respective multi-sets of induced t -subsets would not be disjoint. Also, these m sets are disjoint from S . Hence, $m \leq (v - t + 1)/u$. $\square$

As a corollary, we obtain an upper bound on $m(t, w, v)$.

COROLLARY 2.3. $m(t, w, v) \leq (v - t + 1)/(w - t + 1)$.

Proof. In order to maximize the bound on m , we minimize u . This occurs when $\lambda = 1$. Then $u = w - t + 1$, and $m \leq (v - t + 1)/(w - t + 1)$. $\square$

We can give a nice characterization of when equality can be met in the above bound. This characterization is given in terms of certain combinatorial designs (for a general reference on design theory, we mention [3]). We need to define the concept of partitionable Steiner systems. Let $1 \leq t \leq k \leq v$. A *Steiner system* $S(t, k, v)$ is a k -uniform hypergraph $(X, \mathcal{A})$ on v points such that every t -subset of points occurs in a unique block. That is, $\mathcal{A}(t)$ consists of every t -subset of X , occurring once each. We say that the Steiner system is *partitionable* if we can partition the block set $\mathcal{A}$ into sets $\mathcal{A}_1, \dots, \mathcal{A}_j$ (where $j = (v - t + 1)/(k - t + 1)$) in such a way that each $(X, \mathcal{A}_i)$ ($1 \leq i \leq j$) is a Steiner system $S(t - 1, k, v)$. We prove the following theorem.

THEOREM 2.4. $m(t, w, v) = (v - t + 1)/(w - t + 1)$ if and only if there exists a Steiner system $S(t, w, v)$ that can be partitioned into Steiner systems $S(t - 1, w, v)$.

Proof. First, suppose that we have an $S(t, w, v)$ that can be partitioned into $S(t - 1, w, v)$. The number of these $S(t - 1, w, v)$ is $(v - t + 1)/(w - t + 1)$, and they are t -compatible, so $m(t, w, v) = (v - t + 1)/(w - t + 1)$.

Conversely, suppose $m(t, w, v) = m = (v - t + 1)/(w - t + 1)$. Let $\mathcal{S}$ denote the multi-set of induced $(t - 1)$ -subsets of t -compatible w -uniform hypergraphs $\mathcal{A}_1, \dots, \mathcal{A}_m$. Let S be any $(t - 1)$ -subset in $\mathcal{S}$. Then, S has multiplicity one. Also, since $m = (v - t + 1)/(w - t + 1)$, we see that every t -subset of the form $S' = S \cup \{x\}$ occurs exactly once in $\bigcup_{1 \leq i \leq m} \mathcal{A}_i(t)$.

We now prove that any t -subset of points S'' occurs as an induced t -subset in $\bigcup_{1 \leq i \leq m} \mathcal{A}_i(t)$. Fix some t -subset $S' = S \cup \{x\}$. We prove that S'' occurs as an induced t -subset by reverse induction on $|S' \cap S''|$. As an inductive hypothesis, suppose that S'' occurs as an induced t -subset exactly once if $|S' \cap S''| \geq t - j$. Clearly, we can start the induction at $j = 0$.

Now, suppose that $|S' \cap S''| = t - j - 1$. Let $y \in S' \setminus S''$ and let $z \in S'' \setminus S'$. Define $S^* = S'' \cup \{y\} \setminus \{z\}$. Then, $|S' \cap S^*| = t - j$, so, by the induction assumption, S^* occurs exactly once as an induced t -subset. Now, $S^* \setminus \{y\}$ has cardinality $t - 1$, and is a member of $\mathcal{S}$. Hence, $S^* \setminus \{y\} \cup \{z\} = S''$ occurs exactly once as an induced t -subset in $\bigcup_{1 \leq i \leq m} \mathcal{A}_i(t)$.

So, we have proved that $\bigcup_{1 \leq i \leq m} \mathcal{A}_i$ is an $S(t, w, v)$. Each induced $(t - 1)$ -subset occurs once in each $\mathcal{A}_i(t - 1)$, so each of $\mathcal{A}_1, \dots, \mathcal{A}_m$ is an $S(t - 1, w, v)$. This completes the proof. $\square$

We define an *optimal* (t, w, v) -threshold scheme to be a perfect (t, w) -threshold scheme having v shadows and $(v - t + 1)/(w - t + 1)$ keys. Unfortunately, there are not too many examples of partitionable Steiner systems known, so optimal threshold schemes are difficult to construct. We present two infinite classes in § 3.

We finish this section by presenting an example of a well-known threshold system in this combinatorial setting. The scheme we discuss is the Shamir threshold scheme [9].

A prime number $p > w$ is chosen, and the key can be any integer $k \in \mathbb{Z}_p$ (so $m = p$). The set of shadows $X = \{(x, y) \in \mathbb{Z}_w \times \mathbb{Z}_p, 1 \leq x \leq w\}$ (so $v = pw$). Now, for every polynomial $h(x) \in \mathbb{Z}_p[x]$ having degree at most $t - 1$, we construct a block $B(h)$ as follows. The shadows in $B(h)$ are $(u, h(u))$, $1 \leq u \leq w$, and the key for $B(h)$ is $h(0)$. Hence, the number of blocks $b = p^t$.

This scheme is perfect. It is not difficult to see that any t -subset of shadows determine $h(x)$, and hence $k = h(0)$, uniquely, by means of Lagrange interpolation. Define a subset of shadows to be a *transversal* if no x -coordinate is repeated. Then it is easy to see, for each $\mathcal{A} = \phi^{-1}(k)$ ($k = 0, \dots, p-1$), that $\mathcal{A}(t-1) = \{\text{every transversal of size } t-1, \text{ once each}\}$. Hence, no $(t-1)$ -subset gives any information as to the value of k .

It is also interesting to compare the number of keys to the bound on $m(t, w, v)$. The Shamir scheme has $v = pw$ and $m = p$, so $m = v/w$. The upper bound on m given in Corollary 2.3 is

$$(v-t+1)/(w-t+1).$$

Hence, for large values of v , the number of keys in the Shamir scheme is less than optimal by a factor of about

$$(w-t+1)/w.$$

3. Constructions for two infinite classes of optimal threshold schemes. In this section, we present constructions for two classes of optimal threshold schemes with $t = 3$, and discuss their implementation. Our first construction is based on partitioning the set of all triples (which is an $S(3, 3, v)$) into Steiner triple systems (or $S(2, 3, v)$).

Construction 1. Suppose $p \equiv 7 \pmod{8}$ is a prime. Then there exists an optimal $(3, 3, p+2)$ -threshold scheme.

We describe a partition of the Steiner system $S(3, 3, p+2)$ into p Steiner systems $S(2, 3, p+2)$, due to Wilson [10]. Let $\mathbf{X} = \text{GF}(p) \cup \{\infty, \infty'\}$. Define $\mathcal{A}_0$ to consist of the following blocks:

one block: $\{\infty, \infty', 0\}$;

$(p^2 - 3p + 2)/6$ blocks: $\{a, b, c\}$, where $a \neq b \neq c \neq a$, and $a + b + c = 0$;

$(p-1)/2$ blocks: $\{\infty, a, -2a\}$, where a is a quadratic residue in $\text{GF}(p)$; and

$(p-1)/2$ blocks: $\{\infty', a, -2a\}$, where a is a quadratic nonresidue in $\text{GF}(p)$.

Then $\mathcal{A}_0$ is an $S(2, 3, p+2)$. Define $\infty + i = \infty$ and $\infty' + i = \infty'$, for any $i \in \text{GF}(p)$. Now, for any $i \in \text{GF}(p)$, define

$$\mathcal{A}_i = \{\{x+i, y+i, z+i\} : \{x, y, z\} \in \mathcal{A}_0\}.$$

It is not difficult to show that $\cup_{0 \leq i < p} \mathcal{A}_i$ is an $S(3, 3, p+2)$; hence we have the desired optimal threshold scheme.

Example. Suppose $p = 7$; then $\mathcal{A}_0$ contains the following blocks:

$\{\infty, \infty', 0\}, \{0, 1, 6\}, \{0, 2, 5\}, \{0, 3, 4\}, \{1, 2, 4\}, \{3, 5, 6\}, \{\infty, 1, 5\}, \{\infty, 2, 3\},$

$\{\infty, 4, 6\}, \{\infty', 3, 1\}, \{\infty', 6, 2\}, \{\infty', 5, 4\}.$

Let us now briefly consider implementing such a scheme. We could take p to be some large prime, having 50 digits, for example. A key is any element i of $\text{GF}(p)$. To determine shadows, it is necessary only to generate a random block of $\mathcal{A}_0$ and add i to each element. Observe that we can easily generate a random block of $\mathcal{A}_0$ from the recipe given above. It is unnecessary to construct any blocks ahead of time.

Given a block $\{x, y, z\}$, how is the key computed? Again, this is not difficult, if we consider the various possibilities. If $\{x, y, z\} = \{\infty, \infty', i\}$, then the key is i . If $\infty, \infty' \notin \{x, y, z\}$, then the key is $(x + y + z)/3$. If the block is $\{\infty, x, y\}$, we proceed as follows. We know that $\{x, y\} = \{a + i, -2a + i\}$, where i is the key and a is a quadratic

residue. Hence, $a = \pm(x - y)/3$. Thus, we calculate $(x - y)/3$, and test to see if it is a quadratic residue. We can do this easily in time $O(\log p)$. If $(x - y)/3$ is a quadratic residue, then $a = (x - y)/3$ and $i = x - a$; otherwise, $a = (y - x)/3$ and $i = y - a$. Finally, if the block is $\{\infty', x, y\}$, the calculations are similar.

Hence, the key is calculated in time $O(1)$, unless the given block contains exactly one of ∞, ∞' , in which case the calculation requires time $O(\log p)$. However, this second situation occurs with probability $O(1/p)$, so on average, the calculation requires time $O(1)$. As well, we observe that the only arithmetic operations required (other than testing quadratic residuosity) are a small number of addition or subtraction operations, and dividing by three. The multiplicative inverse of three can be calculated ahead of time, so only a single multiplication operation would be required during key calculation.

We also observe that the number of keys, $v - 2$, is roughly three times the number of keys, $v/3$, in the Shamir scheme when $t = w = 3$.

Our second construction is a partition of the planes of the affine geometry $AG(2m, 2)$ (which form an $S(3, 4, 2^{2m})$) into Steiner systems $S(2, 4, 2^{2m})$. This construction is due to Baker [2].

Construction 2. For every integer $m \geq 1$, there exists an optimal $(3, 4, 2^{2m})$ -threshold scheme.

We construct a Steiner system $S(3, 4, 2^{2m})$ which can be partitioned into $2^{2m-1} - 1$ Steiner systems $S(2, 4, 2^{2m})$.

Let $X = GF(2^{2m-1}) \times GF(2)$. Define $\mathcal{A}$ to consist of the planes of the affine geometry $AG(2m, 2)$, as follows. The blocks are all sets $\{(a, i), (b, j), (c, k), (d, \ell)\}$ where $a + b + c + d = 0$ (in $GF(2^{2m-1})$) and $i + j + k + \ell = 0$ (in $GF(2)$).

Clearly, every block has the form

$$\{(a, 0), (b, 0), (c, 0), (d, 0)\}, \{(a, 1), (b, 1), (c, 1), (d, 1)\},$$

$$\text{or } \{(a, 0), (b, 0), (c, 1), (d, 1)\}.$$

Define a function $K: \mathcal{A} \rightarrow GF(2^{2m-1}) \setminus \{0\}$ as follows:

$$K(A) = (a^3 + b^3 + c^3 + d^3)^{1/3} \quad \text{if } A = \{(a, 0), (b, 0), (c, 0), (d, 0)\},$$

$$K(A) = (a^3 + b^3 + c^3 + d^3)^{1/3} \quad \text{if } A = \{(a, 1), (b, 1), (c, 1), (d, 1)\},$$

$$K(A) = (a^3 + b^3 + c^3 + d^3 + (c + d)^3)^{1/3} \quad \text{if } A = \{(a, 0), (b, 0), (c, 1), (d, 1)\}.$$

Then, define $\mathcal{A}_x = \{A: K(A) = x\}$ ($x \in GF(2^{2m-1}) \setminus \{0\}$). It can be proved that each $\mathcal{A}_x$ is an $S(2, 4, 2^{2m})$ (see [2]).

Key calculation is accomplished as follows. We observe that, since

$$\text{GCD}(2^{2m-1} - 1, 3) = 1,$$

there exists a (unique) multiplicative inverse of three (mod $2^{2m-1} - 1$). Denote this number by t . Then, $x^{1/3} = x^t$, for any $x \in GF(2^{2m-1})$.

Let us also consider how to generate a random block in $\mathcal{A}_x$. This is made easier by the observation that, for any $\omega \in GF(2^{2m-1}) \setminus \{0\}$, $K(\omega A) = \omega \cdot K(A)$, where ωA is defined to be the block

$$\{(\omega a, i), (\omega b, j), (\omega c, k), (\omega d, \ell)\} \text{ (where } A = \{(a, i), (b, j), (c, k), (d, \ell)\} \text{)}.$$

Hence, if we generate any random block $A \in \mathcal{A}$, we can then obtain a random block in $\mathcal{A}_x$ by multiplying A by $x \cdot K(A)^{-1}$.

In this scheme, the number of keys, $(v - 2)/2$, is about two times the number of keys, $v/4$, in the Shamir $(3, 4)$ -scheme.

4. Conclusion. The purpose of this paper is to give a description of threshold schemes from a combinatorial viewpoint, as opposed to the traditional linear algebraic approach. It is our hope that this more general framework will lead to new constructions for threshold schemes, by using techniques from combinatorial design theory. Using combinatorial designs, we give constructions for two infinite classes of threshold schemes which allow more keys to be handled (for the relevant parameters) than in previously known threshold schemes.

REFERENCES

- [1] M. ABADI, J. FEIGENBAUM, AND J. KILIAN, *On hiding information from an oracle*, preprint.
- [2] R. D. BAKER, *Partitioning the planes of $AG_{2m}(2)$ into 2-designs*, Discrete Math., 15 (1976), pp. 205–211.
- [3] TH. BETH, D. JUNGnickel, AND H. LENZ, *Design Theory*, Bibliographisches Institut, Zurich, 1985.
- [4] A. BEUTELSPACHER, *Geometric structures as threshold schemes*, preprint.
- [5] G. R. BLAKELY, *Safeguarding cryptographic keys*, Proc. N.C.C., AFIPS Conference Proceedings, 48 (1979), pp. 313–317.
- [6] G. R. BLAKELY AND C. MEADOWS, *Security of Ramp-Schemes*, Lecture Notes in Computer Science 196, Springer-Verlag, New York, 1985, pp. 242–268.
- [7] I. HOLYER, *The NP-completeness of edge-colouring*, SIAM J. Comput., 10 (1981), pp. 718–720.
- [8] S. C. KOTHARI, *Generalized Linear Threshold Scheme*, Lecture Notes in Computer Science 196, Springer-Verlag, New York, 1985, pp. 231–241.
- [9] A. SHAMIR, *How to share a secret*, Comm. ACM, 22 (1979), pp. 612–613.
- [10] R. M. WILSON, *Some Partitions of All Triples into Steiner Triple Systems*, Lecture Notes in Mathematics 411, Springer-Verlag, New York, 1974, pp. 267–277.

THE UNION OF MATROIDS AND THE RIGIDITY OF FRAMEWORKS*

WALTER WHITELEY†

Abstract. From the pattern of its rigidity matrix, we show that a k -frame on a graph (or multigraph) has the matroid structure of the union of k copies of the cycle matroid of the graph. This matrix pattern is applied to three central results about the rigidity of frameworks. An immediate corollary of this matroid union is a characterization of rigid bar and body frameworks in n -space (Tay's Theorem). This is further specialized to characterize the independence and the rigidity of body and hinge structures in n -space (a new theorem). The two-frame, or union of two copies of the graphic matroid, is truncated to produce plane bar and joint frameworks giving a characterization of minimal infinitesimally rigid bar and joint frameworks in the plane (Laman's Theorem). Finally, these techniques are used to characterize the graphs of infinitesimally rigid frameworks on other surfaces, such as the flat torus, the cylinder, cones, etc., using matroid unions of cycle and bicycle matroids of the graph.

Key words. infinitesimally rigid frameworks, matroid union, graphic matroid, disjoint spanning trees

AMS(MOS) subject classifications. primary 05B35, 05C05, 51M30, 70B99; secondary 73K99, 52A45

1. Introduction. Unions of represented matroids create recognizable matrix patterns using variable or indeterminate entries. In particular, unions of graphic and transversal matroids have a distinctive form. We will show that all of the current results, and several new results, which characterize the graphs of minimal infinitesimally rigid structures can be observed, and proven, from the pattern of their representative "rigidity matrices" and the connection to unions of the cycle matroids of a graph.

In § 2 we describe a matrix, called the general k -frame, which represents the union of k copies of the cycle matroid of a graph (or multigraph). We use this characterization to give simple combinatorial characterizations of the independent sets of rows.

In § 3 we apply the k -frame to the study of bar and body frameworks in n -space, setting $k = n(n + 1)/2$. The characterization as a union of cycle matroids provides a new, simpler proof of a recent theorem of Tay [19], [20]:

G is the graph of an independent bar and body framework in n -space if and only if G is the union of $k = n(n + 1)/2$ forests.

Our techniques, including specialization to an irreducible variety, are further exploited to verify the conjecture of a structural engineer, Janos Baracs, that characterizes independent body and hinge structures in space:

G is the graph of an independent body and hinge framework in three-space if and only if the multigraph G^* , with edges of G multiplied five-fold, is the union of six forests.

This extension, and its generalization to n -space, were independently obtained by Tay and the author [21].

Beyond the unification and simplification which flows from this approach, we are motivated by a series of still unsolved problems of a similar nature. In areas such as scene

* Received by the editors November 5, 1986; accepted for publication October 21, 1987. This research was supported in part by the Fonds pour la formation des chercheurs et l'aide à la recherche (FCAR) of Quebec and the Natural Sciences and Engineering Research Council of Canada under research grants. Preparation of the manuscript was also assisted by a visiting appointment at McGill University, Montreal, Quebec, Canada.

† Champlain Regional College, St. Lambert, Quebec, Canada J4P 3P2.

analysis, geodesy, and computer aided design, we find matroids formed by the rows of the Jacobian of some transformation from R^n to R^m . Equivalently, the rows are the gradients ∇g_i used to examine the solutions of a system of equations $g_i(X) = b_i$. If the functions g_i are quadratic polynomials, then the gradients create a matrix pattern of zeros and variables (see §§ 5, 6 and [18], [21], [27], [30], [31] for examples). The fundamental problem is to characterize the rank and the independence of these rows for general values of the variables.

Typically, the unsolved problems about systems of engineering equations involve more specialized patterns of indeterminates, in which a variable occurs in several rows of the matrix. For such specializations of the underlying matroid, the task is either to prove that the independence structure of the rows is unchanged, or to trace the exact changes in rank which result from this weak map of the matroids. In § 4 we illustrate this possibility by exploring a truncation of a two-frame (union of two cycle matroids) which creates the rigidity matrix of a bar and joint framework in the plane. This analysis provides a new direct proof of Laman's Theorem [10]:

G is the graph of an independent plane framework if and only if $|E'| \leq 2|V'| - 3$ for all nonempty subsets E' of edges.

In § 5 we apply the basic techniques to the study of frameworks on other surfaces, such as the cylinder, the cone, and the flat torus in n -space. Our original motivation for these results was the exploration of other matroid unions as the description of frameworks. However recent work of Connelly has connected the infinitesimal rigidity of frameworks on the flat torus with local maximally dense periodic sphere packing in n -space [3]. Our results answer natural questions which arise in this context.

2. Some matroids on the edges of a graph. We will present a family of matrices, with indeterminates, which represent unions of simple matroids on the edges of a multigraph.

DEFINITION 2.1. A *multigraph* $G = (V, E)$ (or graph for short) is a set of vertices V and edges E , together with incidence functions $\alpha_1, \alpha_2: E \rightarrow V$. We speak of $\alpha_1(e)$, $\alpha_2(e)$ as the ends or vertices of the edge. We allow more than one edge to share the same ends, but will not allow loops (edges with $\alpha_1(e) = \alpha_2(e)$).

2.1. k -frames as unions of cycle matroids. We begin with a matrix pattern which arises in the study of frameworks and scene analysis [20], [26], [29].

DEFINITION 2.2. If the vertices and edges of a multigraph G are indexed by $1 \leq i \leq |E|$ and $1 \leq g \leq |V|$, then a *general k -frame* on the graph G is an E by $k|V|$ matrix $F(G, X)$ with the following entries:

$$f_{ih} = \begin{cases} x_{ij} & \text{if } k(g-1) + j = h, 1 \leq j \leq k, \text{ and } \alpha_1(e_i) = v_g, \\ -x_{ij} & \text{if } k(g-1) + j = h, 1 \leq j \leq k, \text{ and } \alpha_2(e_i) = v_g, \\ 0 & \text{otherwise,} \end{cases}$$

where x_{ij} are distinct indeterminates over the rationals.

A set of edges is *independent* in the *k -frame matroid* of the graph if the corresponding rows of the matrix $F(G, X)$ are independent.

For example, for an edge e_i joining v_1 to v_2 in a graph the row of such a general k -frame is

$$[x_{i1} \ x_{i2} \ \cdots \ x_{ik} \ -x_{i1} \ -x_{i2} \ \cdots \ -x_{ik} \ 0 \ 0 \ \cdots \ 0].$$

In § 3 we assign a vector of real numbers $b_i = (b_{i1}, \dots, b_{ik})$ to each edge e_i and then place b_i in the k columns for $\alpha_1(e_i)$ and $-b_i$ in the k columns for $\alpha_2(e_i)$, with zeros elsewhere. This specialization $G(b)$ of the general k -frame gives the k -frame on G at $b = (b_{11}, \dots, b_{vk})$. Such a k -frame is a realization of the edges of the graph in R^n . It is not a realization of the vertices of the graph in R^n , which we call a k -framework. Each such k -framework induces a k -frame where b_i is the k -vector joining the point for $\alpha_1(e_i)$ to the point for $\alpha_2(e_i)$. However, most k -frames do not define k -frameworks. This relationship will be explored in more detail in § 4.

If we take $k = 1$, the matrix we defined is, up to multiplication of each row by an indeterminate, the standard matrix for the graphic matroid [23, p. 149]: each edge gives a row with 1 under its first vertex and -1 under its second vertex. A set of edges (and their corresponding rows) are *dependent* in the *graphic matroid* if and only if the edges contain a cycle (a polygon) in the undirected graph. A set is independent if and only if the edges form a forest.

For a larger k , the set of all first columns (or j th columns) under the vertices in the matrix $F(G, X)$ is the matrix for the cycle matroid. Such repeated copies of this pattern (with different variables) correspond to a *matroid union*. (We recall that a set is independent in the matroid union $\vee M^i$ of a collection of matroids M^i if and only if the set is the union of sets E^i , with E^i independent in M^i .) The following is a particular case of this general principle for representing matroid unions [2, Prop. 7.14].

THEOREM 1. *A set of edges E is independent in the general k -frame on G if and only if it is the union of k forests (independent sets in the cycle matroid).*

Proof. We reorder the columns F_i of the matrix $F(G, X)$ of the general k -frame, grouping all first column under the vertices as the first block B^1 , etc. to create k blocks B^j ,

$$F' = [B^1 \cdots B^k] = [F_1, F_{k+1}, \dots, F_{k(v-1)+1}, F_2, F_{k+2}, \dots, F_{kv}].$$

If the rows for a set of edges are independent, then there is a nonzero minor. This minor is a polynomial in the given variables, so some monomial is nonzero. If we set the variables of this monomial equal to one, and all other variables equal to zero, this gives a matrix which will identify the k forests. The j th forest is the set of edges E^j with nonzero entries in B^j . Since each row is nonzero under at most one block, these rows for E^i are also independent in the submatrix of the block—and in the cycle matroid. The entire edge set is the union of k edge-disjoint forests.

Conversely, assume that the edge set is the union of k edge-disjoint forests. We set the j th variable for the vertices of $e_i = 1$ if edge e_i is in the j th forest, and zero otherwise. This reduces the matrix to k nonzero sections, each nonzero under only on block B^j . Since each section is the matrix for a forest in the cycle matroid, the rows are independent in this block. Since these blocks are disjoint in their columns, the union of these rows is independent in F' —and in $F(G, X)$. $\square$

2.2. Counts for independent sets. In the study of rigid frameworks it is traditional to characterize the independent sets of edges by a counting property. For example, we know that an independent set in the cycle matroid of a graph is a forest with $|E'| \leq |V'| - 1$ for all nonempty subsets E' of edges. This count is equivalent to defining the cycle matroid by the integer-valued, nondecreasing submodular function $g(E) = |V| - 1$ which is nonnegative on nonempty sets [6], [15]. Given such a submodular function g , a set E is independent if and only if $|E'| \leq g(E')$ for all nonempty subsets E' . It is also well known that the union of matroids M_i defined by integer-valued, nonnegative, non-

decreasing submodular functions g_i is defined by the submodular function $\sum g_i$ [15]. We will use the following version of this result.

THEOREM 2. *If the matroids M_i on S are defined by the integer-valued, nonnegative, nondecreasing, submodular functions g_i , then the independent subsets A of the union M_i are defined by*

$$|A'| \leq \sum g_i(A') \quad \text{for all subsets } A' \text{ of } A.$$

Proof. By the theorem of Pym and Perfect [15], the union of the matroids is defined by the integer-valued, nonnegative, nondecreasing submodular function $g = \sum g_i$. In the matroid defined by this submodular function, the independent sets are defined by the inequality $|B| \leq g(B)$ for all subsets, as required. $\square$

COROLLARY 3. *For a multigraph G the following are equivalent:*

- (i) *A k -frame on the graph G at a point b is independent, for some $b \in R^{k|V|}$.*
- (ii) *G is the union of k edge-disjoint forests.*
- (iii) *$|E'| \leq k|V'| - k$ for every nonempty subset E' of G .*

Proof. (i) $\rightarrow$ (ii). If the k -frame on G at b is independent, then we know that the general k -frame on G is also independent. We also know, from Theorem 1, that G is the matroid union of k independent sets in the cycle matroid of G . These independent sets are the required forests.

(ii) $\rightarrow$ (iii). If the graph is the union of k edge-disjoint forests, then it is clear that $E = k|V| - k$, as each forest satisfies $|E_i| \leq |V| - 1$. Since all subgraphs of G are also the union of the subforests, we have (iii).

(iii) $\rightarrow$ (i). Assume the graph satisfies the count $|E'| \leq k|V'| - k$ for every nonempty set E' of G . This function does not satisfy the assumption of Theorem 2. We will switch to the nonnegative rank function $g(E') = |V'| - c(E')$, where $c(E)$ is the number of connected components in the graph of E ($c(\emptyset) = 0$). If E' has c components E_j , each with $|V_j|$ vertices, then $|E_j| \leq k|V_j| - k$ for each j , so $h(E') = k(\sum |V_j|) - kc = k|V'| - kc$. Therefore, the graph satisfies $|E'| \leq k|V'| - kc = g(E')$ for all subgraphs. By Theorem 2, this is the union of k copies of the cycle matroid defined by g , and by Theorem 1, the general k -frame on G is independent. This means that some minor on these rows is a nonzero polynomial—so some (almost all) sets of real numbers make these rows independent. $\square$

Remark. The equivalence between a graph being covered by k edge-disjoint forests, and the graph having the indicated counts, is a theorem of Tutte and of Nash-Williams

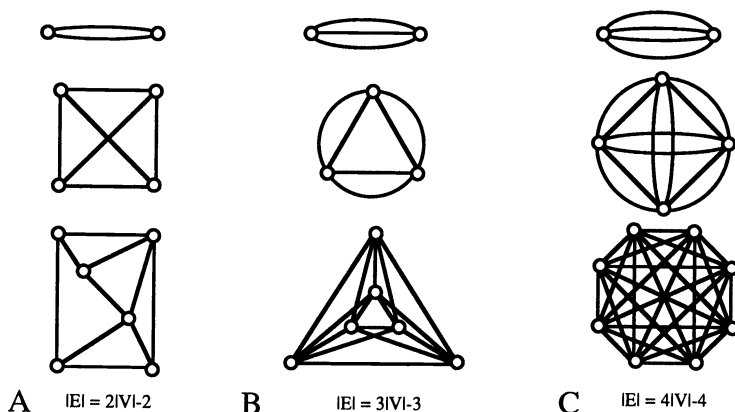


FIG. 1

[13], [14], [22]. The first proof using matroid unions of k cycle matroids of G (i.e., a k -frame) is by Edmonds [5]. Our proof is related to the proof of Welsh [23, p.127].

Figure 1 shows some examples of graphs of isostatic k -frames.

2.3. Motions of k -frames. We will prepare for later results by giving some additional results for these k -frames [26], [29].

DEFINITION 2.3. A *motion* of a k -frame $G(b)$ is an assignment of a k -vector m_i to each vertex v_i such that, for each edge e joining vertices i and j ,

$$(b_e) \cdot (m_i - m_j) = 0.$$

A motion is *trivial* if each vertex is assigned the same vector m_i . Otherwise the motion is *nontrivial*. A k -frame is *rigid* if it has only the trivial motions and *isostatic* if it is rigid and independent (i.e., every subframework formed by deleting edges has nontrivial motions).

The *rigidity matrix* which records the system of edge equations for these motions is a specialization $F(G, b)$ of the general k -frame matrix $F(G, X)$. The trivial motions form a vector space of dimension k in the solution space of $F(G, b) \times Z = 0$. Therefore a k -frame is isostatic if and only if the corresponding general k -frame $F(G, X)$ is independent with $|E| = k|V| - k$, and the rows (or *bars*) remain independent under the given specialization. If a k -frame preserves the independence structure of the rows of the general k -frame on the same graph, we say that the k -frame is *in general position*.

The submodular functions also characterize the rigid k -frames on a set of vertices. The following is a special case of a general result for submodular functions [23], but we give a direct proof for k -frames.

DEFINITION 2.4. Given a graph G and a partition of the vertices $P = (V^1, V^2, \dots, V^p)$, the contracted graph $G^* = (V^*, E^*)$ is formed by identifying all vertices of a class V^h , and removing all loops (with their two end points now identified).

THEOREM 4. *Almost all k -frames on G are rigid if and only if $|E^*| \geq k|V^*| - k$ for all contracted subgraphs of a partition P of the vertices.*

Proof. If the graph has some rigid k -frame $G(b)$, then this contains an independent set of edges E' of size $k|V| - k$. If some partition P induces a graph with $|E^*| < k|V^*| - k$, then the induced framework $G'(b)$ has a nontrivial motion. If we assign to all bodies in $G(b)$ the same centers used in $G'(b)$, we have a nontrivial infinitesimal motion of $G(b)$. Thus the count is necessary for rigidity.

Assume that a general position k -frame has nontrivial motions. All vertices with the same vector m_i in this motion are placed together in a partition of the vertices, and we form the contracted graph G^* . Take the smallest nonempty subset of edges E' in E^* with $|E'| = k|V'| - k$. This will have $|E''| \leq k|V''| - k$ for all nonempty subsets, so this gives a rigid k -frame. All vertices in E' must belong to the same block of the partition. Since the nontrivial motion requires at least two blocks, we conclude that $|E^*| < k|V^*| - k$. Thus the count is sufficient for rigidity. $\square$

We need one further result about motions of these k -frames.

THEOREM 5. *For a multigraph $G = (V, E)$ the following are equivalent:*

(i) *Some general position k -frame on the graph has independent bars and a space of motions of exactly dimension $k + 1$ and any nontrivial motion gives distinct vectors m_i to distinct adjacent vertices of the graph.*

(ii) *If any edge e of the graph is doubled, with e' , the new multigraph G_e is the union of k edge-disjoint spanning trees.*

(iii) $|E| = k|V| - k - 1$ and for any proper subset, $|E'| \leq k|V'| - k - 1$.

Proof. (i) $\rightarrow$ (iii). Since the bars are independent and the motion space has dimension $k + 1$, we know that $|E| = k|V| - k - 1$. For any subgraph with at least one edge, the

motion space has dimension $\geq k + 1$, and the bars are independent, so $|E'| \leq k|V'| - k - 1$.

(iii) $\rightarrow$ (ii). Assuming (iii) we will find that each G_e has $|E_e| = k|V| - k$ and $|E'_e| \leq k|V'| - k$. As noted in Corollary 9, this count is equivalent to the covering of G_e by k edge-disjoint spanning trees.

(ii) $\rightarrow$ (i). Since the general k -frame on G_e is independent (Theorem 8), a general position k -frame on G_e is independent with $|E_e| = k|V_e| - k$. With e removed this leaves an independent k -frame with a nontrivial motion. Since the extra edge e' is independent, it must block this motion. Thus the motion must assign distinct m_i to the two ends of e' and of e . This is true for all edges e , so some k -frame on G will have the required nontrivial motion. $\square$

Remark. By a simple induction, this theorem can be generalized to characterize graphs of general k -frames which give motion spaces of dimension $k + m$, for $1 \leq m < k$, in terms of the count: $|E| = k|V| - k - m$ and $|E'| \leq k|V'| - k - m$ for proper subgraphs.

3. Applications to frameworks of bodies. If $k = n(n + 1)/2$, we can think of the k -vector b_e assigned to the edge e in the k -frame as the coordinates of a projective line (or two-extensor) in n -space. However, such a k -tuple will represent a line (rather than a weighted sum of lines or a screw) if and only if the coordinates also satisfy special quadratic equations—the quadratic p -relations [7, p. 309 ff]. For example, for $n = 3$, ($k = 6$) these equations are of the form $y_1 y_4 + y_2 y_5 + y_3 y_6 = 0$ for distinct coordinate positions $y_1, \dots, y_6$. In general a *two-extensor* in n -space is a d -vector ($d = n(n + 1)/2$) which satisfies the quadratic p -relations.

DEFINITION 3.1. A *body and bar framework* $G(B)$ in n -space is a multigraph G together with an assignment B of a two-extensor in real projective n -space, $B(e) = b_e$, to each edge e . The framework $G(B)$ realizes the graph G at $B = (B(e_1), \dots, B(e_t))$.

The framework is *independent* if the corresponding d -frame on G at B has independent rows and a framework is *infinitesimally rigid* if the corresponding d -frame at B has only trivial motions.

A motion of the corresponding d -frame assigns d -vectors to each body. In n -space, a solid body moves, instantaneously, or infinitesimally, with a screw motion which is the sum of translations and rotations. Each translation or rotation has an axis which is a $(n - 1)$ -extensor (also a d -vector). The d -vector from our matrix equation is the dual coordinates of the *center* (or sum of the axes) for a screw motion [4]. In particular, a trivial motion of the k -frame, which assigns the same d -vector to all bodies, corresponds to a single screw motion, or euclidean motion, of the entire space.

We have restricted the values for the b_e by the set of p -relations, which are irreducible polynomials over any field extending the rational numbers. The collection of all these p -relations for all edges define an irreducible variety over the reals (and the rationals). We will show that such restrictions do not change the independence structure of the rows of the matrix $F(G, B)$ —for general values of describe B satisfying these restrictions.

3.1. Specializing to irreducible varieties. The natural language for this discussion comes from algebraic geometry.

DEFINITION 3.2. For any field K , and any set F of polynomials in n variables, the *variety* of F over K is

$$V(F) = \{(X_1, \dots, X_n) \in K^n \mid F_i(X_1, \dots, X_n) = 0 \forall F_i \in F\}.$$

A variety of $V(G)$ over K is *irreducible* if and only if it cannot be written as the union of a finite set of smaller varieties:

$$V(G) = \bigcup V(H_i) \rightarrow V(G) = V(H_i) \quad \text{for some } i.$$

The following proposition is a straightforward result of algebraic geometry.

PROPOSITION 6. *Given any infinite field K , any irreducible variety $V(G) \in K^n$, and a finite set of polynomials $F = \{F_i\}$ in n variables, if $V(G) - V(F_i) \neq \emptyset$ for each i then $V(G) - \bigcup V(F_i) \neq \emptyset$.*

Sketch of proof. The core of this result lies in the fact that, from an appropriate point of view, the irreducible variety will “look like” K^m for some m , and the following lemma will apply (see, for example, [9, Satz 1.3]).

LEMMA 7. *Given any infinite field K , and any finite set F of polynomials in n variables, if $K^n - V(F_i) \neq \emptyset$ for each $F_i \in F$ then $K^n - \bigcup V(F_i) \neq \emptyset$.*

Sketch of lemma. The essential idea is seen in the case $n = 1$. Nonzero polynomials in one variable can each only have a finite number of zeros, so a finite collection can only touch a finite set of numbers. The rest of the proof involves an induction on the number of variables in the polynomials. $\square$

3.2. Characterizing isostatic bar and body frameworks.

THEOREM 8. *There is an independent bar and body framework F realizing the graph G in n -space if and only if for each nonempty subset E' , $|E'| \leq d|V'| - d$ for $d = n(n+1)/2$.*

Proof. By Corollary 3, the count given is equivalent to the independence of the general k -frame on G .

The specializations of the indeterminates used in Theorem 1 gave coordinates $b_i = (0, \dots, 0, 1, 0, \dots, 0)$ for each edge of G . Such coordinates do satisfy all the quadratic p -relations. Since the p -relations for each edge e_i give an irreducible variety V_i , and the polynomials for distinct edges use distinct variables, we can conclude that $U = \bigcap V_i$, the set of allowed coordinates for a framework in n -space, is also an irreducible variety. We now apply Proposition 6 to conclude that the general k -frame is independent if and only if some bar and body framework on G at $b \in U$ is independent. $\square$

Remark. This result is a theorem of Tay [19, Thm. 4.17]. He employs a direct induction, based on specific properties of the matroid, to prove the equivalent of Theorem 8.

Our specialization of the indeterminates to elements of a nonempty algebraic variety is a new technique which applies to many other general representations of unions of matroids.

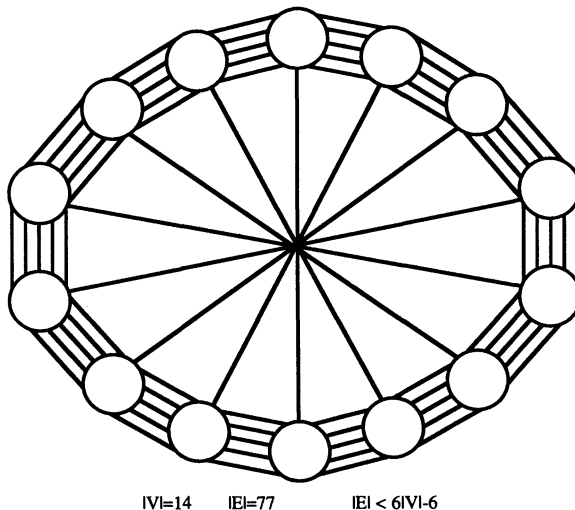


FIG. 2

COROLLARY 9. *If a graph is $2d$ -connected in an edge sense, ($d = n(n+1)/2$), then almost all bar and body frameworks in n -space on this graph are infinitesimally rigid.*

Proof. If G is $2d$ -connected, then any vertex is at least $2d$ -valent, so $|E| > d|V| - d$. Similarly, for any partition of the vertices and the induced subgraph G^* , the vertices of G^* must be $2d$ -connected, as well as $2d$ -valent, to give $|E^*| > d|V^*| - d$. We conclude that almost all frameworks on the graph are infinitesimally rigid. $\square$

Remark. This is the analogue of a theorem of Lovasz and Yemini for bar and joint frameworks in the plane (where $2d = 6$ vertex-connectivity is shown to be sufficient for infinitesimal rigidity [11]). There are counterexamples to show that $(2d - 1)$ edge-connectivity is not sufficient to guarantee infinitesimal rigidity for bar and body frameworks. Figure 2 shows such a counterexample for $n = 3$ ($2d = 12$). We note, in passing, that we can remove any d members from a $2d$ -connected graph and still guarantee rigidity.

3.3. Body and hinge frameworks. Bar and body frameworks are too general for most built structures. However there is an important type of structure which is widely used and which specializes these results one step further. The appropriate structures are those with bodies articulated along spaces of codimension two: pin joints in the plane, line-hinges in three-space, plane hinges in four-space, etc. Since these connections are the same size of vector as the “motions” found from our matrices, we obtain a nice theory.

DEFINITION 3.3. An $(n-1)$ -*extensor* is a $d = n(n+1)/2$ vector which satisfies the p -relations. (It is dual to a two-extensor.) A *screw* is a general d -vector, or a sum of $(n-1)$ -extensors.

DEFINITION 3.4. A *body and hinge structure* $G(L)$ in n -space is a multigraph $G = (V, E)$ together with a mapping L from E into the $(n-1)$ -extensors of projective n -space: $L(e) = L_e = L_{ij}$ if $\alpha_1(e) = i$ and $\alpha_2(e) = j$.

An *infinitesimal motion* of a body and hinge structure is an assignment of screw centers S_i to each vertex v_i of the graph such that for every edge (i, j) :

$$S_i - S_j = w_e L_e \text{ for some scalar } w_e \text{ if } \alpha_1(e) = i \text{ and } \alpha_2(e) = j.$$

A body and hinge framework is *infinitesimally rigid* if every infinitesimal motion is trivial, with all bodies receiving the same center.

THEOREM 10 (Tay, Whiteley). *For a multigraph G the following are equivalent:*

(i) *The graph has realizations as an infinitesimally rigid body and hinge structure in n -space.*

(ii) *The graph contains d spanning trees which use any edge at most $d-1$ times.*

(iii) *$(d-1)|E| \geq d(|V| - 1)$ and for any partition of the vertices creating a contracted subgraph G^* : $(d-1)|E^*| \geq d(|V^*| - 1)$.*

Proof. (i) $\rightarrow$ (ii) A single hinge in n -space leaves one degree of freedom between the two bodies, and thus removes $(d-1)$ degrees of freedom. Thus the hinge is equivalent to $(d-1)$ independent bars joining the two bodies, with all bars passing through the space of the hinge extensor. (For example in three-space, a hinge is equivalent to five bars, all intersecting the line of the hinge.) With this replacement, a body and hinge structure becomes a bar and body framework on a graph G' with $(d-1)$ edges for each edge of G .

If the body and hinge structure is infinitesimally rigid, then the bar and body framework is also infinitesimally rigid and must contain an independent set of $d(|V| - 1)$ edges. Thus the graph G' must contain d edge-disjoint spanning trees, so the graph G contains d spanning trees which use each edge at most $(d-1)$ times.

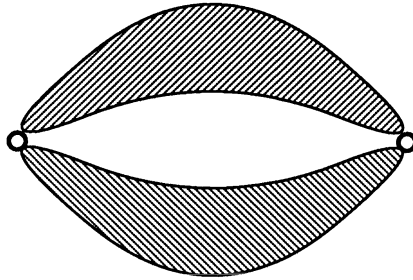


FIG. 3

(ii) $\rightarrow$ (i) Assume that the graph G' contains d edge-disjoint spanning trees. In the specialization used in Theorem 1, each tree is assigned a single vector of the form $(0, \dots, 0, 1, 0, \dots, 0)$. In projective n space, these d vectors form the d edges of a simplex with $n + 1$ vertices. Since any edge of G is matched to at most $(d - 1)$ of these vectors, there is an appropriate $n - 2$ face of the simplex which intersects all these edges. (Take the $(n - 2)$ -face opposite to a missing edge of the simplex.) This creates a specialization which assigns an $(n - 1)$ -extensor to each hinge and preserves the infinitesimal rigidity of the structure.

(ii) $\leftrightarrow$ (iii) is clearly Corollary 3. $\square$

Remark 1. For $n = 3$ this result was conjectured in 1976 by Janos Baracs, a structural engineer. The theorem was observed independently by Tay and the author [21]. In view of the essential difficulties which remain for spatial bar and joint frameworks (see below) it is a pleasant surprise that these hinge structures retain their combinatorial simplicity.

We have treated a hinge in space as a composite of five carefully chosen bars. We can also consider connections between bodies which are equivalent to two, three, or four bars. We must, however, be careful with the interpretation of these composites. For example, three concurrent bars could represent a joint between the two bodies. The obvious count to use is: $3|E| = 3|V| - 6 \leftrightarrow |E| = 2|V| - 2 \leftrightarrow$ the graph is covered by two edge-disjoint forests. (We will replace each forest by three lines.) However, if we choose three concurrent lines for the first forest, we cannot choose three concurrent lines for the second, since these six lines would be dependent. We can pick three coplanar lines for the second tree (the projective dual of three concurrent lines). Thus the count characterizes only body, joint, and "sheet" structures. From the example of two bodies with two joints, which has a nontrivial motion (Fig. 3), we know that the count $3|E| = 3|V| - 6$ does not, in fact, characterize independent sets of joints among bodies.

4. Bar and joint frameworks in the plane. As the previous example illustrates, one of the basic, still unsolved problems is to characterize the graphs which give minimal infinitesimally rigid bar and joint frameworks in three-space. However, we do have a characterization for plane frameworks. For these frameworks, multiple edges are redundant, so we will only use graphs G in this section.

DEFINITION 4.1. A *bar and joint framework* in the plane $G(P)$ is a graph $G = (V, E)$ with an assignment P of a point $p_i \in \mathbb{R}^2$ to each vertex v_i such that $p_i \neq p_j$ if $(i, j) \in E$.

An *infinitesimal motion* of a plane bar and joint framework (G, P) is an assignment M of a vector $m_i \in \mathbb{R}^2$ to each vertex v_i such that, for each edge (i, j) , we have

$$(p_i - p_j) \cdot (m_i - m_j) = 0.$$

A bar and joint framework in the plane is *infinitesimally rigid* if every infinitesimal motion is a restriction to the vertices of a Euclidean motion of the plane.

A bar and joint framework is *isostatic* if it is infinitesimally rigid and independent.

The characterization of isostatic frameworks was first derived by Laman from a specific count on G : $|E| = 2|V| - 3$ and $|E'| \leq 2|V'| - 3$, using a detailed induction [10].

With our general results on k -frames, and an appropriate geometric observation, we can now obtain this result directly from the visual pattern of the rigidity matrix and the resulting connections with two-frames.

We begin with the form of the *rigidity matrix* for a bar and joint framework in the plane. This matrix has two columns for each vertex, and a row for each edge, and records the linear equations imposed on the infinitesimal motions by the edges:

$$\begin{array}{l} \text{edge (1, 2)} \\ \text{edge (1, 3)} \\ \text{edge (i, j)} \end{array} \begin{bmatrix} v_1 & v_2 & v_3 & \cdots \\ p_1 - p_2 & p_2 - p_1 & 00 & \cdots \\ p_1 - p_3 & 00 & p_3 - p_1 & \cdots \\ & \cdots & & \\ 00 & 00 & 00 & p_i - p_j & \cdots \\ & \cdots & & & \end{bmatrix} \begin{bmatrix} m'_1 \\ \vdots \\ m'_k \\ \vdots \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \vdots \end{bmatrix}.$$

If we blur the pattern of this matrix for $F(G, P)$, we can think of $p_i - p_j$ as special values for a vector of two indeterminates (u_{ij}, w_{ij}). The matrix would then assume the pattern of the matrix for a two-frame, with the framework matrix as a specialization.

We will interpret nontrivial motions of this corresponding two-frame as descriptions of bar and joint frameworks in the plane.

PROPOSITION 11. *Given a two-frame on a graph G (not a multigraph) with a vector (a_{ij}, b_{ij}) for each edge (i, j) there are points (x_i, y_i) for the vertices v_i of G such that for each edge of the graph with*

$$(x_i - x_j, y_i - y_j) = h_{ij}(a_{ij}, b_{ij}) \quad \text{with } h_{ij} \neq 0,$$

if and only if there is a motion M of the two-frame with $m_i \neq m_j$ for each pair of adjacent vertices.

Proof. Given the points (x_i, y_i) the required motion M is the rotation of the framework about the origin which assigns $m_i = (-y_i, x_i)$. This clearly gives distinct m_i to adjacent vertices, since the points were distinct (from the assumption $h_{ij} \neq 0$) (Fig. 4).

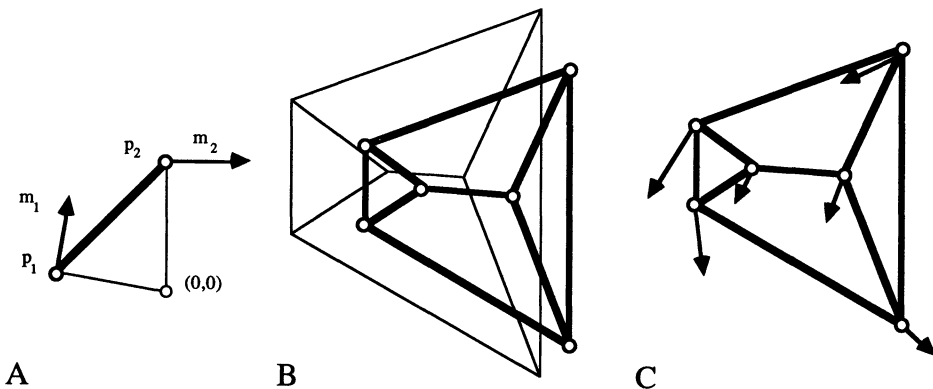


FIG. 4

Conversely, assume there is such a motion M with $m_i = (s_i, t_i)$. We define $x_i = t_i$ and $y_i = -s_i$ (so that M becomes a rotation of the corresponding framework). We know

$$(m_i - m_j) \cdot (a_{ij}, b_{ij}) = 0 \leftrightarrow (-y_i + y_j, x_i - x_j) \cdot (a_{ij}, b_{ij}) = 0 \\ \leftrightarrow (x_i - x_j, y_i - y_j) = h_{ij}(a_{ij}, b_{ij}).$$

Clearly $h_{ij} \neq 0$ if and only if the vertices are distinct, if and only if $m_i \neq m_j$. $\square$

Remarks. Any two frameworks obtained from two distinct motions will have corresponding bars parallel—each will be a *parallel redrawing* of the other. Since the motions of the two-frame are also motions of the framework, we have, in effect, shown that parallel redrawings of a framework (Fig. 4(b)) correspond to infinitesimal motions of the framework (Fig. 4(c)). This is an old engineering technique. Of course any framework has a family of trivial parallel redrawings formed by translation and dilation of the framework and these correspond to the Euclidean motions of translation and rotation about the center of dilation.

If a two-frame has only two degrees of freedom, then we will not be able to build it as a framework. We can see it as a condensed framework with all joints on one point, with edges in definite directions among these joints. Such a condensed framework will have only a two-dimensional space of Euclidean motions, since rotation will not be distinguished from translation of the single point.

For $k > 2$, the motions of a k -frame can also be interpreted as parallel redrawings of a condensed framework in k -space. Unfortunately, parallel redrawings do not correspond to general infinitesimal motions in higher dimensions. A general theory of parallel redrawing matroids is presented in [31].

THEOREM 12. (Laman). *Given a graph $G = (V, E)$, with at least two vertices, the following are equivalent:*

- (i) G can be realized as an isostatic bar and joint framework in the plane.
- (ii) $|E| = 2|V| - 3$ and, for any subgraph G' with more than one vertex, $|E'| \leq 2|V'| - 3$.
- (iii) If any edge of the graph is doubled then the resulting multigraph is the union of two edge-disjoint spanning trees.

Proof. We will prove that (i) $\leftrightarrow$ (ii), since the equivalence of (ii) and (iii) was checked in Theorem 4.

(i) $\rightarrow$ (ii). If the graph is isostatic for bar and joint frameworks in the plane, then the rows of its rigidity matrix at a general point are independent and the solution space has dimension three, so $|E| = 2|V| - 3$. When restricted to the rows on two or more vertices V' , we still have at least the trivial motions of dimension three, so $|E'| \leq 2|V'| - 3$.

(ii) $\rightarrow$ (i). If the graph satisfies the count, then by Theorem 4 a general position two-frame on G has a space of motions of dimension three, and every nontrivial motion gives distinct vectors m_i to distinct vertices. By Theorem 10, we can define a framework such that each row of its rigidity matrix is a nonzero multiple of the row in the two-frame matrix. We conclude that the framework has the same three-dimensional space of infinitesimal motions and is infinitesimally rigid. $\square$

Remark 1. The equivalence of (i) and (ii) is the original theorem of Laman, and has various proofs in the literature [10], [1]. The early proofs used a direct induction on the number of vertices in the graph.

The equivalence of (i) and (iii) is a result of Recski [16], which refines slightly a result of Lovasz and Yemini [11]. The proof of Lovasz and Yemini used techniques of submodular functions and slicing by a hyperplane, from which the matroid unions can

be teased out. The proof of Recski made explicit use of the two spanning trees, and pointed towards the essential role of matroid unions in this result.

We believe that this present proof clearly displays the essential pattern of two-frames as matroid unions of the graphic geometry, as well as the geometric meaning of the final truncation down to a framework.

Remark 2. Laman's Theorem was rediscovered by Snay and Spriggs as a special simple case of a still unsolved problem on the solvability of geodetic nets [17]: given a pattern of measured distances between points and angles between lines (pairs of points), when is this sufficient to draw a unique map of the points and lines? Laman's Theorem corresponds to the case where no angles are measured, but the general case is an example of a matrix pattern of indeterminates which has not yet been characterized. A characterization has been conjectured in [17].

As mentioned above, Lovasz and Yemini have shown that a graph which is six-connected in a vertex sense is generically rigid in the plane. Their proof is based on a more complex counting argument, built around the criterion:

A graph G has realizations as an infinitesimally rigid plane framework if and only if for every partition of the edges into nonempty subgraphs $G_1, \dots, G_k$ $\sum (2|V_i| - 3) \geq 2|V| - 3$.

The simpler criteria used for bar and body frameworks, such as the graph contains three edge-disjoint spanning trees, or any edge-connectivity criteria, are not sufficient for infinitesimal rigidity of plane bar and joint frameworks.

Lovasz and Yemini conjectured similar connectivity results: $n(n+1)$ -connectivity in n -space is sufficient for infinitesimal rigidity of general bar and joint frameworks in n -space.

5. Frameworks on other surfaces. These techniques, based on matroid unions, can be used to derive analogues of Laman's Theorem for frameworks on other surfaces. We begin with a general example which arises in the study of periodic sphere packings in n -space [3].

5.1. The flat n -torus.

DEFINITION 5.1. The *flat n -torus* T^n is the Euclidean space E^n factored by the unit lattice, so that $a = b$ if and only if $a - b = m \in \mathbb{Z}^n$. A *bar and joint framework in T^n* , $G(P, M)$, is a multigraph G , an assignment of positions p_i (in the unit cube) to the vertices and an assignment of $m_e \in \mathbb{Z}^n$ to each edge e . (This m_e gives a choice of a geodesic (image of a straight line in E^n) between the points p_i and p_j , $\alpha_1(e) = v_i$, $\alpha_2(e) = v_j$, which we will use for the bar.)

An *infinitesimal motion* of such a framework is an assignment of velocity vectors u_i to the joints p_i such that

$$(u_i - u_j) \cdot (p_i - p_j + m_{ji}) = 0 \quad \text{for each bar joining } p_i \text{ and } p_j.$$

Figure 5 shows two graphs (A, C) with corresponding frameworks on the two-torus, shown as periodic plane drawings (B, D). For the example in Fig. 5(b), $m_{ca} = (1, 0)$ and all other $m_e = (0, 0)$.

The equations for an infinitesimal motion create a rigidity matrix $R(G, P, M)$ of the following form:

$$\begin{bmatrix} p_1 - p_2 + m_{12} & p_2 - p_1 - m_{12} & 0 & 0 & 0 \\ p_1 - p_3 + m_{13} & 0 & p_3 - p_1 - m_{13} & 0 & 0 \\ 0 & 0 & 0 & p_i - p_j + m_{ij} & p_j - p_i + m_{ij} \end{bmatrix}.$$

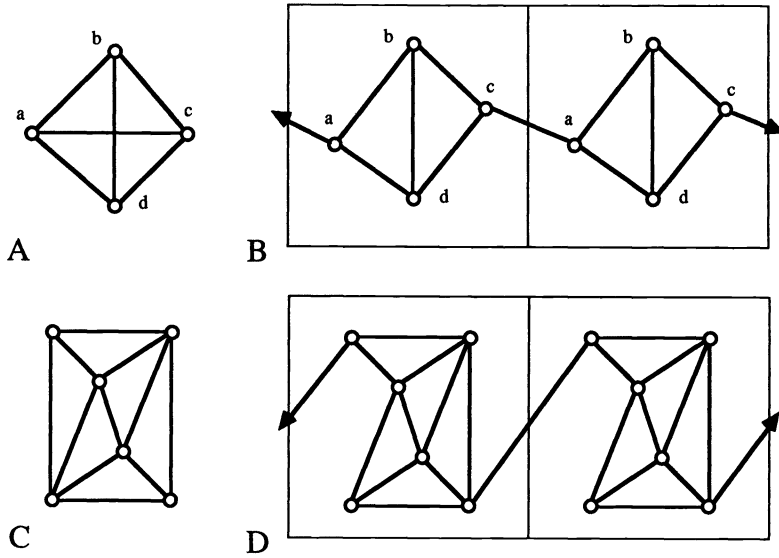


FIG. 5

Clearly each such framework, with at least one joint, has the trivial set of translations in which each joint is assigned the same velocity. If a framework in T^n has only these trivial infinitesimal motions we say it is *infinitesimally rigid*.

THEOREM 13. *For a graph G the following are equivalent:*

- (i) *Some framework in T^n , $G(P, M)$ is isostatic (minimal infinitesimally rigid),*
- (ii) *G is the union of n edge-disjoint spanning trees,*
- (iii) *G satisfies $|E| = n|V| - n$ and every subgraph G' of at least one vertex satisfies $|E'| \leq n|V'| - n$.*

Proof. We know that (ii) $\leftrightarrow$ (iii).

(ii) $\rightarrow$ (i) Assume G is the union of n edge-disjoint spanning trees. We assign all vertices the same positions p . This reduces the rigidity matrix to the matrix of an n -frame with m_{ij} in Z^n . We know that this matrix has rank $n|V| - n$, with m_{ij} chosen as the basic vectors of the form $(0, \dots, 0, 1, 0, \dots, 0)$ which are in Z^n , if and only if the graph contains n edge-disjoint spanning trees. (Just choose m_e to be one in the k th spot if e is in the k th tree and check the rank.) Thus rigidity is implied by containing n edge-disjoint spanning trees.

(i) $\rightarrow$ (ii) Any maximal independent set must satisfy $|E| = n|V| - n$ and $|E'| \leq n|V'| - n$ for all proper subgraphs, because of the trivial motions. $\square$

COROLLARY 14. *If a graph G has infinitesimally rigid realizations in T^n , some of these realizations use only n different geodesics for all the bars and use only wrapping numbers of the form $m_e = (0, \dots, 0, 1, 0, \dots, 0)$.*

No problem is caused by any of the usual dependent frameworks in E^n , since we just "twist" some of the bars onto new geodesics and wrap them around some "holes" of the torus. Note that, unlike frameworks in E^n , the results even allow multigraphs which have multiple edges between a pair of vertices.

For the study of the graphs of periodic sphere packings, etc., we would like to know more about the possible topology for the isostatic frameworks, and the corresponding choices for m_e . For example, can all "toroidal graphs" (graphs which can be realized on the torus with no crossing points, or equivalently, as periodic planar graphs) be realized

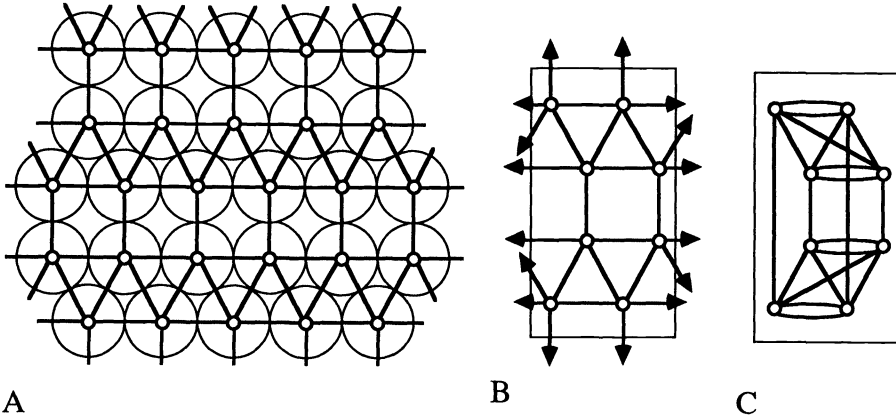


FIG. 6

as infinitesimally rigid frameworks and no crossings on the flat torus? This seems possible, since we do know from Corollary 14 that we can restrict the corresponding wrapping numbers to be similar to such toroidal graphs. Figure 6 shows an example of a periodic circle packing of the plane Fig. 6(a), and its corresponding periodic planar graph Fig. 6(b). Figure 6(c) shows the underlying graph, which gives infinitesimally rigid frameworks on the flat torus by Theorem 13.

5.2. The circular cylinder. For a general, nicely embedded surface most pairs of points on the surface can be connected by geodesics. We will use these locally shortest paths as the bars of the frameworks on the surface. Such a geodesic bar has a unique tangent vector in the tangent plane (or space) of its joints. An infinitesimal motion of the frameworks is a vector in the tangent plane of each joint such that the projections of these velocities on the tangent vectors of a geodesic bar are equal.

For example, for a cylinder in R^3 , a plane can be cut and placed onto the surface, with the geodesics of the surface being the image of the straight lines in the plane. This surface has two degrees of freedom: a rotation about the axis of the cylinder and a translation along the axis of the cylinder. This suggests the counts for isostatic frameworks:

$$|E| = 2|V| - 2 \text{ and } E' \leq 2|V'| - 2 \text{ for every nonempty subset } E'.$$

The cylinder is E^2 with an identification of points if $p - q = (z, 0)$ for an integer z . This means, as before, that two points can be joined by several different geodesic curves along which we can place the bars (Fig. 7). The geodesics joining p_i and p_j will have the form $(x_i - x_j - z_e, y_i - y_j)$.

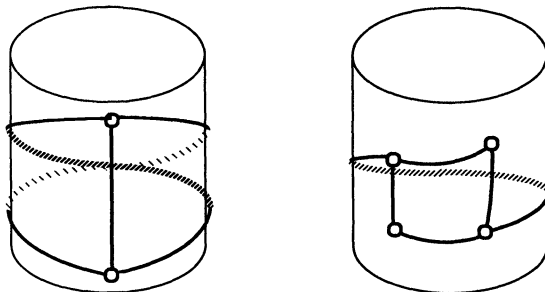


FIG. 7

DEFINITION 5.2. A *bar and joint framework on the cylinder*, $G(P, z)$ is a multigraph G , and assignment of points to the vertices $p_i = (x_i, y_i)$ ($0 \leq y_i < 1$), and integers z_e to the edges.

An *infinitesimal motion* of a bar and joint framework on the cylinder $G(P, z)$ is an assignment of velocities v_i in R^2 to the vertices, such that for each edge e joining p_i and p_j

$$(v_i - v_j) \cdot (x_i - x_j - z_e, y_i - y_j) = 0.$$

We can modify the proof of Theorem 13 to get our characterization of isostatic frameworks on the cylinder.

THEOREM 15. For a multigraph G the following are equivalent:

- (i) Some framework in general position on a cylinder is isostatic,
- (ii) G is the union of two edge-disjoint spanning trees on the vertices.
- (iii) $|E| = 2|V| - 2$ and $|E'| \leq 2|V'| - 2$ for all subsets E' .

Proof. (ii) $\rightarrow$ (i) Assume that the graph is the union of two edge-disjoint spanning trees. We place all the vertices at distinct points $(0, y_i)$ and choose $z_e = 0$ if e is in the first tree, and one for the second tree. This selection creates a nice block for the first tree under the second columns of for each vertex, as well as a set of rows which represents the second tree under the first entries with extra garbage under the second entries. It is not difficult to see that these rows are independent as well, so the rigidity matrix has rank $2|V| - 2$. Thus the framework is isostatic.

The rest of the proof carries on as before. $\square$

5.3. Cones and other surfaces. For a right circular cone, a plane can be cut and placed onto the surface, with the geodesics of the surface being the image of the straight lines in the plane. This surface has only one degree of freedom. If the angle at the apex is sharp enough, there are even geodesics which form a loop from a point back to itself (Fig. 8). Any geodesics, straight lines in the plane which wrap onto the cone, are allowed for the bars.

We can define the motions and the rigidity matrix for the cone from the plane coordinates which wrap onto the cone. We omit these extensive details. A simple count of variables, equations, and degrees of freedom for these motions suggests that:

A framework in general position is isostatic on the cone if and only if $|E| = 2|V| - 1$ and $|E'| \leq 2|V'| - 1$ for all nonempty subsets.

This count requires more than unions of cycle matroids—we need a matroid with independence characterized by the count: $|E'| \leq |V'|$. Such a matroid is well known as

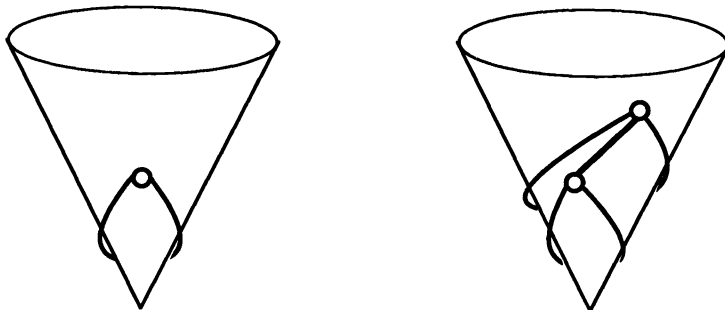


FIG. 8

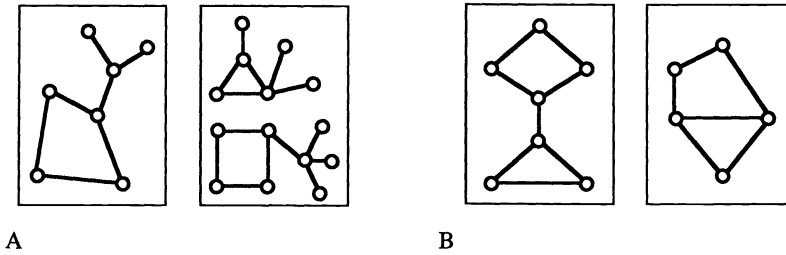


FIG. 9

the transversal matroid of a hypergraph or the bicycle matroid of a graph. We give a quick summary of some basic facts about this matroid [12].

DEFINITION 5.3. The *bicycle matroid* of a graph is defined by the independence structure: a subset E' of E is independent if and only if no connected component of the graph contains more than one cycle.

It is a simple task to check the following alternate characterization. (See Fig. 9 for examples of independent (a) and dependent sets (b).)

PROPOSITION 16. *A set E is independent in the bicycle matroid if and only if $|E'| \leq |V'|$ for all subsets of edges.*

If we take the matroid union of the bicycle matroid and the cycle matroid we get independent sets characterized by the following counts

$$|E'| \leq 2|V'| - 1 \quad \text{for all nonempty subsets } E'$$

by a simple combination of the counts for the two matroids. Clearly we know that an independent framework on the cone satisfies this count, so it must be the union of independent sets in the cycle and bicycle matroid. We can show that any set of edges which is the union of two such sets gives an independent framework on the cone. We omit the details, since they are somewhat messier than those for the cylinder or flat torus.

PROPOSITION 17. *For a graph G (without loops) the following are equivalent:*

- (i) *A framework in general position is isostatic on the cone.*
- (ii) *The graph is the disjoint union of a spanning tree and a basis in the bicycle matroid.*

- (iii) *$|E| = 2|V| - 1$ and $|E'| \leq 2|V'| - 1$ for all nonempty subsets.*

If the angle of the cone is less than 30 degrees, we can have geodesics which are loops. For these cases, Proposition 17 extends to graphs with loops.

Consider a framework on the surface of a regular tetrahedron. This surface has zero degrees of freedom. We place all joints away from the vertices of the tetrahedron and use geodesics as bars (Fig. 10). The trivial motions and a count of variables and equations (one per bar) suggest the following count:

$$|E| = 2|V| \quad \text{and} \quad |E'| \leq 2|V'| \quad \text{for all subsets } E'$$

for the isostatic frameworks.

It is a straightforward, but long, exercise to work out detailed definitions and proof for the following result.

PROPOSITION 18. *For a graph G (without loops) the following are equivalent:*

- (i) *A framework in general position is isostatic on the regular tetrahedron.*
- (ii) *The graph is the disjoint union of two bases in the bicycle matroid.*
- (iii) *$|E| = 2|V|$ and $|E'| \leq 2|V'|$ for all subsets.*

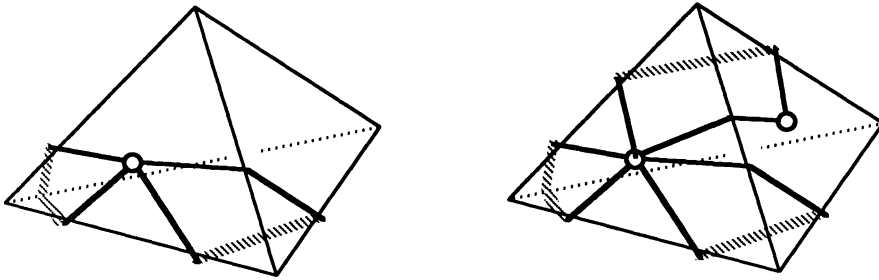


FIG. 10

For each embedded two-dimensional surface, we *conjecture* that the graphs of isostatic frameworks are described by one of four matroids $|E| = 2|V| - d$ and $|E'| \leq 2|V'| - d$ for all subsets (for a fixed d $0 \leq d \leq 3$). The choice for d will be the global degree of freedom of the surface. For example, the sphere has three degrees of freedom, and it has been shown that the frameworks on the sphere correspond to frameworks in the plane, using central projection from the center of the sphere [28].

For higher-dimensional surfaces, including E^3 , we face basic unsolved problems. However 5.1 did characterize the graphs of isostatic frameworks on the flat torus. The methods at hand indicate that the following conjecture should hold.

CONJECTURE 1. For a surface of dimension n , with degree of freedom d in E^n ($d \leq n$), and a graph G (without loops) the following are equivalent:

- (i) A framework in general position on the surface is isostatic.
- (ii) The graph is the disjoint union of d trees and $(n - d)$ bases in the bicycle matroid.
- (iii) $|E| = n|V| - d$ and $|E'| \leq n|V'| - d$ for all subsets.

6. Concluding comments.

6.1. Algorithms for isostatic frameworks. We have characterized the multigraphs of isostatic structures by counts of the following form:

$$|E'| \leq m|V'| - d \text{ for all nonempty subsets } E'.$$

While this appears to be an exponential search, over most subsets E' , these actually yield polynomial algorithms to check the graph. The work of Sugihara [18] and Imai [8] has employed bipartite matchings to give an algorithm of order $O(|V|^2)$.

This type of quick, combinatorial check provides an important first step for the verification of particular examples of built frameworks. On the other hand, in the absence of such a characterization of the graphs of isostatic frameworks in three-space, we do not even have an exponential algorithm!

6.2. Special positions. Given a graph which is combinatorially correct (i.e., has some isostatic realizations) some special values of the variables will lower the rank of the rigidity matrix and destroy the rigidity of the structure. As we noted above, the special positions are given by an algebraic variety formed by the polynomial determinants of the $|E|$ by $|E|$ minors of the rigidity matrix.

In rigidity theory the special nonrigid positions for graphs of isostatic structures (bases in the matroid over the complete graph on a set of vertices) have been intensively studied as critical forms [25], [26]. Although the rigidity matrices for bar and joint frameworks in n -space are not square, we find the very useful result that all the relevant poly-

nomials are the same (up to a scalar), and the critical forms are described by a single polynomial condition [25].

Similarly, for k -frames, and their specializations such as frameworks on the flat torus and the cylinder, a single polynomial condition defines the critical forms for a generally isostatic pattern [26].

The interweaving of the form of these polynomial conditions and the patterns of dependence (or "static stresses") and the motions at the generic critical forms have also been studied [25], [26]. We conjecture that this same pattern of single polynomial conditions and the resulting algebraic geometry of motions and stresses will also apply to frameworks on other surfaces, as well as to many related matrix structures.

Acknowledgments. We thank Henry Crapo and Neil White for numerous conversations which provoked and refined the theorems presented here. The theorems and observations of Tiong-Seng Tay, Laszlo Lovasz, and Andres Recski on isostatic frameworks [11], [16], [19] first suggested that we were studying the union of k graphic matroids, and thus stimulated our work. This paper grew out of our discovery of the new proof of Tay's Theorem in the context of joint work on the k -frames with Neil White [26]. Finally we thank Peter Russell for assistance in forming and checking the precise propositions from algebraic geometry needed in § 3.

REFERENCES

- [1] L. ASIMOW AND B. ROTH, *The rigidity of graphs II*, J. Math. Anal. Appl., 63 (1979), pp. 171–190.
- [2] T. BRYLAWSKI, *Coordinatizing matroids*, in The Theory of Matroids, Neil White, ed., Cambridge University Press, Cambridge, 1985.
- [3] R. CONNELLY, *Rigidity and Sphere Packings I, II*, Structural Topology, to appear.
- [4] H. CRAPO AND W. WHITELEY, *Statics of frameworks and motions of panels structures, a projective geometric introduction*, Structural Topology, 6 (1982), pp. 43–82.
- [5] J. EDMONDS, *Minimum partition of a matroid into independent subsets*, J. Res. Nat. Bur. Standards, 69 B (1965), pp. 67–72.
- [6] ———, *Submodular functions, matroids and certain polyhedra*, in Combinatorial Structures and Their Applications, R. Guy et al., Gordon and Breach, New York, 1970, pp. 69–87.
- [7] W. V. D. HODGE AND D. PEDOE, *Methods of Algebraic Geometry Vol. I*, Cambridge University Press, Cambridge, 1946.
- [8] H. IMAI, *On the combinatorial structure of line drawings of polyhedra*, Discrete Appl. Math., 10 (1985), pp. 79–92.
- [9] E. KUNZ, *Einführung in die kommutative Algebra und algebraische Geometrie*, F. Vieweg, Braunschweig, 1979.
- [10] G. LAMAN, *On graphs and the rigidity of plane skeletal structures*, J. Engrg. Math., 4 (1970), pp. 331–340.
- [11] L. LOVASZ AND Y. YEMINI, *On generic rigidity in the plane*, SIAM J. Algebraic Discrete Methods, 3 (1982), pp. 91–98.
- [12] L. MATHEWS, *Bicircular matroids*, Quart. J. Math. Oxford Ser. (2), 28 (1977), pp. 213–228.
- [13] C. ST. J. NASH-WILLIAMS, *Edge-disjoint spanning trees of finite graphs*, J. London Math. Soc. (2), 36 (1961), pp. 445–450.
- [14] ———, *Decompositions of finite graphs into forests*, J. London Math. Soc. (2), 39 (1964), p. 12.
- [15] J. S. PYM AND H. PERFECT, *Sub-modular functions and independence structures*, J. Math. Anal. Appl., 30 (1970), pp. 1–31.
- [16] A. RECSKI, *A network theory approach to the rigidity of skeletal structures II*, Discrete Appl. Math., 8 (1984), pp. 63–68.
- [17] J. SPRIGGS AND R. SNAY, *An algorithm for testing the solvability of horizontal networks*, NOS-NOAA, National Geodesic Survey, Rockville, MD, 1982, preprint.
- [18] K. SUGIHARA, *Detection of structural inconsistency in systems of equations with degrees of freedom*, Discrete Appl. Math., 10 (1985), pp. 297–312.

- [19] T-S. TAY, *Rigidity problems in bar and joint frameworks and linkages of rigid bodies*, Ph.D. thesis, Department of Pure Mathematics, University of Waterloo, Waterloo, Ontario, Canada, 1982.
- [20] ———, *Rigidity of multi-graphs I: linking rigid bodies in n -space*, J. Combin. Theory Ser. B, 36 (1984), pp. 95–112.
- [21] T-S. TAY AND W. WHITELEY, *Recent advances in the generic rigidity of frameworks*, Structural Topology, 9 (1983), pp. 31–38.
- [22] W. T. TUTTE, *On the problem of decomposing a graph into n connected factors*, J. London Math. Soc. (2), 36 (1961), pp. 221–230.
- [23] D. J. A. WELSH, *Matroid Theory*, Academic Press, London, 1976.
- [24] ———, *On matroid theorems of Edmonds and Rado*, J. London Math. Soc. (2), 2 (1976), pp. 251–256.
- [25] N. WHITE AND W. WHITELEY, *The algebraic geometry of stresses in frameworks*, SIAM J. Algebraic Discrete Methods, 4 (1983), pp. 481–511.
- [26] ———, *The algebraic geometry of motions in bar and body frameworks*, SIAM J. Algebraic Discrete Methods, 8 (1987), pp. 1–32.
- [27] ———, *A class of matroids defined on graphs and hypergraphs by counting properties*, Department of Mathematics, University of Florida, Gainesville, FL, 1984.
- [28] W. WHITELEY, *Cones, infinity and 1-story buildings*, Structural Topology, 8 (1983), pp. 53–70.
- [29] ———, *A correspondence between scene analysis and motions of frameworks*, Discrete Appl. Math., 9 (1984), pp. 269–295.
- [30] ———, *A matroid on hypergraphs with applications in scene analysis and geometry*, Discrete Comput. Geom., to appear.
- [31] ———, *Parallel redrawings of configurations in 3-space*, Champlain Regional College, St. Lambert, Quebec, Canada, 1986.

A FUNDAMENTAL MONOTONICITY IN GROUP TESTING*

Y. C. YAO† AND F. K. HWANG‡

Abstract. The group testing problem is to find by group tests all defectives in a given set of items each of which independently has probability p of being defective. The group testing problem originated with the famous Dorfman's blood testing problem and has since found many industrial applications including a recent one on multiple-access communication. Unfortunately, optimal group testing algorithms are not known in general. In particular, a very plausible conjecture that the expected number of tests should be nondecreasing in p for $0 \leq p < 1$ has remained open since 1981. In this paper we prove this conjecture by an innovative approach.

Key words. group testing, probability algorithm

AMS(MOS) subject classifications. 90B40, 62N10

1. Introduction. The group testing problem originated with the famous Dorfman's blood testing problem [3], [4, p. 225], [9, p. 152]. A set of n items, or blood samples, are to be classified either as good or as defective. The probability of each item being defective (good) is identically, and independently, p ($q = (1 - p)$). We call such a set a *binomial set*. Any subset of the n items can be tested in a batch, called a *group test*. The group is identified as *contaminated* if it contains at least one defective; otherwise it is identified as *pure*. The problem is to find all the defectives and to minimize the expected number of tests in doing it. Sobel and Groll [7] have illustrated many industrial applications of group testing. Recently Hayes [5] (also see [2]) applied group testing algorithms for conflict resolution in multiple-access communication.

Let $E_T(N)$ denote the expected number of tests required by an algorithm T on a set N . Define $E(N) = \min_T E_T(N)$. If N is a binomial set, we will characterize it by its two parameters n and p . Clearly $E(n, 0) = E(n, 1) = 0$ since the state of every item is known without any testing. For $(3 - \sqrt{5})/2 \leq p < 1$, Unger [8] proved $E(n, p) = n$ by showing that the algorithm of testing items one by one is optimal. He also showed that for $0 < p < (3 - \sqrt{5})/2$ the one-by-one testing is not optimal but no optimal algorithm is known for general n . Adding to this mystery is the fact that a proof is still missing for the plausible conjecture raised by Hwang, Song, and Du [6] that $E(n, p)$ is nondecreasing in p for $0 \leq p < 1$. In this paper we prove a generalized version of this conjecture. We then use the generalized version to prove a strict version of the original conjecture.

2. The new approach. Let $N = \{I_1, \dots, I_n\}$ denote a set of n stochastically independent items where I_i has probability p_i of being defective. Let $s_i = 0$ ($s_i = 1$) denote the state that I_i is good (defective). A (group) test accepts any subset S of N as input and yields one of the two outputs $\sum_{I_i \in S} s_i = 0$ (pure) and $\sum_{I_i \in S} s_i \geq 1$ (contaminated). A group testing algorithm can be represented by a binary tree T where associated with every internal node u is a test $t(u)$ with the two links of u representing the two outputs $t(u) = 0$ (pure) and $t(u) = 1$ (contaminated). Define $P(w)$ to be the path from the root of T to a node w but not including w . The length $l(w)$ of the path of w is simply the number of tests contained in $P(w)$. Associated with each terminal node v_i is the unique state vector $(s_1, \dots, s_n)$ consistent with the test outputs on the path of v_i . Let $p(v_i)$ denote the probability that v_i will be traversed. Then it is easily seen that $E_T(N) = \sum_{i=1}^{2^n} p(v_i)l(v_i)$.

* Received by the editors September 18, 1987; accepted for publication November 18, 1987.

† Colorado State University, Fort Collins, Colorado 80523.

‡ AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

LEMMA 1. Let T be a group testing tree on N . Let $G \subseteq N$ be the set of items each of which is contained in at least one test with pure output in T . Then $s_i = 0$ if $I_i \in G$ and $s_i = 1$ otherwise.

Proof. It suffices to prove $s_i = 1$ for any $I_i \notin G$, as the other part is obvious. Suppose to the contrary that $s_i = 0$ for some $I_i \notin G$. Since $s_i = 1$ is also consistent with all test outputs, the state vector is not uniquely determined, a contradiction to the assumption that T is a group testing tree.

Fix a set of $p'_1, \dots, p'_n$ with $p_i \leq p'_i < 1$ for all i . Let $X_1, \dots, X_n$ be n independent 0-1 random variables (also independent of the states of the I_i) with $P(X_i = 1) = 1 - q'_i/q_i = 1 - P(X_i = 0)$. Consider a set N' of items $I'_1, \dots, I'_n$ where $I'_i = I_i$ if $X_i = 0$ and I'_i is defective if $X_i = 1$. Clearly, N' is a set of n stochastically independent items with the probability of I'_i being defective equal to $p_i q'_i/q_i + 1 - q'_i/q_i = p'_i$.

Let T' be a group testing tree on N' . We will construct an algorithm T on N such that $E_T(N) = E_{T'}(N')$. We will use the convention that, for a given subset $t' = \{I'_i: i \in A\}$, t is defined as $\{I_i: i \in A\}$. Also, for ease of notation, we write $t = 0$ if t is a pure subset and $t = 1$ if t is a contaminated subset. Now, suppose T' first tests $t'_1 = \{I'_1, \dots, I'_k\}$. If $t'_1 = 0$, T' tests t'_2 next; if $t'_1 = 1$, T' tests t'_3 next. Let T denote a tree testing on N by imitating T' , though, in an unusual way. The first test of T is on $t_1 = \{I_1, \dots, I_k\}$. What is unusual is that the second test does not depend on the output of t_1 , but on the output of t'_1 , which we do not test. Nevertheless, based on $X_1, \dots, X_n$ and the output of t_1 , we can figure out the output of t'_1 . To see this, if t'_1 contains an item I'_i with $X_i = 1$, then clearly, $t'_1 = 1$; if t'_1 does not contain such an item, then $t'_1 = t_1$. T proceeds by testing t_2 if $t'_1 = 0$, and testing t_3 if $t'_1 = 1$. In general, suppose T' tests t'_x at the node u' , and tests t'_y if $t'_x = 0$ and tests t'_z if $t'_x = 1$. Then T will test t_x at the corresponding node u . After figuring out the output of t'_x (without testing), T tests t_y if $t'_x = 0$ and tests t_z if $t'_x = 1$. It is clear that by our construction T and T' make the same number of tests; hence $E_T(N) = E_{T'}(N')$.

LEMMA 2. T is a group testing tree on N .

Proof. Let $G \subseteq N$ denote the set of items each of which is contained in at least one test with pure output in T . Then clearly $s_i = 0$ for $I_i \in G$. We now show $s_i = 1$ for $I_i \notin G$.

Suppose to the contrary that at some terminal node v of T , $s_i = 0$ is consistent with all test outputs on $P(v)$ for some $I_i \notin G$. Since $t = 1$ implies $t' = 1$, we have $I'_i \notin G'$. Furthermore $I_j = 1$ implies $I'_j = 1$; hence at the corresponding terminal node v' of T' , $s'_i = 0$ is consistent with all test outputs on $P(v')$. By Lemma 1 $s'_i = 0$ and $I'_i \notin G'$ contradict the fact that T' is a group testing tree on N' .

We will write $E_T(N)$ by $E_T(p_1, \dots, p_n)$. Then we have proved the following.

THEOREM 1. For every algorithm T' on N' there exists an algorithm T on N such that $E_T(p_1, \dots, p_n) = E_{T'}(p'_1, \dots, p'_n)$ if $p_i \leq p'_i < 1$ for $i = 1, \dots, n$.

It is now straightforward to prove the following.

THEOREM 2. $E(p_1, \dots, p_n)$ is nondecreasing in each $p_i < 1$ for every n .

Proof. It suffices to show that $E(p_1, \dots, p_n) \leq E(p'_1, \dots, p'_n)$ if $p_i \leq p'_i < 1$ for every i . Let T' be an optimal algorithm on N' . Then

$$\begin{aligned} E(p_1, \dots, p_n) &\leq E_T(p_1, \dots, p_n) = E_{T'}(p'_1, \dots, p'_n) \quad \text{by Theorem 1} \\ &= E(p'_1, \dots, p'_n). \end{aligned}$$

COROLLARY. $E(n, p)$ is nondecreasing in $p < 1$ for every n .

3. The strict monotonicity. In this section we prove a strict version of the monotonicity conjecture.

LEMMA 3. $E(p_1, \dots, p_n) \geq E(p_1, \dots, p_{n-1}) + p_n$.

Proof. Let T be an optimal procedure for $N = \{I_1, \dots, I_n\}$ so that $E_T(N) = E(N) = E(p_1, \dots, p_n)$. Let T' be the procedure which tests I_n first and then apply an optimal procedure for $\{I_1, \dots, I_{n-1}\}$. Let T'' be the procedure which tests I_n first and then follows T except that any test in T whose output is predictable by knowing the state of I_n is skipped in T'' . Then clearly,

$$1 + E(N) \geq E_{T''}(N) \geq E_{T'}(N) \geq E(N).$$

We can tighten the first inequality by noting that I_n must be included in at least one test in T so that it can be identified; but when I_n is defective, that test can be skipped in T'' . Therefore,

$$\begin{aligned} 1 - p_n + E(N) &\geq E_{T''}(N) \\ &\geq E_{T'}(N) \\ &= 1 + E(p_1, \dots, p_{n-1}). \end{aligned}$$

Lemma 3 is proved.

LEMMA 4. Suppose that $E(p_1, \dots, p_{n-1}, p'_n) = E(p_1, \dots, p_{n-1}, p''_n)$ for some $0 < p'_n < p''_n < 1$. Then $E(p_1, \dots, p_{n-1}, p'_n) = E(p_1, \dots, p_{n-1}) + 1$.

Proof. For given $\{p_1, \dots, p_{n-1}\}$ and a procedure T on N with parameters $\{p_1, \dots, p_n\}$,

$$E_T(N) = \sum_{i=1}^{2^n} p(v_i)l(v_i)$$

is linear in p_n since $p(v_i)$ is. Let R denote the class of procedures for N which do not test any set more than once. Then clearly, an optimal procedure must be a member of R . Note that R is a finite set since there are only $2^n - 1$ distinct tests. Now it is well known (see [1], for example) that the minimum of concave functions is concave. Since linear functions are concave,

$$E(p_1, \dots, p_{n-1}, p_n) = E(N) = \min_{T \in R} E_T(N)$$

is concave for $0 < p_n < 1$. Furthermore, $E(N)$ is nondecreasing in p_n by Theorem 2. Therefore

$$E(p_1, \dots, p_{n-1}, p'_n) = E(p_1, \dots, p_{n-1}, p''_n)$$

implies

$$E(p_1, \dots, p_{n-1}, p'_n) = E(p_1, \dots, p_{n-1}, p_n)$$

for all $p'_n < p_n < 1$. From Lemma 3 we have

$$E(p_1, \dots, p_{n-1}) + p_n \leq E(p_1, \dots, p_n) \leq E(p_1, \dots, p_{n-1}) + 1.$$

Letting $p_n \rightarrow 1$, we obtain

$$E(p_1, \dots, p_{n-1}, p'_n) = E(p_1, \dots, p_{n-1}, p_n) = E(p_1, \dots, p_{n-1}) + 1.$$

THEOREM 3. $E(n, p)$ is increasing in p for $0 < p < (3 - \sqrt{5})/2$ for every $n \geq 2$.

Proof. Since $E(2, p) = -p^2 + 3p + 1$ for $0 < p < (3 - \sqrt{5})/2$, Theorem 3 holds for $n = 2$. We prove the general case by induction on n . Suppose to the contrary that there exist $0 < p_1 < p_2 < (3 - \sqrt{5})/2$ such that $E(n, p_1) = E(n, p_2)$. By Theorem 2

$$E(n, p_1) \leq E(p_1, \dots, p_1, p_2) \leq E(p_1, p_2, \dots, p_2) \leq E(n, p_2).$$

Hence all the above inequalities are actually equalities. By Lemma 4 the first equality implies $E(n, p_1) = E(n - 1, p_1) + 1$ and the last equality implies $E(n, p_2) = E(n - 1, p_2) + 1$. Therefore, $E(n, p_1) = E(n, p_2)$ implies $E(n - 1, p_1) = E(n - 1, p_2)$, a contradiction to our inductive assumption.

4. Conclusion. The monotonicity conjecture has been around for at least ten years and in published form since 1981. The difficulty in proving the seemingly obvious conjecture lies in the fact that we do not know $E(n, p)$ in general. Several attempts of proof by comparing $E_T(N)$ with $E_T(N')$ directly have not passed rigorous examination. In this paper we proposed a new approach of transplanting not the algorithm but its probability structure to assure that the expected number of tests is preserved. This approach leads to a proof of the conjecture for the more general case when the states of the items are independent but not necessarily identical. We then use the generalized result to prove a strict version of the original conjecture.

REFERENCES

- [1] R. BELLMAN, *Dynamic Programming*, Princeton University Press, Princeton, NJ, 1957.
- [2] T. BERGER, N. MEHRVARI, D. TOWNSLEY, AND J. WOLF, *Random multiple-access communication and group testing*, IEEE Trans. Comm., 34 (1984), pp. 769–779.
- [3] R. DORFMAN, *The detection of defective members of large population*, Ann. of Math. Statist., 14 (1943), pp. 436–440.
- [4] W. FELLER, *An Introduction to Probability Theory and Its Applications*, Vol. 1, John Wiley, New York, 1957.
- [5] J. F. HAYES, *An adaptive technique for local distribution*, IEEE Trans. Comm., 26 (1978), pp. 1178–1186.
- [6] F. K. HWANG, T. T. SONG, AND D. Z. DU, *Hypergeometric and generalized hypergeometric group testing*, SIAM J. Algebraic Discrete Methods, 2 (1981), pp. 426–428.
- [7] M. SOBEL AND P. A. GROLL, *Group testing to eliminate efficiently all defectives in a binomial sample*, Bell System Tech. J., 38 (1959), pp. 1179–1252.
- [8] P. UNGER, *The cutoff point in group testing*, Comm. Pure Appl. Math., 13 (1960), pp. 49–54.
- [9] S. S. WILKS, *Mathematical Statistics*, John Wiley, New York, 1962.

ALMOST EVERY RANDOMLY NEAR-TRACEABLE GRAPH HAS DIAMETER AT MOST TWO*

BING ZHOU†

Abstract. Randomly near-traceable graphs are those graphs for which depth-first search can be completed with at most one backtracking. In this paper it is proved that every randomly near-traceable graph that is not a cycle has diameter at most two, thus confirming a conjecture of J. F. Fink. Some other special properties of randomly near-traceable graphs are also discussed.

Key words. graph, depth-first search, backtrack, diameter

AMS(MOS) subject classification. 05C

1. Introduction. The goal of this paper is to study a particular type of depth-first search in connected graphs and to settle in the affirmative the following conjecture due to Fink [4].

CONJECTURE. If G is a randomly near-traceable graph and G is not a cycle, then the diameter of G is at most two.

All the graphs that are considered in this paper will be connected graphs. The first section of this paper is devoted to introducing the concept of a depth-first walk on a graph and providing essential background material. Throughout, we assume that the reader is familiar with the concept of depth-first search in a graph (see [1], for example).

Recall that a *walk* in a graph G is a sequence $W = (w_1, w_2, \dots, w_m)$ of vertices from G with repetition allowed so that w_i is adjacent to w_{i+1} for all $i = 1, 2, \dots, m - 1$. A *depth-first search* of G is a search of the vertex set of G , and each step of the search is performed in accordance with the following rules:

Choose an arbitrary vertex w_1 from G . Now suppose that for some integer i with $1 \leq i < m$, we have a walk $(w_1, w_2, \dots, w_i)$. We let $V(i) = \{w_1, w_2, \dots, w_i\}$ denote the set of vertices visited on this walk. The remaining vertices in G are unvisited. If there are no unvisited vertices, STOP. Otherwise, let $S(i)$ denote the set of unvisited vertices that are adjacent to w_i . If $S(i) \neq \emptyset$, then choose w_{i+1} from $S(i)$. If $S(i) = \emptyset$, let k be the least integer for which $w_i = w_k$. Then set $w_{i+1} = w_{k-1}$.

A *depth-first walk* is an initial segment of a walk generated by the above search. Such a walk is *spanning* if every vertex is visited and *partial* otherwise. Note that the number m of vertices in a depth-first walk $W = (w_1, w_2, \dots, w_m)$ may exceed the number of vertices in the graph, since a vertex can be visited in a walk more than once. We say that the walk W *backtracks* at step i if, in the procedure outlined above, the set $S(i)$ is empty so that w_{i+1} is chosen from among the vertices previously visited. If W has no backtracks, then W is just a path in the graph. When W backtracks, we define its *backtrack size* as the largest integer t for which there exists an integer i so that W backtracks at step $i + j$ for each j with $0 \leq j < t$. In other words, the backtrack size of W is the maximum number of consecutive steps at which it backtracks. If W has no backtracks, i.e., W is a path, we say it has backtrack size zero.

* Received by the editors August 25, 1986; accepted for publication (in revised form) September 22, 1987.

† Department of Mathematics, Northeastern University, Boston, Massachusetts 02115. This paper forms part of the author's doctoral dissertation written under the supervision of W. T. Trotter at the University of South Carolina, 1986. This research was supported in part by the National Science Foundation under grant DMS-8401281.

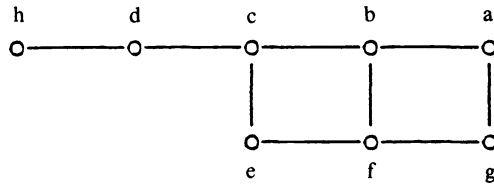


FIG. 1

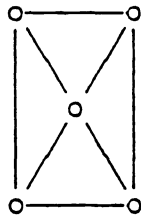
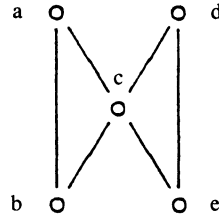
 G_1  G_2

FIG. 2

We illustrate these definitions for the graph shown in Fig. 1. The walk $W_1 = (a, g, f, b, a, b, c)$ is not a depth-first walk; the walk $W_2 = (h, d, c, b, a, g)$ is a depth-first walk with no backtracks (a path); and the walk $W_3 = (c, d, h, d, c, b, a, g, f, e)$ is a depth-first walk with backtrack size two. W_2 is partial and W_3 is spanning.

It is natural to ask which graphs admit spanning depth-first walks with a limited backtrack size. The extreme case is the well-known problem of classifying graphs which have a Hamiltonian path. In [3], Chartrand and Kronk defined a connected graph G on n vertices to be *randomly traceable* if every path in G can be extended to a Hamiltonian path in G ; i.e., for every integer i with $1 \leq i < n$ and every path $p_1, p_2, \dots, p_i$ in G , there exists a Hamiltonian path $v_1, v_2, \dots, v_i, v_{i+1}, \dots, v_n$ in G so that $p_j = v_j$ for each $j = 1, 2, \dots, i$. Alternately, a graph is randomly traceable if every spanning depth-first walk is a Hamiltonian path. Chartrand and Kronk proved that the class of randomly traceable graphs consists of the cycles, the complete graphs, and those complete bipartite graphs with the same number of vertices in both parts.

The primary aim of this paper is to study a natural generalization of randomly traceable graphs. In [4], Fink proposed that a graph be called *randomly near-traceable* if every partial depth-first walk with backtrack size at most one can be extended to a spanning depth-first walk with backtrack size at most one. We illustrate this definition with the graphs shown in Fig. 2. As the reader may verify, G_1 is randomly near-traceable. However, G_2 is not since the partial depth-first walk $W = (c, a, b)$ has backtrack size zero but cannot be extended to a spanning depth-first walk with backtrack size at most one.

In sharp contrast to the very limited structure of randomly traceable graphs, randomly near-traceable graphs have more variation and their structure is difficult to characterize. In § 2 we will give a partial characterization of randomly near-traceable graphs which does not depend on the concept of depth-first search. We will then show that every graph is an induced subgraph of some randomly near-traceable graph, so that it will be difficult to obtain a meaningful characterization of randomly near-traceable graphs in terms of forbidden substructures. In § 3, we will investigate the structure of randomly near-traceable graphs in more detail. Our principal result, the proof of Fink's conjecture, will also be given in § 3.

In this paper, all graphs will be simple and connected. The distance between two vertices is the number of edges on the shortest path between these two vertices. The diameter of a graph G , denoted by $\text{diam}(G)$, is the maximum distance between two vertices of G . All other graph theoretic terminology not defined here is given in [2].

2. A characterization of randomly near-traceable graphs. In [4], Fink gave the following characterization of randomly near-traceable graphs.

LEMMA 1. *A graph G is randomly near-traceable if and only if every spanning depth-first walk $W = (w_1, w_2, \dots, w_m)$ in G is either a Hamiltonian path or satisfies the condition that for some integer k , with $1 \leq k \leq n - 3$, the vertices $w_1, w_2, \dots, w_{k+1}$ form a path in G , $w_k = w_{k+2} = w_{k+4} = \dots = w_{n-1}$, and $\{w_{k+1}, w_{k+3}, \dots, w_n\}$ is an independent set of vertices (see Fig. 3). $\square$*

This characterization, like the definition of a randomly near-traceable graph, is based on the concept of depth-first search. Now we give a second characterization of randomly near-traceable graphs which will prove to be quite useful in subsequent arguments. First we define a path-cut P of a graph G to be a (vertex) cut set of G with the property that the subgraph induced by P has a Hamiltonian path.

THEOREM 2. *A graph G is randomly near-traceable if and only if for every path-cut P of G , all components of $G - P$ are trivial, i.e., $G - P$ is an independent set.*

Proof. Suppose first that G is a randomly near-traceable graph and that there is a path-cut P of G for which some component of $G - P$ is nontrivial. Among all such path-cuts, choose P so that $|P|$ is as small as possible. We show that there is a partial depth-first walk W with backtrack size at most one which cannot be completed to form a spanning depth-first walk with backtrack size at most one. In fact, we will construct W so that the first k vertices in W form a Hamiltonian path $(p_1, p_2, \dots, p_k)$ in P .

If p_k is adjacent to a vertex x in a nontrivial component C of $G - P$, then we take $W = (p_1, p_2, \dots, p_k, x)$. It is clear that any spanning depth-first walk extending W has backtrack size at least two since such a walk must backtrack at two consecutive steps in order to leave C .

Now we assume that p_k is not adjacent to any nontrivial component of $G - P$. Let N denote the set of neighbors of p_k in $G - P$. Suppose first that $N = \emptyset$. Then the path $P' = P - \{p_k\}$ is a path-cut and $G - P'$ has a nontrivial component. This contradicts our assumption that $|P|$ is minimum. Thus we must have $N \neq \emptyset$. Let $N = \{y_1, y_2, \dots, y_q\}$. We then take $W = (p_1, p_2, \dots, p_k, y_1, p_k, y_2, p_k, \dots, y_q)$. Note that W is a partial depth-first walk of G and in any depth-first walk W' extending W , the next step must be a backtrack, to p_k . Since all of the neighbors of p_k have been visited, W' has backtrack size at least two. With this contradiction, we conclude that for every path-cut P of G , all components of $G - P$ are trivial.

We now assume that G satisfies the property that whenever P is a path-cut of G , every component of $G - P$ is trivial. Let $W = (w_1, w_2, \dots, w_m)$ be a spanning depth-

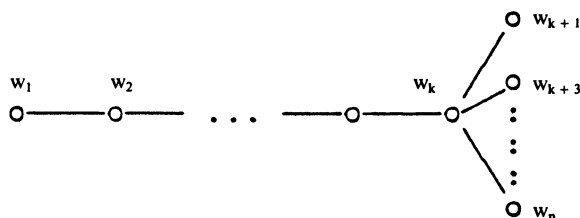


FIG. 3

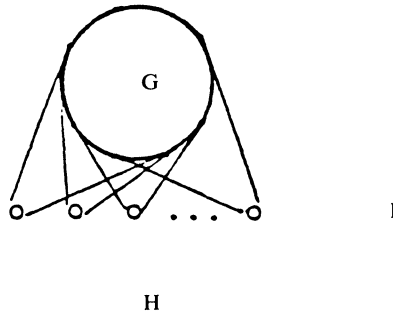


FIG. 4

first walk of G . If W is not a Hamiltonian path, let i be the least integer for which there is some j so that $i < j$ and $w_i = w_j$. Then $P = (w_1, w_2, \dots, w_i)$ is a path-cut of G . Since every component of $G - P$ is trivial, every vertex of $G - P$ is adjacent in G to w_i (otherwise i is not minimum). Thus W has backtrack size one, and we conclude that G is randomly near-traceable. $\square$

THEOREM 3. *Every graph G is an induced subgraph of a randomly near-traceable graph.*

Proof. Given a graph G of n vertices, we construct a randomly near-traceable graph H as follows (see Fig. 4):

Let I be an independent set with $n - 1$ vertices. We let

$$V(H) = V(G) \cup V(I),$$

$$E(H) = E(G) \cup \{(u, v) : u \in V(G), v \in V(I)\}.$$

It is clear that G is an induced subgraph of H . Now we show that H is randomly near-traceable. Let P be a path-cut of H . P must contain all vertices of G or all vertices of I . If P contains all vertices of G , then $H - P$ is independent. So we assume that P contains all vertices of I . Then P must contain at least $n - 2$ vertices of G . Thus $H - P$ must be a single vertex or a disconnected graph on two vertices. In either case each component is trivial. It now follows from Theorem 2 that H is randomly near-traceable. $\square$

3. The diameter of randomly near-traceable graphs. Among the randomly traceable graphs, the complete graphs have diameter one, and the complete bipartite graphs have diameter two. Only the cycles can have large diameter. Since a cycle is also randomly near-traceable, this family of graphs also includes graphs of arbitrarily large diameter. We will show that the cycles are the only randomly near-traceable graphs with large diameter.

In the remainder of this paper, we consider only those randomly near-traceable graphs which are not randomly traceable. We call such graphs proper randomly near-traceable graphs. Every proper randomly near-traceable graph has a depth-first walk with backtrack size one.

For a depth-first walk of randomly near-traceable graph as described in Lemma 1, we will call the path $(w_1, w_2, \dots, w_k)$ the trunk of the walk and the vertices $w_{k+1}, w_{k+3}, \dots, w_n$ the leaves of the walk. If a proper randomly near-traceable graph G has a depth-first walk with more than two leaves, we say that G is of type 1. If every depth-first walk of a proper randomly near-traceable graph G has at most two leaves, we say that G is of type 2.

Every type 1 randomly near-traceable graph has a structure similar to the graph H we constructed in the proof of Theorem 3.

LEMMA 4. *Suppose G is a type 1 randomly near-traceable graph. Then there is a partition $V(G) = F \cup I$ of the vertex set of G so that I is an independent set in G and every vertex of F is adjacent to every vertex of I .*

Proof. Let W be a depth-first walk of G so that the number of leaves of W is as large as possible. Let $T = \{x_1, x_2, \dots, x_k\}$ be the trunk of W and let $L = \{y_1, y_2, \dots, y_r\}$ be the set of leaves of W . By Lemma 1, L is an independent set.

CLAIM 1. x_1 is adjacent to y_i for each $i = 1, 2, \dots, r$.

Suppose the claim is false. Without loss of generality, we may assume that x_1 is not adjacent to y_1 . Then the path $P = (x_2, x_3, \dots, x_k)$ separates y_1 from $\{x_1, y_2, y_3, \dots, y_r\}$. According to Theorem 2, $\{x_1, y_2, y_3, \dots, y_r\}$ is an independent set. So if $x_1 x_k$ is an edge in G , the depth-first walk having P as its trunk has more leaves than W . Otherwise, it has backtrack size at least two. The contradictions complete the proof of this claim.

CLAIM 2. For each i with $1 < i < k$, if x_i is not adjacent to some y_j , then it is not adjacent to y_h for all h with $1 \leq h \leq r$.

Suppose the claim is false. Without loss of generality, let x_i be adjacent to y_1 but not y_2 . The path $P = (x_{i+1}, x_{i+2}, \dots, x_k, y_3, x_1, x_2, \dots, x_{i-1})$ is a path-cut but $\{x_i, y_1\}$ is a component of $G - P$. This stands in contradiction to Theorem 2, so the proof of the claim is complete.

Next we define a partition $\{x_1, x_2, \dots, x_k\} = X_1 \cup X_2$, where $x_i \in X_1$ if x_i is adjacent to all the leaves, otherwise $x_i \in X_2$.

CLAIM 3. If $x_i \in X_2$ for some i with $2 \leq i \leq k-1$, both x_{i-1} and x_{i+1} are in X_1 .

Suppose x_i and x_{i+1} are both in X_2 . Then the path

$$P = (x_{i+2}, x_{i+3}, \dots, x_k, y_1, x_1, x_2, \dots, x_{i-1})$$

is a path-cut and $\{x_i, x_{i+1}\}$ is a component in $G - P$. The contradiction shows that $x_{i+1} \in X_1$. An analogous argument shows $x_{i-1} \in X_1$.

CLAIM 4. X_2 is an independent set.

Suppose to the contrary that there exist adjacent vertices $x_i, x_j \in X_2$. We assume $1 < i < j < k$. Then $\{x_i, x_j\}$ is a component in $G - P$ where P is the path-cut $(x_1, x_2, \dots, x_{i-1}, y_1, x_{i+1}, x_{i+2}, \dots, x_{j-1}, y_2, x_{j+1}, x_{j+2}, \dots, x_k)$.

CLAIM 5. For every $x_i \in X_1$ and $x_j \in X_2$, x_i is adjacent to x_j .

Suppose x_i is not adjacent to x_j . Without loss of generality, we assume that $i < j$. The path $P = (x_{i-1}, x_{i-2}, \dots, x_1, y_1, x_k, x_{k-1}, \dots, x_{j+1}, y_2, x_{j-1}, x_{j-2}, \dots, x_{i+1})$ separates x_j from $\{x_i, y_3, y_4, \dots, y_r\}$ and is therefore a path-cut of G . But $\{x_i, y_3, y_4, \dots, y_r\}$ is a component of $G - P$.

Now that we have established this series of claims, we are ready to complete the proof of our lemma. To accomplish this, we set $I = X_2 \cup \{y_1, y_2, \dots, y_r\}$ and $F = X_1$. $\square$

For type 2 randomly traceable graphs, the proof of Theorem 2 can be easily modified to yield the following strengthened version.

THEOREM 2'. *A graph G is a type 2 randomly near-traceable graph if and only if for every path-cut P of G , there are at most two components in $G - P$ and one vertex in each component, i.e., $|G - P| \leq 2$. $\square$*

Every type 2 randomly near-traceable graph with at least seven vertices also has a structure similar to the graph H in the proof of Theorem 3, but the similarity is less obvious than is the case for type 1 randomly near-traceable graphs.

LEMMA 5. Suppose G is a type 2 randomly near-traceable graph and $|V(G)| \geq 7$. Then there is a partition $V(G) = F \cup B$ of the vertex set of G so that every vertex in F is adjacent to every vertex in B .

Proof. We fix a depth-first walk W of G that has two leaves. Let the trunk of W be $T = (x_1, x_2, \dots, x_k)$ and let the leaves be $\{y_1, y_2\}$. Let

$$V_0 = \{x_i : x_i \text{ is adjacent to neither of the leaves}\};$$

$$V_1 = \{x_i : x_i \text{ is adjacent to exactly one of the leaves}\};$$

$$V_2 = \{x_i : x_i \text{ is adjacent to both of the leaves}\}.$$

CLAIM 1. x_1 is adjacent to both y_1 and y_2 .

Suppose, for example, that x_1 is not adjacent to y_1 . Then the path

$$P = (x_2, x_3, \dots, x_k)$$

separates y_1 from $\{x_1, y_2\}$, so it is a path-cut. But $|G - P| = 3$, which is a contradiction.

By Claim 1, the path $(x_k, x_{k-1}, \dots, x_1)$ is also the trunk of a depth-first walk W' having the same two leaves as W . In subsequent arguments, we will take advantage of the observation that we can reverse the direction of the trunk T whenever so desired.

We will use the same type of argument as in the proof of Claim 1 in the following cases and we will just simply mention the path and the separated sets.

CLAIM 2. If x_i is not adjacent to y_j , then both x_{i-1} and x_{i+1} are adjacent to y_j . In particular, if $x_i \in V_0$, then $x_{i-1}, x_{i+1} \in V_2$.

By the observation after Claim 1, we only need to prove the result of this claim for one side of x_i . But if both x_{i-1} and x_i are not adjacent to y_1 , then

$$(x_{i-2}, x_{i-3}, \dots, x_1, y_2, x_k, x_{k-1}, \dots, x_{i+1})$$

separates $\{x_{i-1}, x_i\}$ from y_1 .

CLAIM 3. There do not exist three consecutive vertices on T each of which belongs to V_1 .

Suppose the claim is false, and that there exists some p so that $x_{p-1}, x_p, x_{p+1} \in V_1$. Without loss of generality, we may assume that $x_{p+2} \notin V_1$. Since $x_{p+2} \in V_0$ implies $x_{p+1} \in V_2$ according to Claim 2, we conclude that $x_{p+2} \in V_2$. We may assume that x_{p+1} is adjacent to y_2 but not y_1 . Then according to Claim 2, these assumptions imply that x_p is adjacent to y_1 but not y_2 and x_{p-1} is adjacent to y_2 but not y_1 .

Also x_{p+1} is not adjacent to x_1 , otherwise

$$(x_{p-1}, x_{p-2}, \dots, x_1, x_{p+1}, x_{p+2}, \dots, x_k)$$

separates $\{x_p, y_1\}$ from y_2 . Similarly, x_p is not adjacent to x_1 .

Now consider the path $P = (x_2, x_3, \dots, x_{p-1}, y_2, x_k, x_{k-1}, \dots, x_{p+2}, y_1)$. By the preceding paragraph this path separates $\{x_p, x_{p+1}\}$ from x_1 . This contradicts the conclusion of Theorem 2'. This observation completes the proof of Claim 3.

CLAIM 4. If $|V(G)| \geq 7$, there do not exist two consecutive vertices on T both of which belong to V_1 .

By Claim 3, there are at most two consecutive vertices on T both in V_1 . Suppose x_p, x_{p+1} are two such vertices. We assume that x_p is adjacent to y_1 but not y_2 ; thus x_{p+1} is adjacent to y_2 but not y_1 . By Claim 2, neither x_{p-1} nor x_{p+2} is in V_0 , so they are both in V_2 .

As in Claim 3, neither x_{p+1} nor x_{p+2} is adjacent to x_1 . Since $|V(G)| \geq 7$, we cannot have both $p-1=1$ and $p+2=k$. In light of Claim 1, we may assume that

$p + 2 < k$. Also we must have that x_p is not adjacent to x_{p+2} for otherwise the path $P = (x_1, x_2, \dots, x_p, x_{p+2}, x_{p+3}, \dots, x_k)$ separates $\{x_{p+1}, y_2\}$ from y_1 , which contradicts Theorem 2'. Similarly, we also must have that x_p is adjacent to x_{p+3} , otherwise the path $(x_{p+1}, y_2, x_{p-1}, x_{p-2}, \dots, x_1, y_1, x_k, \dots, x_{p+4})$ separates x_p from $\{x_{p+2}, x_{p+3}\}$.

Now consider the path $p = (x_2, x_3, \dots, x_{p-1}, y_2, x_k, x_{k-1}, \dots, x_{p+3}, x_p, y_1)$ which separates x_1 from $\{x_{p+1}, x_{p+2}\}$. This is a contradiction to Theorem 2' and this contradiction completes the proof of Claim 4.

By Claim 2 and Claim 4, we see that if $|V(G)| \geq 7$, T is formed by blocks of vertices in V_2 (we will call those V_2 -blocks) with a single vertex from V_0 or V_1 between these blocks. If there is only one V_2 -block, i.e., the trunk contains no vertex from V_0 or V_1 , then we can take $F = V_2$ and $B = \{y_1, y_2\}$ and Lemma 5 holds. Therefore we may assume there are two or more V_2 -blocks.

CLAIM 5. x_1 and x_k are adjacent to every vertex in $V_0 \cup V_1$.

By symmetry, we only need to prove this claim for x_1 . Suppose $x_p \in V_0 \cup V_1$, and x_1 is not adjacent to x_p . Without loss of generality, we assume that x_p is not adjacent to y_1 . Then both x_{p-1} and x_{p+1} are in V_2 -blocks and are adjacent to y_2 . The path $P = (x_2, x_3, \dots, x_{p-1}, y_2, x_{p+1}, x_{p+2}, \dots, x_k)$ separates x_p from $\{x_1, y_1\}$, which contradicts Theorem 2'.

If $x_i \in V_2$ and either x_{i-1} or x_{i+1} is not in V_2 , then we say that x_i is an endpoint of a V_2 -block.

CLAIM 6. For every endpoint x_i of a V_2 -block, there is a depth-first walk W' with $\{y_1, y_2\}$ being the set of leaves and x_i being the first vertex on the trunk.

Suppose x_{i-1} is not in V_2 . Then by Claim 5, x_{i-1} is adjacent to x_k and the walk $W' = (x_i, x_{i+1}, \dots, x_k, x_{i-1}, x_{i-2}, \dots, x_1, y_1, x_1, y_2)$ satisfies Claim 6. If x_{i+1} is not in V_2 , then the argument is similar.

Claim 5 and Claim 6 together show that every endpoint of a V_2 -block is adjacent to every vertex in V_0 or V_1 . Let F_0 consist of all the endpoints of V_2 -blocks and let $B_0 = V_0 \cup V_1 \cup \{y_1, y_2\}$. If there is no V_2 -block with more than two vertices, then $V(G) = F_0 \cup B_0$ and the lemma is proved. So we assume that there is at least one V_2 -block with at least three vertices. According to Claim 6, we may assume that this block is the first V_2 -block on the trunk of W . Therefore we assume that $x_1, x_2, \dots, x_p$, where p is at least three, are all in V_2 , and x_{p+1} is not in V_2 . We assume that x_{p+1} is not adjacent to y_2 .

In this case x_1 is not adjacent to x_k , for otherwise

$$(x_p, x_{p-1}, \dots, x_1, x_k, x_{k-1}, \dots, x_{p+2})$$

separates y_2 from $\{x_{p+1}, y_1\}$. Similarly, x_p is not adjacent to x_k .

CLAIM 7. Every V_2 -block that is not the first one has only one vertex.

Suppose this claim is false. By the proof of Claim 6, we may assume that the last V_2 -block has more than one vertex. Note that we have already assumed that the first one has at least three vertices. The path $P = (y_1, x_2, \dots, x_{p-1}, y_2, x_{k-1}, x_{k-2}, \dots, x_{p+1})$ separates x_k from $\{x_1, x_p\}$, which is a contradiction.

CLAIM 8. Every vertex in the first V_2 -block is either adjacent to both x_k and x_1 or adjacent to every vertex in $V_0 \cup V_1$.

We have already seen that x_1 and x_p are adjacent to every vertex in $V_0 \cup V_1$, so we only need to consider the vertices in the middle of the block.

Suppose for $1 < i < p$, x_i is not adjacent to x_k (the case that x_i is not adjacent to x_1 can be treated the same way), and there is $x_h \in V_0 \cup V_1$, such that x_i is not adjacent to x_h . Then the path

$$P = (x_1, x_2, \dots, x_{i-1}, y_1, x_{i+1}, x_{i+2}, \dots, x_{h-1}, y_2, x_{h+1}, x_{h+2}, \dots, x_{k-1})$$

separates x_i from $\{x_h, x_k\}$, which is a contradiction.

We can partition the first V_2 -block into two parts:

$$B_1 = \{x_i: 1 \leq i \leq p, x_i \text{ is adjacent to every vertex in } V_0 \cup V_1\}$$

and

$$B_2 = \{x_i: 1 \leq i \leq p, x_i \text{ is not in } B_1\}.$$

By Claim 8, every vertex in B_2 is adjacent to both x_1 and x_k . According to Claim 9 and the proof of Claim 6, we can see that for every vertex x_j that is in V_2 but not in the first block, there is a depth-first walk W' of G so that x_j is the last vertex on the trunk of W' , and W' has the same V_0 , V_1 , V_2 sets and the same first V_2 -block as W . So every vertex in B_2 is adjacent to every vertex in V_2 that is not in the first block, i.e., every vertex in F_0 . Let $F = F_0 \cup B_1$ and $B = B_0 \cup B_2$. To complete the proof of this lemma we only need to establish the following claim.

CLAIM 9. Every vertex in B_1 is adjacent to every vertex in B_2 .

It is easy to see that if the first V_2 -block has no more than three vertices, then this claim is true. So we assume that the first V_2 -block has at least four vertices. Actually, in this case, x_1 and x_p are the only two vertices in B_1 . Suppose there is some x_i with $i \neq 1$, $i \neq p$ in B_1 . Since $p \geq 4$, either $i \neq 2$ or $i \neq p-1$. Without loss of generality, we assume that $i \neq 2$. The path

$$P = (y_1, x_2, x_3, \dots, x_{i-1}, y_2, x_{p-1}, x_{p-2}, \dots, x_i, x_{p+1}, x_{p+2}, \dots, x_{k-1})$$

separates x_k from $\{x_1, x_p\}$, which is a contradiction.

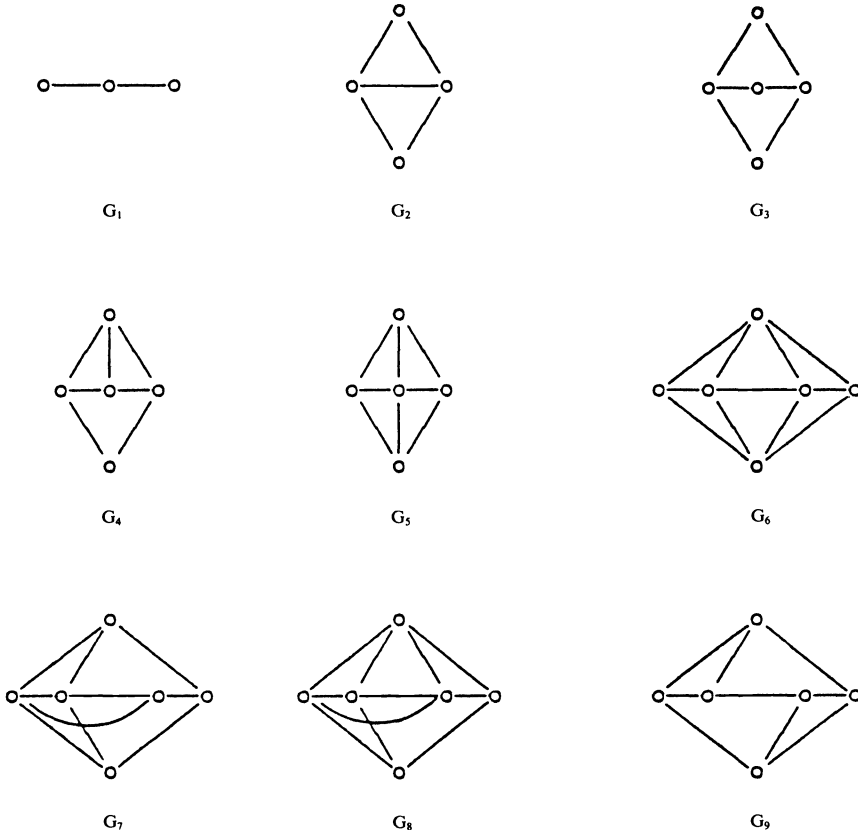


FIG. 5

Now suppose the claim is not true. Then there is an x_i in B_2 such that x_i is not adjacent to either x_1 or x_p . Without loss of generality we assume x_i is not adjacent to x_1 and $i \neq 2$. Then the path $P = (y_1, x_2, x_3, \dots, x_{i-1}, y_2, x_{i+1}, x_{i+2}, \dots, x_{k-1})$ separates x_1 from $\{x_i, x_k\}$. The contradiction completes the proof of the lemma. $\square$

Remark. Let G be the graph obtained by adding one edge to each of the two parts of the complete bipartite graph $K_{n,n}$ ($n \geq 3$). It is easy to check that G satisfies the condition of Theorem 2' and is therefore a type 2 randomly near-traceable graph. But it is impossible to obtain a partition of the vertex set of G as described in Lemma 4. This example shows that the result of Lemma 5 is best possible.

THEOREM 6. *For every randomly near-traceable graph G that is not a cycle, the diameter of G is at most 2.*

Proof. We have seen that all randomly traceable graphs that are not cycles have diameter at most 2. All type 1 randomly near-traceable graphs and all type 2 randomly near-traceable graphs with seven or more vertices have diameter at most two according to Lemma 4 and Lemma 5. To complete the proof of this theorem, we list all the type 2 randomly near-traceable graphs with at most six vertices in Fig. 5 (where G_6 represents the class of graphs that can be obtained by adding edges among the four vertices on the path in the middle of G_6). It is easy to see they all have diameter at most two. It is also interesting to notice that the last graph, G_9 , is the only graph in this list that cannot be partitioned as described in Lemma 5. $\square$

Remark. Not every connected graph G with $\text{diam}(G) \leq 2$ is a randomly near-traceable graph. One example is the graph G_2 in Fig. 2: $\text{diam}(G_2) = 2$ but G_2 is not a randomly near-traceable graph.

Acknowledgments. The author thanks the referees for their helpful comments.

REFERENCES

- [1] A. V. AHO, M. R. GAREY, AND J. D. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.
- [2] J. A. BONDY AND U. S. R. MURTY, *Graph Theory with Applications*, American Elsevier, New York, 1976.
- [3] G. CHARTRAND AND H. V. KRONK, *Randomly traceable graphs*, SIAM J. Appl. Math., 16 (1968), pp. 696–700.
- [4] J. F. FINK, *Randomly near-traceable graphs*, SIAM J. Algebraic Discrete Methods, 6 (1985), pp. 251–258.

SORTING, APPROXIMATE SORTING, AND SEARCHING IN ROUNDS*

NOGA ALON† AND YOSSIE AZAR‡

Abstract. The worst case number of comparisons needed for sorting or selecting in rounds is considered. The following results are obtained.

(a) For every fixed $k \geq 2$, $\Omega(n^{1+1/k}(\log n)^{1/k})$ comparisons are required to sort n elements in k rounds. ($O(n^{1+1/k} \log n)$ are known to be sufficient.) This improves the previous known bounds by a factor of $(\log n)^{1/k}$, which separates deterministic algorithms from randomized ones, as there are randomized algorithms whose expected number of comparisons is $O(n^{1+1/k})$.

(b) For every fixed $k \geq 2$, $\Omega(n^{1+1/(2^k-1)}(\log n)^{2/(2^k-1)})$ comparisons are required to select the median from n elements in k rounds. ($O(n^{1+1/(2^k-1)}(\log n)^{2-2/(2^k-1)})$ are known to be sufficient.) This improves the previous known bounds by a factor of $(\log n)^{2/(2^k-1)}$ and separates the problem of finding the median from that of finding the minimum, as $O(n^{1+1/(2^k-1)})$ comparisons suffice for finding the minimum.

(c) We show that “approximate sorting” in one round requires asymptotically more than $c \cdot n \log n$ comparisons, for every constant c , and can be done in $O(n \log n \log \log n)$ comparisons. This settles a problem raised by Rabin.

Key words. sorting in rounds, searching, approximate sorting, parallel comparison algorithms

AMS(MOS) subject classification. 68E05

1. Introduction. Sorting and selecting are two of the principal problems in computer science. As mentioned in [Kn73], existing computers devote approximately one quarter of their time to sorting. The advent of parallel computers stimulated intensive research of sorting and selecting with respect to various models of parallel computation. Extensive lists of references that recorded this activity are given in [Ak85] and [BHe85].

Most of the fastest serial and parallel sorting or selecting algorithms are based on binary comparisons. In these algorithms the number of comparisons is typically the primary measure of time complexity. Any lower bound on the number of comparisons required for a problem, clearly implies a time lower bound for such algorithms. In the present paper, we restrict our attention to a parallel comparison model, introduced by Valiant [Va75] (see also [BHo82]), where only comparisons are counted. In measuring time complexity within this model, we do not count steps in which communication among the processors, movement of data, and memory addressing are performed. We also avoid counting steps in which consequences are deduced from comparisons that were performed. Note that our lower bounds apply to all comparison-based algorithms in any model.

In a serial decision tree model, we wish to minimize the number of comparisons. The goal of an algorithm in a parallel comparison model is to minimize the number of comparison rounds as well as the total number of comparisons performed.

We consider three problems: sorting, “approximate sorting,” and selecting.

1.1. Sorting in rounds. Let k stand for the number of comparison rounds (time) of an algorithm in the parallel comparison model. Let $c(k, n)$ denote the *minimum total* number of comparisons required to sort any n elements in k rounds (over all possible algorithms).

* Received by the editors March 15, 1987; accepted for publication (in revised form) February 5, 1988.

† School of Mathematical Sciences, Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, Israel. This research was supported in part by an Allon Fellowship, by a grant from Bat Sheva de Rothschild Foundation, and by the Fund for Basic Research Administered by the Israel Academy of Sciences.

‡ School of Mathematical Sciences, Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, Israel.

Clearly $c(1, n) = \binom{n}{2}$. This is since any sorting algorithm that works in one round must perform all comparisons. Otherwise, suppose that a dispensed comparison is between two successive elements in the sorted order; the algorithm will clearly fail to distinguish their order. On the other hand, performing all comparisons simultaneously yields a one round algorithm with that many comparisons.

In general, it is known that for every fixed k

$$\Omega(n^{1+1/k}) \leq c(k, n) \leq O(n^{1+1/k} \log n).$$

The upper bound is due to Bollobás and Hell [BHe85] (see also [Pi87]). The lower bound is due to Häggkvist and Hell [HH81] (see also [AV87], [AAV86] for the case of not necessarily constant k).

Even the first nontrivial case, that of sorting n elements in two rounds, received considerable attention. Häggkvist and Hell [HH81] showed that

$$\frac{1}{8}n^{3/2} - \frac{1}{2}n \leq c(2, n) = O(n^{5/3} \log n).$$

Bollobás and Thomason [BT83] improved it and showed that

$$c_1 \cdot n^{3/2} \leq c(2, n) = O(n^{3/2} \log n)$$

for any $c_1 < \sqrt{2/3}$, if $n > n(c_1)$.

Explicit algorithms for sorting in two rounds with $\alpha(n^2)$ comparisons are given in [Pi85], [Al85], and [Pi87].

Here we (slightly) improve both bounds and prove Theorem 1.1.

THEOREM 1.1.

$$\Omega(n^{3/2} \sqrt{\log n}) \leq c(2, n) \leq O\left[n^{3/2} \frac{\log n}{\sqrt{\log \log n}}\right].$$

We also prove the following theorem.

THEOREM 1.2. *For every fixed $k \geq 2$*

$$c(k, n) = \Omega(n^{1+1/k}(\log n)^{1/k}).$$

This improves on Häggkvist and Hell's lower bound mentioned above. Notice that the only difference between our improved lower bound and the previously known one, is an extra factor of $(\log n)^{1/k}$. Nevertheless, this is precisely the factor that separates the asymptotic behavior of the best randomized algorithm (which is $\Theta(n^{1+1/k})$; see [AA87], [AAV86]) from that of the best deterministic one.

1.2. Approximate sorting in one round. An algorithm that approximately sorts n elements in one round with p comparisons is a set of p pairs of elements $(a_i, b_i)_{i=1}^p$ from the set V of n elements we have to sort, such that for each possible set of answers for the p questions "is $a_i < b_i$ " the relative order of all but $\alpha(n^2)$ of the pairs of elements of V will be known.

Note that such an algorithm is simply the first round of a two rounds sorting algorithm that uses only $\alpha(n^2)$ comparisons in the second round. Let $a(n)$ denote the minimum p such that an approximate sorting algorithm in one round with p comparisons exists. Bollobás and Rosenfeld studied these algorithms in [BR81] (also see [BHe85]) and their results imply that for every fixed $\epsilon > 0$, $a(n) = \alpha(n^{1+\epsilon})$.

M. Rabin (cf. [BHe85]) asked whether $a(n) = O(n \log n)$. The next proposition shows that this is false.

THEOREM 1.3. (i) $\lim_{n \rightarrow \infty} a(n)/n \log n \rightarrow \infty$. *More precisely, for every $\varepsilon > 0$, any two rounds sorting algorithm that uses at most εn^2 comparisons in the second round must use $\Omega((1/\varepsilon)n \log n)$ comparisons in the first round.*

(ii) $a(n) \leq O(n \log n \log \log n)$.

More precisely, there is a two rounds sorting algorithm that uses

$$O(n \log n \log \log n)$$

comparisons in the first round and $O(n^2/\log \log \log n) = o(n^2)$ comparison in the second round.

Ajtái et al. [AKSS86b] have recently proved, independently of our work, a slightly weaker result than Theorem 1.3. Their bounds are:

$$\Omega(n \log n) \leq a(n) \leq O(n \log n \log \log nw(n))$$

where $w(n)$ is any function that tends to infinity as n tends to infinity. (Note that this lower bound does not suffice to settle Rabin's problem.) More recently, Bollobás and Brightwell [BB87] improved the upper bound and showed that

$$a(n) = O(n \log n \log \log n \cdot w(n)/\log \log \log n),$$

where $w(n)$ is as above.

1.3. Selecting in a fixed number of rounds. Häggkvist and Hell [HH80] showed that $\Omega(n^{1 + 1/(2^k - 1)})$ comparisons are necessary for selection in k rounds and that $O(n^{1 + 1/(2^k - 1)})$ comparisons are sufficient for finding the minimum (or selecting any element of a fixed rank) among n elements in k rounds. Pippenger [Pi87] proved that $O(n^{1 + 1/(2^k - 1)}(\log n)^{2 - 2/(2^k - 1)})$ suffice for selecting the median (or any element of a given rank) in k rounds. A natural problem that arises here is whether the median can be found as efficiently as the minimum. Solving this problem we prove that finding the median in k rounds requires $\Omega(n^{1 + 1/(2^k - 1)}(\log n)^{2/(2^k - 1)})$ comparisons. This separates the asymptotic number of comparisons needed for selecting the median from that needed for selecting the minimum. We also slightly improve the best known upper bounds for selection, which are given in [Pi87]. Let $s(k, n)$ denote the minimum total worst case number of comparisons needed to select the median of n elements in k rounds. Our result is the following theorem.

THEOREM 1.4. *For every fixed $k \geq 2$*

$$\begin{aligned} \Omega(n^{1 + 1/(2^k - 1)}(\log n)^{2/(2^k - 1)}) &\leq s(k, n) \\ &\leq O(n^{1 + 1/(2^k - 1)}(\log n)^{2 - 2/(2^k - 1)}/(\log \log n)^{1 - 1/(2^k - 1)}). \end{aligned}$$

All our results are proved by combining probabilistic arguments with various results from extremal graph theory. The paper is organized as follows. In § 2 we first prove a lemma about proper vertex colorings of graphs and then deduce from it the lower bounds of Theorems 1.1, 1.2, 1.3, and 1.4. In § 3 we combine certain probabilistic arguments with some of the ideas of [AKSS86a], [BT83], and [Pi87] to obtain all the upper bounds.

2. The lower bounds.

2.1. The parallel computation model. Let V be a set of n elements taken from a totally ordered domain. The *parallel comparison model of computation* allows algorithms that work as follows. The algorithm consists of timesteps called *rounds*. In each round binary comparisons are performed simultaneously. The input for each comparison are two elements of V . The output of each comparison is one of the following two: $<$ or $>$. Note that we do not allow equality between two elements of V . This can be done with-

out loss of generality, since we define the order between two equal input elements to be the order of their indices. Each item may take part in several comparisons during the same round.

Our discussion uses the following correspondence between each round and a graph. The elements are the vertices. Each comparison to be performed is an undirected edge that connects its input elements. Each computation results in orienting this edge from the largest element to the smallest. Thus in each round we get an acyclic orientation of the corresponding graph, and the transitive closure of the union of the r oriented graphs obtained until round r represents the set of all pairs of elements whose relative order is known at the end of round r .

Suppose we performed r rounds where $r > 0$ is some integer. Consider any function of V that can be computed using the comparisons performed in these r rounds without any further comparisons of elements in V . Our model defines such a function to be *computable following round r* . Note that this definition suppresses all computational steps that do not involve comparisons of elements in V . Which comparisons to perform at round $r + 1$ and the input for each such comparison should be functions which are computable following round r . We are interested in algorithms that sort the elements of V or select an element of a given rank from V .

2.2. A graph theoretic lemma. The following lemma is crucial in all our lower bound proofs.

LEMMA 2.1. *Every graph with n vertices and at most dn edges, $d \geq 1$ integer, contains an induced subgraph with $\lceil n/4 \rceil$ vertices and maximum degree $< 4d$, which has a $4d$ proper vertex coloring with color classes $V_1, V_2, \dots, V_{4d}$ such that for each $1 \leq i, j \leq 4d$ and each $v \in V_i$, v has at most $2^{|i-j|+1}$ neighbors in V_j .*

Proof. Let $G = (V, E)$ be a graph with n vertices and at most dn edges. Omit successively the highest degree vertex $\lfloor n/2 \rfloor$ times. We are left with an induced subgraph K on a set U of $\lceil n/2 \rceil$ vertices and maximum degree $< 4d$ (otherwise, we omitted at least $\lfloor n/2 \rfloor \cdot 4d \geq nd$ edges) and with at most $\frac{1}{2}nd$ edges (we omitted at least half of the edges). By a standard result from extremal graph theory, K has a proper $4d$ vertex coloring. Let $U_1, \dots, U_{4d}$ be the color classes. For every vertex u of K , let $N(u)$ denote the set of all neighbors of u in K . For a permutation π of $1, 2, \dots, 4d$ and any vertex u of K , define the π -degree $d(\pi, u)$ of u as follows. Let i satisfy $u \in U_{\pi(i)}$; then

$$d(\pi, u) = \sum_{j=1}^{4d} |N(u) \cap U_{\pi(j)}| / 2^{|i-j|}.$$

Now consider the sum $\sum_{u \in U} d(\pi, u)$. We claim that the expected value of this sum over all the permutations π of $\{1, \dots, 4d\}$ is at most $|U|$. Indeed for a random permutation π , the probability that a fixed edge $\{u, v\}$ in U contributes $2 \cdot 1/2^r$, $r > 0$, to the sum ($1/2^r$ to $d(\pi, u)$ and $1/2^r$ to $d(\pi, v)$) is at most $(4d - r) / \binom{4d}{2} \leq 1/2d$ for all $r > 0$. But there are at most $\frac{1}{2}nd$ edges and therefore this sum is at most $\frac{1}{2}nd \cdot 1/2d \sum_{r>0} 2/2^r \leq n/2 \leq |U|$. Hence, there is a fixed permutation σ such that $\sum_{u \in U} d(\sigma, u) \leq |U|$. It follows that $d(\sigma, u) \leq 2$ for at least $|U|/2 \geq n/4$ vertices of K . Let W be a set of $\lceil n/4 \rceil$ of these vertices. Let H be the induced subgraph of G on W , and define $V_i = U_{\sigma(i)} \cap W$ ($1 \leq i \leq 4d$). Clearly for every $1 \leq i, j \leq 4d$ and every $v \in V_i$, $\sum_{j=1}^{4d} |N(v) \cap V_j| / 2^{|i-j|} \leq 2$ and thus v has at most $2^{|i-j|+1}$ neighbors in V_j . This completes the proof. $\square$

Remark. In the last lemma we can replace $2^{|i-j|+1}$ by $O(f(|i-j|))$ for every function f such that $\sum_{i>0} 1/f(i) < \infty$. For our purpose here the function 2^i suffices.

2.3. Lower bounds for sorting. In this section we prove the lower bounds in Theorems 1.1, 1.2, and 1.3.

LEMMA 2.2. Every graph $G = (V, E)$ with n vertices and at most $n \cdot d$ edges, where $d = o(n)$ and $d = \Omega(\log n)$, has an acyclic orientation whose transitive closure has at most $\binom{n}{2} - \Omega((n^2/d) \log(n/d))$ edges.

Proof. By Lemma 2.1 there is a subset W of cardinality $\lfloor n/4 \rfloor$ of V and a proper $4d$ vertex coloring of the induced subgraph of G on W with color classes $V_1, \dots, V_{4d}$ satisfying the conclusions of the lemma. Put $V_0 = V - W$ and orient each edge (u, v) of G with $u \in V_i, v \in V_j$ and $0 \leq i < j \leq 4d$ from u to v . The other edges of V (that join two members of V_0) will be oriented in an arbitrary acyclic order. Let T be the transitive closure of this oriented graph. For $v \in V$, let $N_T(v)$ denote the set of neighbors of v in T . Suppose $v \in V_i, 1 \leq i < i + j \leq 4d$. We claim that the number of directed paths in our oriented G that start at v and end at some member of $\bigcup_{r=1}^j V_{i+r}$ is at most 2^{3j} . Indeed, each such path must be of the form $v, v_{i_1}, v_{i_2} \dots v_{i_r}$, where

$$i < i_1 < i_2 < \dots < i_r \leq i + j, \quad v_{i_1} \in V_{i_1}, \dots, v_{i_r} \in V_{i_r}.$$

There are 2^j possibilities for choosing $i_1, i_2, \dots, i_r$, and since each vertex of the path is a neighbor of the previous one, there are at most $2^{i_1 - i + 1}$ choices for v_{i_1} , $2^{i_2 - i_1 + 1}$ choices for v_{i_2} , etc. Hence the total number of paths is smaller than 2^{3j} and thus if $v \in V_i$ then

$$\left| N_T(v) \cap \left(\bigcup_{r=1}^j V_{i+r} \right) \right| < 2^{3j}.$$

Put $r = \lfloor \frac{1}{6} \log_2(n/4d) \rfloor$, and partition the set of blocks $V_1, V_2, \dots, V_{4d}$ into $s = \lceil 4d/r \rceil$ blocks $W_1, \dots, W_s$ of consecutive V_i 's, each containing at most r blocks. By the preceding paragraph, the total number of edges of T in each block W_i is not bigger than $|W_i| \cdot 2^{3r} \leq |W_i| \sqrt{n/4d}$. Thus there are at least

$$\sum_{i=1}^s \left(\binom{|W_i|}{2} \right) - \frac{n}{4} \left(\frac{n}{4d} \right)^{1/2}$$

pairs of elements that are not adjacent in T . By the convexity of the function $g(x) = \binom{x}{2}$

$$\sum_{i=1}^s \left(\binom{|W_i|}{2} \right) \geq s \binom{n/4s}{2} = \Omega \left(\frac{n^2 \log(n/d)}{d} \right)$$

and thus T does not contain at least

$$\Omega \left(\frac{n^2}{d} \log \frac{n}{d} \right) - \frac{n}{4} \left(\frac{n}{4d} \right)^{1/2} = \Omega \left(\frac{n^2}{d} \log \frac{n}{d} \right)$$

edges. This completes the proof. $\square$

We can now prove the lower bounds in Theorems 1.1, 1.2, and 1.3.

To prove the lower bound in Theorem 1.1, consider any two rounds algorithm that sorts a set V of n elements. The first round of the algorithm consists of some set E of comparisons. Define d by $d = \lceil |E|/n \rceil$. Clearly we may assume that $d = o(n^{2/3})$ and $d = \Omega(n^{1/3})$. By Lemma 2.2 the graph $G = (V, E)$ has an acyclic orientation whose transitive closure misses $\Omega((n^2/d) \log n)$ edges. If the answers in the first round correspond to this orientation then clearly in the second round the algorithm has to compare all these $\Omega((n^2/d) \log n)$ pairs. Thus, by the trivial inequality $a + b \geq 2\sqrt{ab}$

$$\begin{aligned} c(2, n) &\geq nd + \Omega \left(\frac{n^2}{d} \log n \right) \\ &\geq \Omega(n^{3/2} (\log n)^{1/2}), \quad \text{as needed.} \end{aligned}$$

The proof of Theorem 1.3(i) is analogous. If a two rounds sorting algorithm uses $c \cdot n \log n$ comparisons in the first round, then by Lemma 2.2, it must use $\Omega(n^2/c)$ comparisons in the second round.

Theorem 1.2 is derived from the lower bound of Theorem 1.1 proved above by induction on k , starting with $k = 2$. For $k = 2$, the result is just the statement of Theorem 1.1. Suppose, by induction, that $c(k, n) \geq c_k n^{1+1/k} (\log n)^{1/k}$, where $c_k > 0$ is a constant, depending only on k . Consider an algorithm for sorting a set V of n elements in $k + 1$ rounds. Let E be the set of comparisons between pairs of elements of V made in the first round. As before, E corresponds to a set of edges of a graph $G = (V, E)$. Define d by $|E| = d \cdot n$. By a standard result from extremal graph theory (that follows, e.g., from the trivial part of Lemma 2.1), any graph with m vertices and average degree f , contains an independent set of size $\Omega(m/(1+f))$. By a repeated application of this, we conclude that G contains $\Omega(1+d)$ pairwise disjoint independent sets, each of size $\Omega(n/(1+d))$. Denote these sets by $V_1, \dots, V_s$ ($s = \Omega(1+d)$) and define $V_0 = V - \cup_{i=1}^s V_i$. Restrict our attention now only to linear orders on V for which each $v_i \in V_i$ is smaller than each $v_j \in V_j$, for all $0 \leq i < j \leq s$. Clearly, if $0 < i \leq s$, and $u, v \in V_i$ we do not have any information about the relative order of u and v from the results of the first round, and such an information can be obtained only from comparisons between elements of V_i . Thus, in the next k rounds, all the sets $V_1, \dots, V_s$ have to be sorted. By the induction hypothesis the number of comparisons for this task is at least

$$\sum_{i=1}^s c_k |V_i|^{1+1/k} (\log |V_i|)^{1/k} \geq \Omega_k \left((1+d) \cdot \left(\frac{n}{1+d} \right)^{1+1/k} \cdot \left(\log \left(\frac{n}{1+d} \right) \right)^{1/k} \right).$$

The total number of comparisons is thus at least

$$nd + \Omega \left(n^{1+1/k} \left(\log \frac{n}{1+d} \right)^{1/k} / (1+d)^{1/k} \right).$$

We can easily check that this number is $\geq \Omega(n^{1+1/(k+1)} \cdot (\log n)^{1/(k+1)})$. (Indeed, at least one of the two summands must be that big.) This completes the induction and Theorem 1.2 follows. $\square$

2.4. Lower bounds for selecting. In this section we prove the lower bound in Theorem 1.4.

LEMMA 2.3. *Any algorithm for finding the median in two rounds that uses at most nd comparisons in the first round where $\frac{1}{6} \log n < d = o(n)$ must use in the worst case $\Omega(n^2 \log^2(n/d)/d^2)$ comparisons in the second round.*

Proof. As usual we look at the elements as vertices and the comparisons as edges. Then we look at $G = (V, E)$, which is the first round graph. By Lemma 2.1 there is a subset W , $|W| = \lceil n/4 \rceil$ of V and a proper $4d$ vertex coloring of the induced subgraph G on W with color classes $V_1, \dots, V_{4d}$, satisfying the conclusions of the lemma. Let $|V_i| = x_i$, $r = \lfloor \frac{1}{6} \log_2(n/(4d)) \rfloor \leq d$. We claim that there is a t , $0 \leq t \leq 4d - r$ and r successive classes $V_{t+1}, \dots, V_{t+r}$ such that $l_t = \sum_{i=t+1}^{t+r} x_i \geq nr/20d$. The claim follows from

$$l_0 + l_r + l_{2r} + \dots + l_{r(\lfloor 4d/r \rfloor - 1)} + (l_{4d-r}) \geq \sum_{i=1}^{4d} x_i \geq \frac{n}{4}.$$

In this sum there are $\lfloor 4d/r \rfloor + 1$ elements; therefore there must be t and l such that $l = l_t \geq (n/4)/(\lfloor 4d/r \rfloor + 1) \geq nr/(4(4d+r)) \geq nr/(4 \cdot 5d) = nr/20d$ as claimed.

Take an arbitrary set of vertices of size $\lfloor n/2 - (\sum_{i=1}^t x_i + l/2) \rfloor$ of $V' = V - W$ and define it as V_0 . (This is possible since $|V_0| \leq \lfloor n/2 \rfloor \leq \lfloor 3n/4 \rfloor = |V'|$.) Define $V_{4d+1} =$

$V' - V_0$. Now orient each edge (u, v) of G with $u \in V_i, v \in V_j$ for $0 \leq i < j \leq 4d + 1$ from u to v (i.e., u is smaller than v).

The other edges of V (that join two members of V_0 or two elements of V_{4d+1}) will be oriented in an arbitrary acyclic order. Put

$$A = \bigcup_{i=0}^t V_i, \quad B = \bigcup_{i=t+1}^{t+r} V_i, \quad C = \bigcup_{i=t+r+1}^{4d+1} V_i,$$

$$|A| = \left\lfloor \frac{n-l}{2} \right\rfloor, \quad |B| = l, \quad |C| = \left\lfloor \frac{n-l}{2} \right\rfloor.$$

We now let the algorithm know for free the total order of A and C and that for every $v_a \in A, v_b \in B, v_c \in C, v_a < v_b < v_c$.

We now prove that for each $u \in B$ there are less than $l/4$ elements of B that are known to be smaller than u and less than $l/4$ elements of B that are known to be greater than u . To this end let T be the transitive closure of the above oriented graph. For $v \in V$, let $N_T(v)$ denote the set of neighbors of v in T . Suppose that $v \in V_i, t+1 \leq i < i+j \leq t+r$. We claim that the number of directed paths in our oriented G that start at v and end at some member of $\bigcup_{s=1}^j V_{i+s}$ is at most 2^{3j} . Indeed, each such path must be of the form $v_{i_1}, \dots, v_{i_s}$, where $v_{i_1} \in V_{i_1}, \dots, v_{i_s} \in V_{i_s}$. There are 2^j possibilities for choosing $i_1, i_2, \dots, i_s$ and since each vertex of the path is a neighbor of the previous one there are at most 2^{i_1-i+1} choices for v_{i_1} , etc. Hence the total number of paths is smaller than 2^{3j} and thus if $v \in V_i$ then $|N_T(v) \cap (\bigcup_{s=1}^j V_{i+s})| < 2^{3j}$. As $t+1 \leq i < i+j \leq t+r, j < r$ we conclude that the number of elements of B known to be less than v is less than $2^{3r} \leq 2^{(1/2)\log_2(n/4d)} = \sqrt{n/4d} < \frac{1}{4}nr/20d = \frac{1}{4}l$. (In the last inequality we assume that n is large enough, and that $d = o(n)$.) The number of elements of B known to be greater than v can be bounded analogously. Note that the condition for two un-compared elements to possibly be a median and an element next to the median is that together they have less than $n/2$ elements known to be less than at least one of them, and less than $n/2$ elements known to be greater than at least one of them. Therefore, we must compare each pair of such elements in the second round, since, otherwise, even knowing all the order apart from the relative order of these two might still leave each of them a possible candidate for the median. Suppose $u, v \in B$. There are less than $2 \cdot l/4 = l/2$ elements of B and $|A|$ elements of A (which sum up to be less than $l/2 + |A| = n/2$ elements) that are known to be smaller than at least one of u on v . Similarly there are less than $n/2$ elements that are known to be greater than at least one of them. If u and v are not adjacent in T , they satisfy the previous condition and therefore must be compared in the second round.

For each element $v \in B$, there are at least $l - 2 \cdot l/4 = l/2$ elements of B that are not adjacent to it in T . Altogether, there are l elements in B , therefore there are at least $\frac{1}{2} \cdot l \cdot l/2 = l^2/4$ comparisons we must perform in the second round, which means $\Omega(n^2 r^2 / d^2) = \Omega(n^2 \log^2(n/d) / d^2)$. This completes the proof. $\square$

We now prove the lower bound of Theorem 1.4 for $k = 2$.

PROPOSITION 2.4. $s(2, n) = \Omega(n^{4/3} \log^{2/3} n)$

Proof. Consider any two rounds algorithm that finds the median of a set V of n elements. The first round consists of some set E of comparisons. If

$$|E| = \Omega(n^{4/3} \log^{2/3} n),$$

there is nothing to prove. Therefore $|E| = O(n^{4/3} \log^{2/3} n)$. Define $d = \lceil |E|/n \rceil$. By the previous lemma we need $\Omega(n^2 \log^2(n/d) / d^2) = \Omega(n^2 \log^2 n / (n^{1/3} \log^{2/3} n)^2) = \Omega(n^{4/3} \log^{2/3} n)$ comparisons in the second round, which completes the proof. $\square$

Next we prove the lower bound in Theorem 1.4 by induction on k . For $k = 2$ this is just the previous proposition. Consider an algorithm for finding the median of a set V of n elements in $k + 1$ rounds. Let E be the set of comparisons between pairs of elements of V made in the first round. As before, E corresponds to a set of edges of a graph $G = (V, E)$. Define $d = \lceil |E|/n \rceil$. By a standard result from extremal graph theory, any graph with n vertices and average degree $\leq 2d$ contains an independent set V' of size $l = \Omega(n/(1 + 2d))$. Now we restrict our attention to linear orders on V for which the set of ranks of V' is $[(n - l)/2, (n + l)/2]$, and all the other elements are known to be greater or smaller than each element of V' . Clearly we cannot get any information on the order in V' and the median by comparisons including elements out of V' . In V' all the orders are possible and we should find the median there in the next k rounds. Therefore by the induction hypothesis,

$$\begin{aligned} S(k + 1, n) &\geq n(d - 1) + s(k, l) = n(d - 1) + \Omega(l^{1 + 1/(2^k - 1)} \log^{2/(2^k - 1)} l) \\ &= (d - 1)n + \Omega\left(\left(\frac{n}{1 + 2d}\right)^{1 + 1/(2^k - 1)} \log^{2/(2^k - 1)} \left(\frac{n}{1 + 2d}\right)\right) \\ &\geq \Omega(n^{1 + 1/(2^{k+1} - 1)} \log^{2/(2^{k+1} - 1)} n). \end{aligned}$$

(Indeed, at least one of the last two summands must be at least that big.) This completes the induction and the proof. $\square$

3. The upper bounds. In this section we prove the upper bounds in Theorems 1.1 and 1.3. The proof of the upper bounds in Theorem 1.4 is similar, and is omitted. In some of the probabilistic lemmas proven below we can simplify the computations a little by applying Chernoff's inequality (cf., e.g., [ES74]). As this simplification is not essential we prefer to use only the direct estimate $\binom{n}{k} \leq (ea/b)^b$. We start with a few lemmas. The first one (which is not essential but somewhat simplifies the proof) is the following result, which is proved in [AKSS86a] (see also [Pi87]).

LEMMA 3.1. *Let G be a graph in which any two disjoint sets of vertices are joined by an edge. Then for every set X containing at least $5a$ vertices, there exists a set Y disjoint from X and containing at most a vertices such that every set Z disjoint from $X \cup Y$ of $z \leq a$ vertices has at least $2z$ neighbors in X .* $\square$

Our next lemma deals with random graphs. Let $G = (V, E)$ be a random graph on n vertices in which each edge is independently present with probability p , where $p = (100 \log n)/a$ and $a > 100 \log n$ will be chosen later. In what follows we always assume that n is sufficiently large, and when we say that G has a property P "almost surely" we mean that the probability that G does not satisfy P tends to zero as n tends to infinity.

LEMMA 3.2. *G satisfies the following almost surely;*

- (i) *The number of edges of G is at most $200n^2 \log n/a$.*
- (ii) *There are no two disjoint sets $A, B \subseteq V$, with $|A| = |B| = a$ so that each $b \in B$ has less than $\log n$ neighbors in A .*
- (iii) *There are no two disjoint sets $X, Y \subseteq V$, with $|X| = x \leq a/e^{\sqrt{\log n}}$ and $|Y| = \lfloor x \log^{1/4} n \rfloor$ such that each $v \in X$ has at least $\log n$ neighbors in Y .*

Proof. Part (i) is trivial and follows, e.g., from Chebyshev's inequality and the fact that the mean and the variance of the number of edges in G are $\binom{n}{2}p < 100n^2 \log n/a$ and $\binom{n}{2}p(1 - p) < 100n^2 \log n/a$, respectively. (It also follows, of course, from the standard estimates of binomial distributions.) To prove (ii), fix a subset A of cardinality a of V and let $b \notin A$ be a vertex. Let E_b denote the event that b has less than $\log n$ neighbors in

A. Clearly $\text{Prob}(E_b) = \sum_{0 \leq i < \log n} b(a, i, p)$ where $b(a, i, p) = \binom{a}{i} p^i (1-p)^{a-i}$. We can easily check that for all $0 \leq i < \log n$, $b(a, i+1, p) > 2b(a, i, p)$, and hence

$$\begin{aligned} \text{Prob}(E_b) &\leq 2b(a, \lfloor \log n \rfloor, p) \leq 2 \left(\frac{ea}{\log n} \right)^{\log n} \cdot \left(\frac{100 \log n}{a} \right)^{\log n} \left(1 - \frac{100 \log n}{a} \right)^{a - \log n} \\ &\leq 2 \cdot (100e)^{\log n} \cdot e^{-100 \log n + 100 \log^2 n/a} < \frac{1}{n^{10}}, \end{aligned}$$

where the last inequality holds for all sufficiently large n , taking into account the fact that $a > 100 \log n$. It follows that the probability that there are $A, B \subseteq V$ with $|A| = |B| = a$ so that each $b \in B$ has less than $\log n$ neighbors in A is at most

$$\binom{n}{a} \binom{n}{a} \cdot \frac{1}{n^{10a}} < \frac{1}{n^{8a}} \rightarrow 0 \quad \text{as } n \rightarrow \infty.$$

This completes the proof of (ii).

The probability that G contains two sets X and Y as in part (iii) is at most

$$\binom{n}{x} \binom{n}{x \log^{1/4} n} \cdot \left[\binom{x \log^{1/4} n}{\log n} \cdot \left(\frac{100 \log n}{a} \right)^{\log n} \right]^x.$$

Indeed the first two binomial coefficients bound the number of possible sets X and Y , respectively, and the last factor bounds the probability that each $v \in X$ has at least $\log n$ neighbors in Y . This probability is bounded by

$$n^{2x \log^{1/4} n} (ex)^{x \log n} \cdot \left(\frac{100 \log n}{a} \right)^{x \log n} \leq n^{2x \log^{1/4} n} \left(\frac{100e \log n}{e^{\log^{1/2} n}} \right)^{x \log n} \rightarrow 0$$

as n tends to infinity. This completes the proof of the lemma. $\square$

The next lemma is similar though somewhat more complicated than the key lemma in [Pi87].

LEMMA 3.3. *If n elements are compared according to the edges of a graph G that satisfies all the assertions of Lemma 3.2 then, for every rank m , all but at most $O(a \log n / (\log \log n))$ elements will be known to be too small or too large to have this rank.*

Proof. Put $b = \lceil \sqrt{\log n \log_2 e} \rceil$ and $c = \lceil 4 \log n / \log \log n \rceil$. We show that after all the comparisons all but at most $a(1 + 6b + 2c) = O(a \log n / \log \log n)$ of the elements with rank greater than m will be known to have rank greater than m . Combining this with an analogous argument for the elements with rank less than m we get the desired result.

If $m > n - a(1 + 6b + 2c)$ there is nothing to prove. Otherwise, let C_0 contain the smallest m elements, let C_1 contain the next $6a \cdot b$ elements, let C_2 contain the next $2a \cdot c$ elements, and let C_3 contain the remaining elements. Since $|C_1 \cup C_2| = 6ab + 2ac$ it suffices to prove that C_3 contains at most a elements whose rank is not known to be greater than m . Let $C_1 = C_{1,1} \cup C_{1,2} \cdots \cup C_{1,b}$ be a partition of C_1 into b pairwise disjoint sets of $6a$ consecutive elements each, where $C_{1,1}$ is the set of smallest $6a$ elements of C_1 , $C_{1,2}$ the next $6a$ elements of C_1 , etc. Similarly, let $C_2 = C_{2,1} \cup \cdots \cup C_{2,c}$ be a partition of C_2 into c pairwise disjoint sets of $2a$ consecutive elements each, where $C_{2,1}$ is the set of smallest $2a$ elements of C_2 , $C_{2,2}$ the next $2a$ elements of C_2 , etc. We now classify the elements in $C_1 \cup C_2$ as “good” or “bad” as follows. All the elements of $C_{1,1}$ are good. Suppose $1 \leq i < b$ and all the elements of $C_{1,i}$ have already been classified, and that there are at least $5a$ good elements in $C_{1,i}$. To classify the elements of $C_{1,i+1}$, apply Lemma

3.1 with X being the set of good elements in $C_{1,i}$. (Notice that by Lemma 3.2(ii), G satisfies the assumptions of Lemma 3.1.) Classify as bad those elements of $C_{1,i+1}$ that belong to the resulting set Y , whose size is at most a . The other elements of $C_{1,i+1}$, whose number is at least $5a$, are good. This completes the classification of elements in C_1 . Now put $C_{2,0} = C_{1,b}$. Suppose $0 \leq i < c$ and suppose all elements of $C_{2,i}$ have already been classified and that there are at least a good elements in $C_{2,i}$. An element of $C_{2,i+1}$ is good if it has at least $\log n$ good neighbors in $C_{2,i}$. Otherwise, it is bad. Since G satisfies all the assertions of Lemma 3.2 we conclude, by (ii), that there are at least a good elements in $C_{2,i+1}$. Moreover, by Lemma 3.2(iii), any set X of $x \leq a/e^{\sqrt{\log n}}$ good elements in $C_{2,i+1}$ has at least $x \log^{1/4} n$ good neighbors in $C_{2,i}$. Similarly, by Lemma 3.1, any set Z of $z \leq a$ good elements in $C_{1,i+1}$ has at least $2z$ good neighbors in $C_{1,i}$.

Since in G there is an edge between any two disjoint sets of size a , and since there are at least a good elements in $C_{2,c}$, it follows that all but at most a elements in C_3 have at least one good neighbor in $C_{2,c}$. Let $v \in C_3$ be such an element, and let $\phi \neq X$ be the set of its good neighbors in $C_{2,c}$. Clearly, v is known to be larger than all members of X . If $|X| < a/e^{\sqrt{\log n}}$ it has at least $|X| \cdot \log^{1/4} n$ good neighbors in $C_{2,c-1}$, and by transitivity, v is known to be larger than these elements as well. Continuing in this manner, we conclude that v is known to be larger than at least $a/e^{\sqrt{\log n}}$ good elements of $C_{2,0} = C_{1,b}$. Let Z be the set of these elements. If $|Z| \leq a$, Z has at least $2|Z|$ good neighbors in $C_{1,b-1}$. Continuing in this manner, we conclude that v is known to be larger than at least a elements in $C_{1,1}$, and hence it is known to be larger than all but at most a elements in C_0 . Consequently, the rank of v is known to be greater than m . This discussion was true for all but at most a elements of C_3 . Hence, there are at most $a + |C_1 \cup C_2|$ elements whose rank is greater than m , that are not known to have rank greater than m . This completes the proof of the lemma. $\square$

COROLLARY 3.4. *If n elements are compared according to the edges of a graph G as in Lemma 3.2, then for every interval $[i, j]$ of $O(a \log n / \log \log n)$ successive ranks there are only $O(a \log n / \log \log n)$ elements whose ranks can possibly belong to this interval. Hence, we can form $O(n \log \log n / a \log n)$ (not necessarily disjoint) subsets of elements of size $O(a \log n / \log \log n)$ each, so that the total ordering will be known by sorting each of these sets separately.*

Proof. By the previous lemma, all but $O(a \log n / \log \log n)$ of the elements with ranks less than i are known to have ranks less than i , and all but $O(a \log n / \log \log n)$ of the elements with ranks more than j are known to have ranks more than j . As only $O(a \log n / \log \log n)$ elements have ranks in the interval $[i, j]$ there are only $O(a \log n / \log \log n)$ candidates for the interval. By partitioning all ranks into $t = O(n \log \log n / a \log n)$ contiguous intervals of size $O(a \log n / \log \log n)$ each we split the sorting problem into t smaller subproblems, as needed. $\square$

Proof of the Upper Bound of Theorem 1.1. Let G be a graph as in Lemma 3.2 with $a = \lfloor n^{1/2} (\log \log n)^{1/2} \rfloor$. In the first round of the algorithm compare the elements according to G . This gives $O(n^2 \log n / a)$ comparisons. By the last corollary, the number of comparisons left for the second round is at most $O(n \log \log n / a \log n \cdot (a \log n / \log \log n)^2)$. Altogether we get

$$c(2, n) \leq O(n^2 \log n / a + na \log n / \log \log n) = O(n^{3/2} \log n / (\log \log n)^{1/2}). \quad \square$$

The proof of Theorem 1.3(ii) is similar to that of Theorem 1.1. Instead of Lemma 3.2 we need a similar lemma for random graphs. Define $a = \lfloor n / \log n \rfloor$ and $p = 100 \log n \log \log n / n$. Let $G = (V, E)$ be a random graph on n vertices in which each edge is independently present with probability p . Again, we always assume that n is sufficiently large.

LEMMA 3.5. *G satisfies the following almost surely:*

- (i) *The number of edges of G is at most $200n \log n \log \log n$.*
- (ii) *There are no two disjoint sets $A, B \subseteq V$, with $|A| = |B| = a$, so that each $b \in B$ has less than $\log \log n$ neighbors in A .*

- (iii) *There are no two disjoint sets $X, Y \subseteq V$, with $|X| = x = a/r \leq a/\log^2 n$ and $|Y| = \lfloor x(\log \log n)^{1/2} \rfloor$ such that each $v \in X$ has at least $\log \log n$ neighbors in Y .*

Proof. Part (i) is trivial and follows, e.g., from Chebyshev's inequality. To prove (ii), fix $A \subseteq V$, $|A| = a$, and $b \in V \setminus A$ and let E_b denote the event that b has less than $\log \log n$ neighbors in A . Clearly $\text{Prob}(E_b) = \sum_{0 \leq i < \log \log n} b(a, i, p)$ where $b(a, i, p) = \binom{a}{i} p^i (1-p)^{a-i}$. Since $b(a, i+1, p) > 2b(a, i, p)$, for all $0 \leq i < \log \log n$,

$$\text{Prob}(E_b) \leq 2b(a, \lfloor \log \log n \rfloor, p) \leq 2 \cdot \left(\frac{ea}{\log \log n} \right)^{\log \log n} p^{\log \log n} (1-p)^{a-\log \log n} < e^{-50 \log \log n}.$$

Therefore, the probability that there are $A, B \subseteq V$ as in (ii) is at most

$$\binom{n}{a} \binom{n}{a} \cdot e^{-50a \log \log n} \leq (e \log n)^{2a} \cdot e^{-50a \log \log n} \rightarrow 0$$

as $n \rightarrow \infty$. This establishes (ii).

The probability that G contains two sets X and Y as in (iii) can be bounded by

$$\binom{n}{x} \binom{n}{x(\log \log n)^{1/2}} \cdot \left[\binom{x(\log \log n)^{1/2}}{\log \log n} \cdot p^{\log \log n} \right]^x.$$

Here the first two binomial coefficients bound the number of possible sets X and Y , respectively, and the last factor bounds the probability that each $v \in X$ has at least $\log \log n$ neighbors in Y . This probability is at most

$$\left(\frac{er \log n}{(\log \log n)^{1/2}} \right)^{2n(\log \log n)^{1/2}/r \log n} \cdot \left(\frac{en}{r \log n (\log \log n)^{1/2}} \cdot \frac{100 \log n \log \log n}{n} \right)^{n \log \log n / r \log n},$$

which tends to zero as n tends to infinity, since $r > \log^2 n$. This completes the proof. $\square$

The proof of the next lemma is analogous to that of Lemma 3.3 (simply replace Lemma 3.2 by Lemma 3.5, replace b by $b' = \lceil \log_2 \log^2 n \rceil$, and c by

$$c' = \lceil 2 \log n \log \log \log n \rceil.$$

We omit the details.

LEMMA 3.6. *If n elements are compared according to the edges of a graph G that satisfies all the assertions of Lemma 3.5 then, for every rank m , all but at most $O(a \log n / \log \log \log n) = O(n / \log \log \log n)$ elements will be known to be too small or too large to have the rank.*

Theorem 1.3(ii) follows from the last lemma. As in Corollary 3.4 we observe that if n elements are compared according to the $O(n \log n \log \log n)$ edges of a graph G as in Lemma 3.5, then for every interval of $O(n / \log \log \log n)$ successive ranks there are only $O(n / \log \log \log n)$ possible candidates. It then follows, as in Corollary 3.4, that at most $O(n^2 / \log \log \log n) = o(n^2)$ pairs can possibly remain unknown.

REFERENCES

- [AA87] N. ALON AND Y. AZAR, *The average complexity of deterministic and randomized parallel comparison sorting algorithms*, in Proc. 28th Annual IEEE Foundations of Computer Science, Los Angeles, CA, 1987, pp. 489–498; also: SIAM J Comput., to appear.
- [AA88] ———, *Finding an approximate maximum*, SIAM J. Comput., to appear.

- [AAV86] N. ALON, Y. AZAR, AND U. VISHKIN, *Tight complexity bounds for parallel comparison sorting*, in Proc. 27th Annual IEEE Foundations of Computer Science, Toronto, Ontario, Canada, 1986, pp. 502–510.
- [AKSS86a] M. AJTAI ET AL., *Deterministic selection in $O(\log \log n)$ parallel time*, in Proc. 18th Annual ACM Symposium on Theory of Computing, Berkeley, CA, 1986, pp. 188–195.
- [AKSS86b] ———, *Almost sorting in one round*, Adv. Comput. Res., to appear.
- [Ak85] S. AKL, *Parallel Sorting Algorithms*, Academic Press, New York, 1985.
- [Al85] N. ALON, *Expanders, sorting in rounds and superconcentrators of limited depth*, in Proc. 17th Annual ACM Symposium on Theory of Computing, Providence, Rhode Island, 1985, pp. 98–102.
- [AV87] Y. AZAR AND U. VISHKIN, *Tight comparison bounds on the complexity of parallel sorting*, SIAM J. Comput., 3 (1987), pp. 458–464.
- [BB87] B. BOLLOBÁS AND G. BRIGHTWELL, *Graphs whose every transitive orientation contains almost every relation*, Israel J. Math., 59 (1987), pp. 112–128.
- [BHe85] B. BOLLOBÁS AND P. HELL, *Sorting and graphs*, in Graphs and Orders, I. Rival, ed., D. Reidel, Boston, MA, 1985, pp. 169–184.
- [Bo86] B. BOLLOBÁS, *Random Graphs*, Academic Press, New York, 1986, Chap. 15.
- [BH82] A. BORODIN AND J. E. HOPCROFT, *Routing, merging and sorting on parallel models of computation*, in Proc. 14th Annual ACM Symposium on Theory of Computing, San Francisco, 1982, pp. 338–344.
- [BR81] B. BOLLOBÁS AND M. ROSENFELD, *Sorting in one round*, Israel J. Math., 38 (1981), pp. 154–160.
- [BT83] B. BOLLOBÁS AND A. THOMASON, *Parallel sorting*, Discrete Appl. Math., 6 (1983), pp. 1–11.
- [ES74] P. ERDŐS AND J. SPENCER, *Probabilistic Methods in Combinatorics*, Academic Press, New York, 1974, p. 18.
- [HH80] R. HÄGGKVIST AND P. HELL, *Graphs and parallel comparison algorithms*, Congress Numer., 29 (1980), pp. 497–509.
- [HH81] ———, *Parallel sorting with constant time for comparisons*, SIAM J. Comput., 10 (1981), pp. 465–472.
- [Kn73] D. E. KNUTH, *The Art of Computer Programming, Vol. 3: Sorting and Searching*, Addison Wesley, Reading, MA, 1973.
- [Pi85] N. PIPPENGER, *Explicit construction of highly expanding graphs*, preprint (1985).
- [Pi87] ———, *Sorting and selecting in rounds*, SIAM J. Comput. 16 (1987), pp. 1032–1038.
- [Va75] L. G. VALIANT, *Parallelism in comparison problems*, SIAM J. Comput., 4 (1975), pp. 348–355.

THE COMPLEXITY OF COLOURING BY SEMICOMplete DIGRAPHS*

JØRGEN BANG-JENSEN†, PAVOL HELL‡, AND GARY MACGILLIVRAY‡

Abstract. The following problem, known as the H -colouring problem, is studied. An H -colouring of a directed graph D is a mapping $f: V(D) \rightarrow V(H)$ such that $(f(x), f(y))$ is an edge of H whenever (x, y) is an edge of D . The H -colouring problem is the following. Instance: A directed graph D . Question: Does there exist an H -colouring of D ? In this paper it is shown that for semicomplete digraphs T the T -colouring problem is NP-complete when T has more than one directed cycle, and polynomially decidable otherwise.

Key words. graph colouring, graph homomorphism, complexity, polynomial algorithm, NP-completeness, tournaments, semicomplete digraphs

AMS(MOS) subject classifications. 05C15, 05C20, 68Q25

1. Introduction and summary. Let H be fixed directed graph. An H -colouring of a directed graph D (or a homomorphism $D \rightarrow H$) is a mapping $f: V(D) \rightarrow V(H)$ such that $(f(x), f(y))$ is an edge of H whenever (x, y) is an edge of D . The term H -colouring arises from the fact that an n -colouring of an undirected graph G is just a homomorphism $G \rightarrow K_n$. In analogy with this traditional terminology, if $f(x) = y$ we say that x has been coloured by y . The H -colouring problem is described as follows:

Instance: A directed graph D .

Question: Does there exist an H -colouring of D ?

The H -colouring problem is clearly in NP for any fixed H . (If H is given as part of the instance we have the graph homomorphism problem, which is known to be NP-complete [4].)

The H -colouring problem has received some attention in the literature [1], [2], [5], [6], [8], [10]. In particular, Hell and Nesetril [6] have proved that the H -colouring problem (for undirected graphs) is NP-complete for any fixed H with chromatic number at least three, and is polynomial otherwise. For directed graphs, Maurer et al. [8] have shown that the H -colouring problem is in P when H is a directed cycle or a transitive tournament, but that $C_{n,1}$ -colouring is NP-complete for any odd integer n greater than two. (The digraph $C_{n,1}$ is formed by joining the opposite vertices of a directed two-cycle by a directed path of length $n - 1$.) In [7], on the one hand, polynomial algorithms are given for the case when H is a member of a large family of oriented paths, while, on the other hand, it is shown that there are balanced acyclic digraphs for which the H -colouring problem is NP-complete. At the present time there have been no conjectures as to which digraphs give rise to NP-complete H -colouring problems and none seem to suggest themselves.

In this paper we study the H -colouring problem when H is a semicomplete digraph. We are able to classify all semicomplete digraphs (and hence all tournaments) according to whether the H -colouring problem is NP-complete or admits solution by a polynomial time algorithm. In particular we prove that the T -colouring problem (where T is a semi-

* Received by the editors February 15, 1988; accepted for publication March 9, 1988.

† Department of Mathematics and Computer Science, Odense University, DK-5230 Odense M, Denmark. This work was done while the author was visiting the School of Computing Science at Simon Fraser University with support from the Danish Research Academy and the Natural Sciences and Engineering Research Council.

‡ Simon Fraser University, Burnaby, British Columbia, Canada V5A 1S6. Support from the Natural Sciences and Engineering Research Council of Canada is gratefully acknowledged.

complete digraph) is NP-complete if T has at least two directed cycles and polynomial otherwise (Theorem 4.1). We also prove the NP-completeness of the H -colouring problem when H is the special digraph consisting of a directed k -cycle C_k and a directed l -cycle C_l and all edges between C_k and C_l directed toward the l -cycle (Theorem 3.6).

2. Terminology and preliminaries. A *digraph* D consists of a pair $V(D), E(D)$, where $V(D)$ is a finite set of *vertices* and $E(D)$ is a set of ordered vertex pairs xy , called *edges*. (In our definition of a digraph we do not allow multiple edges in the same direction between two vertices.) An *oriented* graph is a digraph without directed cycles of length 2. A *semicomplete* digraph is a digraph without nonadjacent vertices. A *tournament* is an oriented graph without nonadjacent vertices. Thus tournaments are a special subclass of the semicomplete digraphs.

If xy is an edge of the digraph D then we say that x *dominates* y ; we will use the notation $x \rightarrow y$ to denote this. If we also have that $y \rightarrow x$, i.e., if there is a double edge between x and y , then we let $[x, y]$ denote this double edge. If A and B are subsets of $V(D)$, such that all edges between vertices in A and vertices in B are directed toward B , then we use the notation $A \rightarrow B$ to denote this.

For any subset A of $V(D) \cup E(D)$, $D - A$ denotes the subgraph obtained by deleting all vertices of A and their incident edges and then deleting the edges of A still present. We write $D - x$ instead of $D - \{x\}$ when $x \in V(D) \cup E(D)$.

The subgraph *induced* by a vertex set A of D is defined as $D - (V(D) \setminus A)$ and is denoted by $D(A)$.

We may write $x \in D$ instead of $x \in V(D)$ or $x \in E(D)$, but the meaning will always be clear.

A *directed path* is a digraph with vertex set $\{x_1, x_2, \dots, x_n\}$ and edge set $\{x_1 \rightarrow x_2, x_2 \rightarrow x_3, \dots, x_{n-1} \rightarrow x_n\}$, such that all vertices and edges shown are distinct. We call such a directed path an (x_1, x_n) -path and denote it by $x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_n$. In the following an (x, y) -path always means a directed path from x to y .

A *directed cycle* is defined as a directed path, the only difference being that the endvertices are the same. An *oriented cycle* is any orientation of the edges of an undirected cycle. A digraph is *acyclic* if it has no directed cycle. The *net length* of an oriented cycle is the absolute value of the number of forward edges minus the number of backward edges, going clockwise around the cycle. In the following the word cycle will always mean oriented cycle. When directed cycles are discussed, we state this explicitly. We will sometimes use the term *double edge* to denote a directed cycle of length 2.

A *directed walk* is defined as a directed path, only now we allow repetition of vertices and edges. We let O_v^i (respectively, I_v^i), $i \geq 1$, denote the set of vertices u in $V(D)$ for which D has a (v, u) -walk (respectively, a (u, v) -walk) of length i . When $i = 1$ we just write O_v , respectively, I_v . Let $d^+(v) = |O_v|$, respectively, $d^-(v) = |I_v|$. We call $d^+(v)$ (respectively, $d^-(v)$) the *outdegree* (respectively, *indegree*) of v in D .

A *source* (respectively, a *sink*) is a vertex of indegree (respectively, outdegree) zero.

A *strong component* D' of a digraph D is a maximal subdigraph, such that for any two vertices $x, y \in D'$, D' contains an (x, y) -path and a (y, x) -path. A digraph D is *strong* or *strongly connected* if it has only one component. D is *k-connected* if for any set A of at most $k - 1$ vertices, $D - A$ is strong. If a digraph is not strong, then we can label its (strong) components $D_1, D_2, \dots, D_k$, $k \geq 2$, such that no vertex of D_i dominates any vertex of D_j if $j < i$. For general digraphs this labeling is not unique, but it is so for semicomplete digraphs. Thus here we can talk of the *initial* component of D , D_1 , and the *terminal* component D_k . The remaining components (if any) are called *intermediate* components.

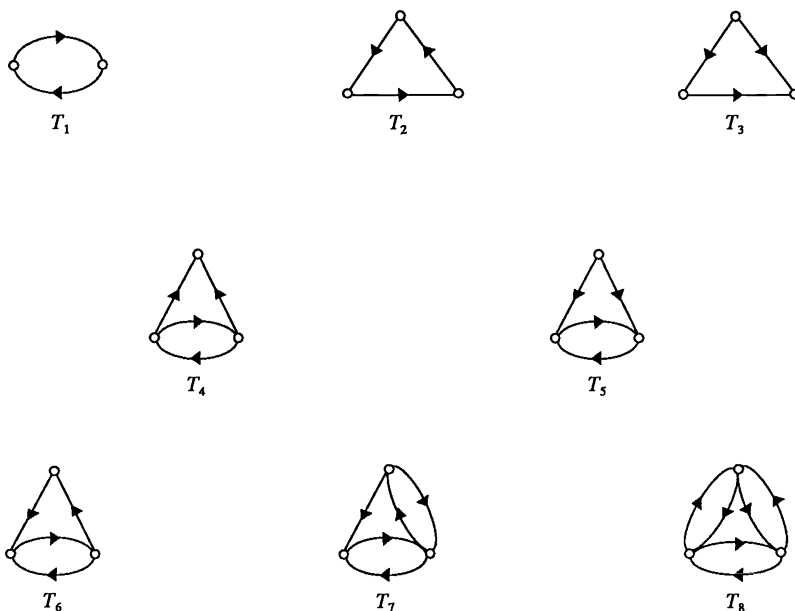


FIG. 2.1

A digraph is called *transitive* if for every pair of edges $x \rightarrow y, y \rightarrow z$ in $E(D)$ with x, y, z different vertices, the edge $x \rightarrow z$ also belongs to $E(D)$.

The following two theorems are taken from [8].

THEOREM 2.1. *If T is a transitive tournament then the T -colouring problem is polynomially decidable.*

THEOREM 2.2. *Let T_i denote one of the eight semicomplete digraphs in Fig. 2.1. If $i \leq 5$ then the T_i -colouring problem is polynomially decidable. If $i \geq 6$ then the T_i -colouring problem is NP-complete.*

We note that Theorem 2.2 conforms (for $|T| \leq 3$) to our characterization for semicomplete digraphs.

One very important construction we shall be using frequently in our proofs is the following construction used in [6]. If H is a subdigraph of H' , then a *retraction* of H' to H is a homomorphism $r: H' \rightarrow H$ such that $r(h) = h$ for all vertices of H . A digraph is a *core* if it does not admit a retraction to a proper subdigraph. We note that every semicomplete digraph is a core.

Let J be a fixed digraph, with specified vertices j and $k_1, k_2, \dots, k_t$. The *subindicator construction* with respect to $j, k_1, k_2, \dots, k_t$ transforms a core H with specified vertices $h_1, h_2, \dots, h_t$ into its subdigraph $H \sim$ induced by the vertex set $V \sim$ defined as follows: Let W be the digraph obtained from the disjoint union of J and H by identifying each k_i with the corresponding h_i ($i = 1, 2, \dots, t$). A vertex v of H belongs to $V \sim$ just in case there is a retraction of W mapping j to v .

LEMMA 2.3 [6]. *Let H be a core. If the $H \sim$ -colouring problem is NP-complete, then so is the H -colouring problem.*

Lemma 2.3 will be a very important tool in our proofs in §§ 3 and 4.

In the next lemma C_k denotes the directed cycle of length k .

LEMMA 2.4 [5]. *A digraph D is C_k -colourable if and only if the net length of every cycle in D is divisible by k .*

Lemma 2.4 yields a polynomial algorithm for the C_k -colouring problem. A polynomial algorithm for this problem is also given in [8].

3. Some lemmas needed for the proof of the theorem. In this section we prove some lemmas we will need for the proof of the main theorem in the next section. Some of these are special cases that must be treated separately. Theorem 3.6 is of interest in its own right.

LEMMA 3.1. *Let T be a semicomplete digraph for which T -colouring is polynomial and let T' (respectively, T'') be obtained from T by adding a source (respectively, sink). Then T' -colouring (respectively, T'' -colouring) is also polynomial.*

Proof. Let D be a digraph; D is T' -colourable if and only if the digraph D' obtained from D by deleting all sources is T -colourable: To extend a T -colouring of D' to a T' -colouring of D , colour all sources in D by the source that was adjoined to T in order to construct T' . Then no two such vertices are adjacent and any colour from T , i.e., vertex of T , is admissible for any of their neighbours. Conversely, any T' -colouring of D restricted to D' is a T -colouring, because the outdegree of each vertex of D' is at least one. Since D' may be constructed and tested for T -colourability in polynomial time, T' -colourability is in P . An analogous argument shows that T'' -colouring is polynomial.

Of the tournaments on four vertices, only one has more than one directed cycle, namely the tournament T_4^1 in Fig. 3.1.

LEMMA 3.2. *The T_4^1 -colouring problem is NP-complete.*

Proof. We prove the NP-completeness by reducing 3-SAT to the T_4^1 -colouring problem.

CLAIM 3.2.1. *The tournament T_4^1 has*

- (a) *exactly two directed three-cycles, namely 023 and 013.*
- (b) $O_0^5 = \{1, 2, 3\}$.
- (c) $O_1^5 = \{0, 2, 3\}$.

Proof. The proof is omitted. (Note that T_4^1 cannot have a closed directed walk of length 5.)

Our proof that T_4 -colouring is NP-complete uses a “component design” construction (see [4]). The two components we need are the graphs X^1 and Y^1 shown in Fig. 3.2. In the next two claims we determine those properties of these graphs necessary for the proof.

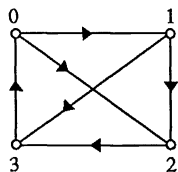


FIG. 3.1. The tournament T_4^1 .

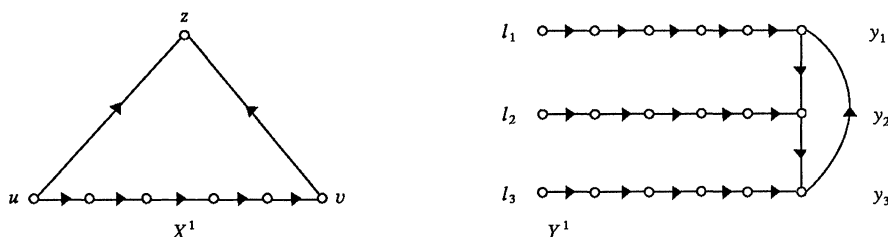


FIG. 3.2

CLAIM 3.2.2. In any T_4^1 -colouring of X^1 where colour $(z) = 2$ either

- (a) colour $(u) = 0$ and colour $(v) = 1$, or
- (b) colour $(u) = 1$ and colour $(v) = 0$.

Furthermore, the partial colourings (a) and (b) can both be extended to T_4^1 -colourings of X^1 .

Proof. Suppose colour $(z) = 2$. Then the vertices u and v must be coloured using initial vertices of edges that terminate at 2 (i.e., 0 or 1). Furthermore, Claim 3.2.1(b) and (c) together imply that the two colours used must be distinct. The last part of the claim is easily verified from Claim 3.2.1.

CLAIM 3.2.3. There is no T_4^1 -colouring of Y^1 in which the vertices l_1 , l_2 and l_3 are all coloured 0. Any colouring of the vertices l_1 , l_2 , and l_3 by 0 and 1, such that at least one 1 is used, can be extended to a T_4^1 -colouring of Y^1 .

Proof. In any T_4^1 -colouring of Y^1 , the vertices y_1 , y_2 , and y_3 must map onto a directed three-cycle of T_4^1 . If l_1 , l_2 , and l_3 are all coloured 0 then, by Claim 3.2.1(b), y_1 , y_2 , and y_3 must be coloured using 1, 2, and 3. As these three vertices do not induce a directed three-cycle of T_4^1 , it follows that Y^1 has no T_4^1 -colouring of this type. For the second part of the claim we may assume without loss of generality that l_1 is coloured 1. The five-path to y_1 can be coloured so that vertex y_1 is coloured 0. By Claim 3.2.1(b) and (c) the vertices y_2 and y_3 can be coloured 2 and 3, respectively, regardless of the colour (0 or 1) assigned to l_2 and l_3 . Since 023 is a directed three-cycle of T_4^1 , the proof is complete.

Now we are ready to prove Lemma 3.2. We have already noted that H -colouring is in NP for any fixed H . We now show that 3-SAT polynomially transforms to T_4^1 -colouring.

Let an instance of 3-SAT be given, with variables $x_1, x_2, \dots, x_n$ and clauses $C_1, C_2, \dots, C_m$. We construct a digraph D (cf. Fig. 3.3) as follows. We take a copy X_i of X^1 for each variable x_i ($i = 1, 2, \dots, n$), a copy Y_j of Y^1 for each clause C_j ($j = 1, 2, \dots, m$), and one copy of T_4^1 ; then we identify all the vertices z of all the X_i 's with the vertex 2 of T_4^1 , and, for each clause $C_j = l_1 \vee l_2 \vee l_3$ ($j = 1, 2, \dots, m$) we identify each vertex labeled l_k with the vertex u_i of X_i if $l_k = x_i$ and with the vertex v_i of X_i if $l_k = \neg x_i$.

The construction of D may clearly be carried out in polynomial time. We claim that D is T_4^1 -colourable if and only if $C_1, C_2, \dots, C_m$ are simultaneously satisfiable.

Suppose D is T_4^1 -colourable. Since T_4^1 has no nontrivial automorphisms, the vertex z in each copy X_i of X^1 is coloured 2. Thus, for each i , one of the vertices u_i or v_i is

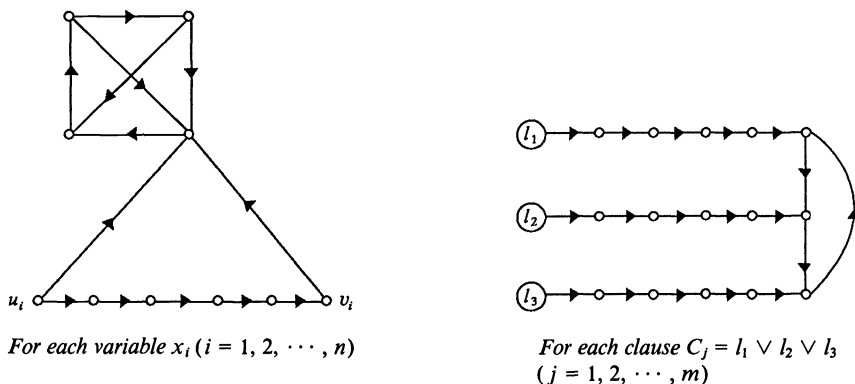
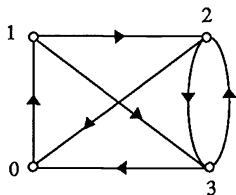


FIG. 3.3. The digraph D .

FIG. 3.4. The semicomplete digraph T_4^2 .

coloured 0 and the other is coloured 1. This means that in each copy Y_j of Y^1 the vertices l_1 , l_2 , and l_3 are coloured using 0 and 1. Moreover, Claim 3.2.2 guarantees that at least one of them is coloured with 1. A satisfying truth assignment can now be constructed by giving x_i the value true if and only if vertex u_i is coloured 1.

On the other hand, suppose that A is a satisfying truth assignment for $C_1, C_2, \dots, C_m$. We define a partial colouring of D by colouring vertex u_i with colour one if and only if variable x_i is true. This extends easily to a T_4^1 -colouring of all X_i 's and, by the second part of Claim 3.2.3, it can be extended to a T_4^1 -colouring of D .

Therefore 3-SAT polynomially transforms to T_4^1 -colouring and so T_4^1 -colouring is NP-complete.

The proofs of the next two lemmas use the same ideas as the proof above. In each case we state the claims necessary for the proof and leave the details to the reader.

Let T_4^2 be the semicomplete digraph in Fig. 3.4.

LEMMA 3.3. The T_4^2 -colouring problem is NP-complete.

Proof.

CLAIM 3.3.1. In T_4^2 we have

$$O_i^3 = \{0, 1, 2, 3\} \quad \text{for } i = 1, 2, 3, \quad \text{and}$$

$$O_0^3 = \{0, 2, 3\}.$$

The vertices 0, 2, 3 do not induce a directed three-cycle in T_4^2 .

CLAIM 3.3.2. Let X^2, Y^2 be the digraphs in Fig. 3.5. In any T_4^2 -colouring of X_2 either

- (a) colour $(u) = 0$ and colour $(v) = 1$, or
- (b) colour $(v) = 0$ and colour $(u) \in \{2, 3\}$.

Moreover, the partial colourings (a) and (b) can be extended to T_4^2 -colourings of X_2 . (Hint. Every directed three-cycle in T_4^2 contains the vertex 0, and 0 does not dominate a vertex in a directed two-cycle.)

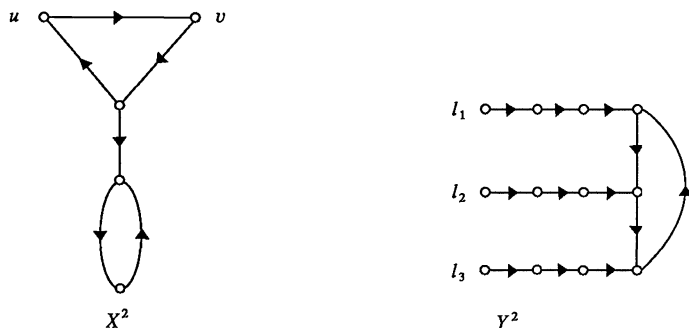
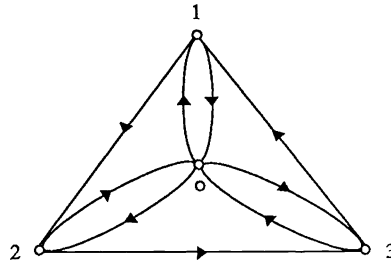


FIG. 3.5

FIG. 3.6. The semicomplete digraph T_4^3 .

CLAIM 3.3.3. *There is no T_4^2 -colouring of Y^2 in which the vertices l_1, l_2, l_3 are all coloured 0. Any colouring of the vertices l_1, l_2, l_3 by colours in $\{0, 1, 2, 3\}$ in which at least one colour from $\{1, 2, 3\}$ is used can be extended to a T_4^2 -colouring of Y^2 .*

(Hint. If, say l_1 is coloured by an element of $\{1, 2, 3\}$, then the end of the three-path starting at l_1 may be coloured 1; the other two vertices of the directed three-cycle may then be coloured by, say 0 and 2.)

Having established these claims, the reader can easily finish the proof in the same manner as in the proof of Lemma 3.2.

Note that in this proof there is no need for a special vertex like z in the proof of Lemma 3.2: Here we can use 0 to mean false and all of 1, 2, and 3 to mean true (compare with the proof of Lemma 3.2).

Let T_4^3 be the semicomplete digraph in Fig. 3.6.

LEMMA 3.4. *The T_4^3 -colouring problem is NP-complete.*

Proof. Consider the graphs X^3, Y^3 defined below in Fig. 3.7.

CLAIM 3.4.1. *In any T_4^3 -colouring of X^3 either*

- (a) colour $(u) = 0$ and colour $(v) \in \{1, 2, 3\}$, or
- (b) colour $(v) = 0$ and colour $(u) \in \{1, 2, 3\}$.

Moreover, the partial colourings (a) and (b) can be extended to T_4^3 -colourings of X_3 .

CLAIM 3.4.2. *There is no T_4^3 -colouring of Y^3 in which the vertices l_1, l_2, l_3 are all coloured zero. Any colouring of the vertices l_1, l_2, l_3 by the colours $\{0, 1, 2, 3\}$ such that at least one colour in $\{1, 2, 3\}$ is used can be extended to a T_4^3 -colouring of Y^3 .*

(Hint. The vertex z must always receive the colour 0 and it is impossible to colour the directed seven-cycle by the colours in $\{1, 2, 3\}$ only.)

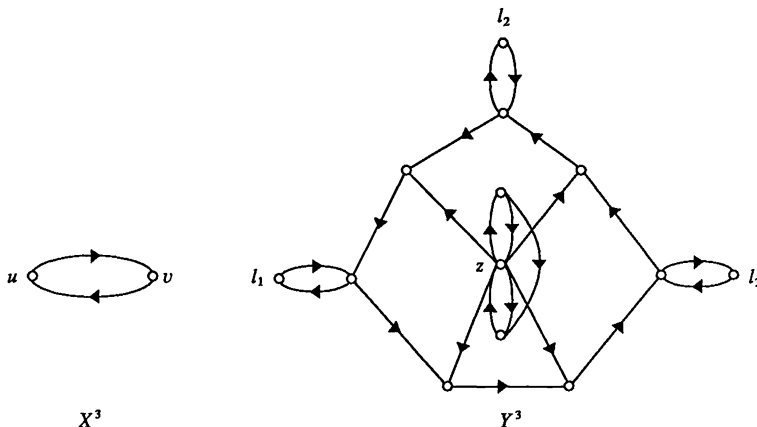


FIG. 3.7

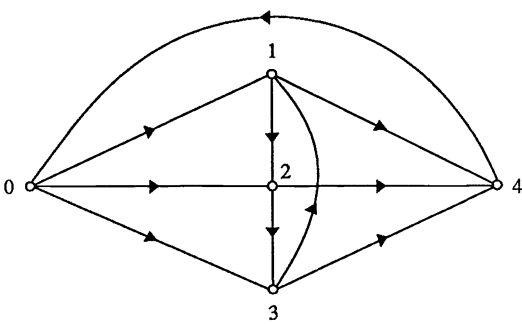


FIG. 3.8. The tournament T_5 .

Having established these claims, the reader can easily finish the proof in the same manner as in the proof of Lemma 3.2.

Let T_5 be the tournament in Fig. 3.8.

LEMMA 3.5. *The T_5 -colouring problem is NP-complete.*

Proof. The proof is similar to the ones above, only this time we use a reduction from 4-SAT. Once again we state the needed claims and leave the details to the reader.

CLAIM 3.5.1. *In T_5 we have*

$$O_i^3 = \{0, 1, 2, 3, 4\} \quad \text{for } i = 0, 1, 2, 3, \quad \text{and}$$
$$O_4^3 = \{1, 2, 3, 4\}.$$

The set $\{1, 2, 3, 4\}$ does not induce a strong tournament.

Let X and Y be the digraphs in Fig. 3.9 below.

CLAIM 3.5.2. *In any T_5 colouring of X , either*

- (a) colour $(u) = 4$ and colour $(v) \in \{1, 2, 3\}$, or
- (b) colour $(v) = 4$ and colour $(u) \in \{1, 2, 3\}$.

Moreover, the partial colourings (a) and (b) can be extended to T_5 -colourings of X .

CLAIM 3.5.3. *There is no T_5 -colouring of Y in which l_1, l_2, l_3, l_4 are all coloured 4. Any colouring of the vertices l_1, l_2, l_3, l_4 by colours in $\{1, 2, 3, 4\}$, such that at least one colour in $\{1, 2, 3\}$ is used, can be extended to a T_5 -colouring of Y .*

Having established these claims, the reader can easily finish the proof in the same manner as in the proof of Lemma 3.2.

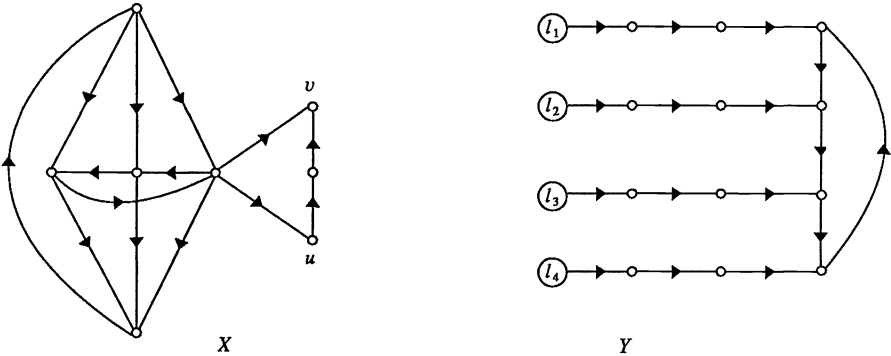
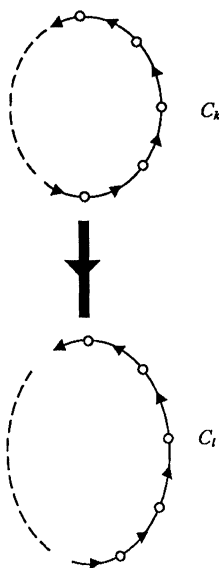


FIG. 3.9

FIG. 3.10. The digraph $D(k, l)$.

The next theorem, which is also needed in the proof of the main theorem, is of interest in its own right, since it does not deal exclusively with semicomplete digraphs.

Let $D(k, l)$ denote the digraph obtained from two directed cycles C_k (respectively, C_l) of length k (respectively, l), by adding all the edges between vertices in C_k and vertices in C_l and orienting them towards C_l . See Fig. 3.10.

THEOREM 3.6. *The $D(k, l)$ -colouring problem is NP-complete for $k, l \geq 2$.*

Proof. The proof is similar to the ones above, so we only give the necessary claims and definitions and leave the rest to the reader.

Let us call C_k the *upper directed cycle* and C_l the *lower directed cycle*.

Let X' and Y' be the digraphs in Fig. 3.11.

CLAIM 3.6.1. *In any $D(k, l)$ -colouring of X' , either*

(a) *u is coloured on the upper directed cycle and v is coloured on the lower directed cycle, or*

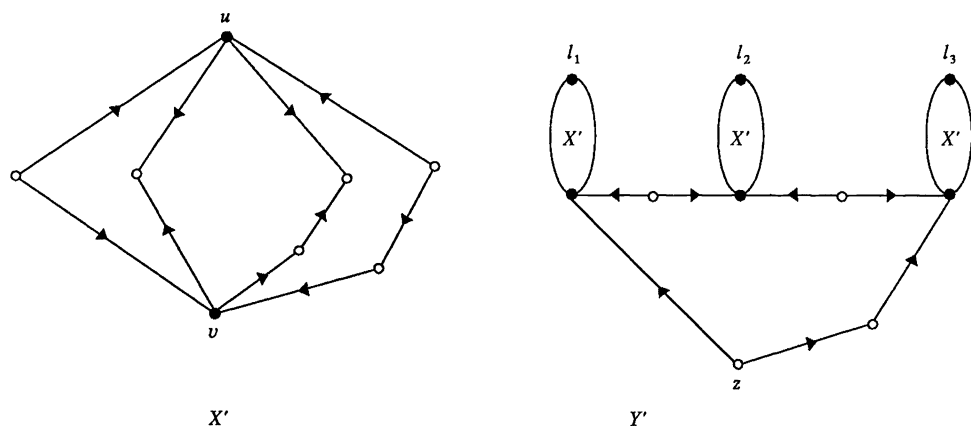


FIG. 3.11

(b) u is coloured on the lower-directed cycle and v is coloured on the upper-directed cycle.

Moreover, the partial colourings (a) and (b) can be extended to $D(k, l)$ -colourings of X' .

(Hint. If they are both coloured by the upper directed cycle then the whole outer cycle of X' must be coloured by the upper directed cycle of $D(k, l)$, but that is impossible, by Lemma 2.4, since the net length of the outer cycle of X' is 1. This is the same for the lower directed cycle.)

CLAIM 3.6.2. *There is no $D(k, l)$ -colouring of Y' such that l_1, l_2, l_3 are all coloured using the lower-directed cycle. Any assignment of colours from the upper and lower directed cycles to l_1, l_2, l_3 , such that at least one colour from the upper directed cycle is used, can be extended to a $D(k, l)$ colouring of Y' .*

(Hint. The net length of the cycle containing z is 1.)

4. The theorem. In this section we complete the characterization of all semicomplete digraphs, according to whether the H -colouring problem is NP-complete or admits a solution by a polynomial time algorithm. We do this by proving the following theorem.

THEOREM 4.1. *Let T be a semicomplete digraph. If T contains at least two directed cycles then the T -colouring problem is NP-complete. If T is acyclic or has a unique directed cycle then the T -colouring problem is in P.*

Proof. From Theorem 2.2 we may assume that $|T| \geq 4$.

If T is acyclic then T is a transitive tournament and the result follows from Theorem 2.1.

Suppose that T has a unique directed cycle. From a basic theorem on tournaments [9] (which also holds for semicomplete digraphs in general), we know that if a semicomplete digraph has a directed cycle of length $k \geq 3$, then it has directed cycles of all lengths i , $3 \leq i \leq k$. Furthermore a semicomplete digraph is strong if and only if it has a directed Hamiltonian cycle. It follows that the unique directed cycle in T is a directed two-cycle or a directed three-cycle. We also get that T is not strong, since $|T| \geq 4$. Let $T_1, \dots, T_r$ be the strong components of T . Since T is unicyclic, all but one T_i have size one. Now it is easy to see that T can be obtained from its unique directed cycle by adding a sequence of sources and sinks. Thus by Lemma 3.1 T -colouring is polynomially decidable.

Now suppose that there exists a semicomplete digraph T such that T has at least two directed cycles and the T -colouring problem is not NP-complete. Let T be chosen such that T is minimal with respect to this property; i.e., for every proper induced subdigraph T' of T , such that T' has more than one directed cycle, the T' -colouring problem is NP-complete. (Note that T' is a semicomplete digraph.) We want to derive a contradiction.

Note that the following holds for T .

(*) No vertex v of T has more than one directed cycle in either $T(O_v)$ or $T(I_v)$.

Proof. This follows from Lemma 2.3 and the assumption on T by using as a sub-indicator a path of length 1 out of v , respectively, into v . (If T' is a proper induced subdigraph of T containing at least two directed cycles then the T' -colouring problem is NP-complete, by the minimality of T .)

Case 1. T is not strong. Let $T_1, T_2, \dots, T_k$, $k \geq 2$ be the strong components of T . (T_1 denotes the initial component. Compare § 2.) By (*) it follows that each T_i has size at most 3 for $i = 1, 2, \dots, k$. (A strong semicomplete digraph on n vertices has directed cycles of all lengths between 3 and n [9].)

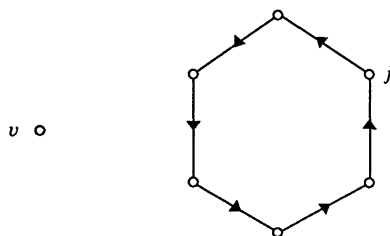


FIG. 4.1

By Theorem 3.6, and (*), $k \geq 3$. Now get from (*) that each $T_2, \dots, T_{k-1}$ has size 1. Thus T has precisely two directed cycles and no vertex in $T_2 \cup \dots \cup T_{k-1}$ is in a directed cycle. By using the disconnected subindicator shown in Fig. 4.1 we obtain the semicomplete digraph $T \sim$ induced by T_1 and T_k (v is any vertex in T). Since $T \sim$ colouring is NP-complete by Lemma 3.6 we obtain a contradiction.

This completes the proof in Case 1. Thus we may assume that T is strong.

Case 2. T is strong and every vertex is incident with a double edge.

(**) The digraph induced by the double edges in T is bipartite.

Proof. In [6] it is shown that if D is a digraph for which the subdigraph induced by the double edges in D is not bipartite, then the D -colouring problem is NP-complete.

We now derive a series of properties of T that finally produce a contradiction.

(1) Every vertex in T is incident with precisely one double edge.

Proof. Suppose that b is a vertex of T that is incident with at least two double edges. Let a and c be chosen such that b has double edges with these. By (**) there is no double edge between a and c . We may assume without loss of generality that $a \rightarrow c$.

Now look at the vertices in $T' = T - \{a, b, c\}$.

Suppose that v is a vertex of T' such that a and c both dominate v . Then the only edge between b and v is $v \rightarrow b$, since otherwise $I_v \supseteq \{a, b, c\}$ and these induce more than one directed cycle, contradicting (*). Now $O_a \supseteq \{b, c, v\}$ and they induce more than one directed cycle, contradicting (*).

Similarly we get that no vertex dominates both a and c .

Suppose that u is a vertex of T' such that a dominates u and u dominates c . Then $I_c \supseteq \{a, b, u\}$ and thus, by (*), the only edge between b and u is $b \rightarrow u$, since otherwise $T(\{a, b, u\})$ contains more than one directed cycle. Now $O_a \supseteq \{b, c, u\}$ and they induce more than one directed cycle, contradicting (*). Thus all edges between a and T' must enter a and all edges between c and T' must leave c .

Suppose first that there is some vertex x in T' , such that there is a double edge between x and b (i.e., b is incident with at least three double edges). Then, by Lemma 3.4, $|T| \geq 5$. Let y be a vertex of $T' - \{x\}$. y is incident with a double edge, and this either lies entirely in T' or goes to b . In any case I_a contains at least two double edges, contradicting (*). Thus b is incident with precisely two double edges. Since every vertex of T' is incident with a double edge and this lies entirely in T' , we get that $|T'| = 2$, since otherwise I_a contains more than one directed cycle, contradicting (*). Let $V(T') = \{u, v\}$. From (*) we easily get that b either dominates both of u and v or is dominated by both of them. Then either I_b or O_b contains the two directed three-cycles $a \rightarrow c \rightarrow u \rightarrow a$, $a \rightarrow c \rightarrow v \rightarrow a$, contradicting (*). This proves (1).

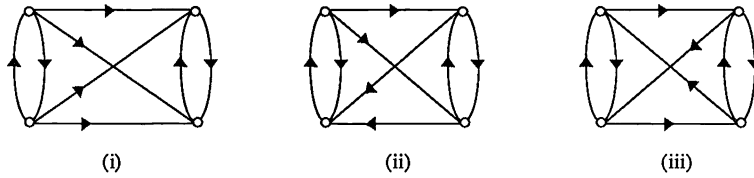


FIG. 4.2

- (2) The edges between two double edges in T must go as shown in (i)–(iii) of Fig. 4.2.

Proof. Suppose that a vertex in one of the double edges dominates both vertices in the other. Call this vertex a and let $[a, b]$ and $[c, d]$ be the double edges. Then $O_a \supseteq \{b, c, d\}$, so by (*) b must either dominate both of c, d or be dominated by both. These are (i) and (ii) in Fig. 4.2. Similarly, starting from a vertex, which is dominated by both vertices of the other double edge, we get (i) and (ii). The only remaining possibility is (iii) in Fig. 4.2.

Let $V(T) = \{a_1, b_1, \dots, a_r, b_r\}$, where $[a_i, b_i]$ is a double edge for $i = 1, 2, \dots, r$.

- (3) $r \geq 3$.

Proof. Suppose that $r = 2$. Then T is one of the semicomplete digraphs in Fig. 4.2. Then we get, by Theorem 3.6, that T is either (ii) or (iii). Figure 4.3 shows T with the appropriate subindicator used to prove that the T -colouring problem is NP-complete, contradicting the choice of T .

- (4) Every vertex v in T has $d^+(v) = d^-(v) = r$.

Proof. Clearly every vertex v has in- and outdegree between $r - 1$ and $r + 1$, since otherwise some vertex v has at least two double edges in either I_v or O_v , contradicting (*). It suffices to prove that no vertex has outdegree $r + 1$. Suppose that $d^+(a_1) = r + 1$. We may assume without loss of generality that a_1 dominates $b_1, a_2, b_2, a_3, \dots, a_r$ and is dominated by the rest of the vertices in T .

Let $T \sim$ be obtained from T by using the subindicator shown in Fig. 4.4. $V(T \sim) = \{a_1, a_2, b_2, \dots, b_r\}$ and $T \sim$ contains the double edge $[a_2, b_2]$. By Lemma 2.3 and the minimality of T we get that $T \sim$ does not contain two directed cycles. Thus we must have that $b_3, \dots, b_r$ all dominate a_2 and b_2 . Then b_1 must dominate $b_3, \dots, b_r$, since otherwise some $b_i, i \geq 3$, dominates all of the vertices in $\{a_1, b_1, a_2, b_2\}$,

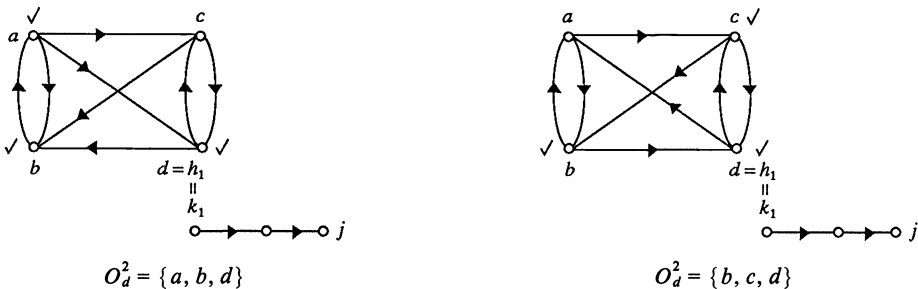
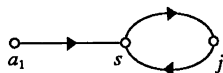


FIG. 4.3

FIG. 4.4. The subindicator $a_1 \rightarrow [s, j]$.

contradicting (*). Now using subindicator $b_3 \rightarrow [s, j]$ (compare Fig. 4.4; in the remainder of this paper, we shall use this kind of symbolic—and hopefully self-explanatory—abbreviation for very simple digraphs) we get a semicomplete digraph T^* that contains a_2, b_1, b_2, b_3 , but not a_1 . Furthermore, we must have that $a_2 \rightarrow b_1$, because otherwise a_2 is dominated by a_1, b_1 , and b_3 and they induce two directed cycles, contradicting (*). Therefore $T(\{a_2, b_1, b_2, b_3\})$ has at least two directed cycles. Using the minimality of T and Lemma 2.3 we obtain a contradiction. This contradiction proves (4).

- (5) Every vertex in T dominates exactly one vertex in each of the $r - 1$ other double edges. That is, the edges between double edges follow the pattern in Fig. 4.2 (iii).

Proof. Suppose that some vertex dominates both vertices in one of the other double edges. Then we may assume without loss of generality that a_1 dominates both a_2 and b_2 .

If T has some other double edge such that a_1 dominates precisely one vertex in this, then we may assume that this is the case for the double edge $[a_3, b_3]$. Now arguing in the same way as in the proof above, we obtain a contradiction. Thus $r = 3$ (cf. (*) and (4)) and a_3, b_3 both dominate a_1 . Now using (2) on the edges between $[a_1, b_1]$ and $[a_2, b_2]$, and then using (4), we get that T is one of T_1, T_2 in Fig. 4.5 (possibly by interchanging the names of a_3 and b_3 in T_2).

Figure 4.5 also shows the appropriate subindicators, along with the vertices, that are images of j under retractions, needed to derive a contradiction to the choice of T from Lemma 2.3 (Both of the semicomplete subdigraphs induced by the marked vertices contain more than one directed cycle.) This proves (5).

Now we may assume, by relabeling if necessary, that a_1 dominates precisely $b_1, a_2, \dots, a_r$.

Note that, since a_1 does not dominate both vertices of any double edge, the semicomplete digraph $T \sim$ obtained from T by using the subindicator $a_1 \rightarrow [u, v] \leftarrow j$ does not contain a_1 (here we mean that u and v are vertices not in T and a_1 dominates u and j dominates v). On the other hand, it is easy to check that $V(T \sim) = V(T) - a_1$, so

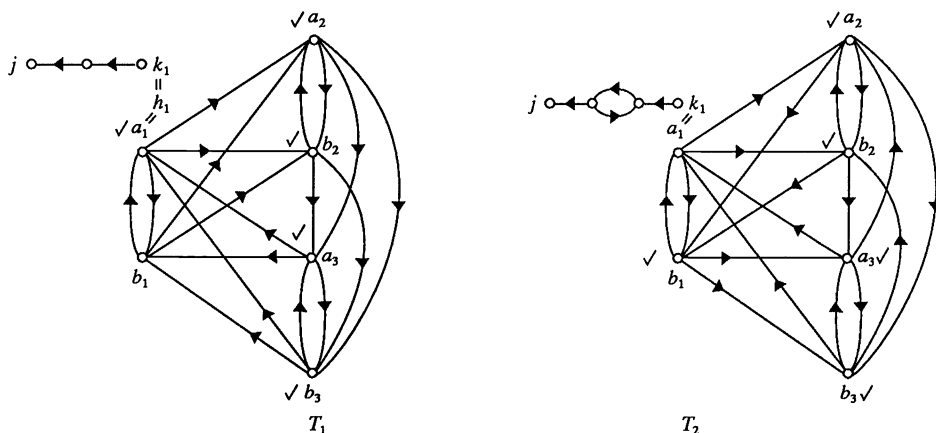


FIG. 4.5

$T \sim$ contains at least two directed cycles, contradicting the choice of T . This contradiction concludes the proof in Case 2. Thus we may assume that T has a vertex that is not incident with a double edge. This is Case 3.

Case 3. T is strong and it contains a vertex v that is not incident with a double edge.

Case 3a. T contains a double edge.

(A) T has exactly one double edge.

Proof. If T contains at least two double edges, then we use the subindicator in Fig. 4.6 to get a contradiction: Let $T \sim$ be obtained from T by using this subindicator; i.e., $T \sim$ is the proper subsemicomplete digraph induced by those vertices that are incident with a double edge. Then $T \sim$ has at least two directed cycles and does not contain v , but this is a contradiction to the choice of T , by Lemma 2.3.

Let $[a, b]$ denote the double edge in T .

(B) $T - \{a, b\}$ does not contain two vertices u, v , such that $(u \rightarrow a, b \text{ or } a, b \rightarrow u)$ and $(a \rightarrow v \rightarrow b \text{ or } b \rightarrow v \rightarrow a)$.

Proof. We show that we cannot have vertices $u \rightarrow a, b$ and $a \rightarrow v \rightarrow b$. (The proofs in the rest of the cases are analogous.) So suppose that u and v are as above. Then $u \notin O_u^2 \supseteq \{a, b, v\}$, so the semicomplete digraph $T \sim$ obtained from T , by using subindicator $u \rightarrow x \rightarrow j$ (i.e., a directed path of length two from u to j) does not contain u , but it has at least two directed cycles, contradicting the choice of T .

(C) $T - \{a, b\}$ does not contain a vertex u , such that $u \rightarrow a, b$ or $a, b \rightarrow u$.

Proof. Suppose that (C) does not hold. Let

$$I = \{x : x \rightarrow a, b\} \quad \text{and} \quad O = \{x : a, b \rightarrow x\}.$$

By (B), $V(T) = \{a, b\} \cup I \cup O$.

Since T is strong and (C) does not hold both I and O are nonempty and there is an edge from the terminal component O_k of $T(O)$ to the initial component I_1 of $T(I)$. Here we let $I_1, \dots, I_r, r \geq 1$ (respectively, $O_1, \dots, O_k, k \geq 1$) denote the strong components of $T(I)$ (respectively, $T(O)$). It is easy to see that $b \notin O_a^2 \supseteq O$. Furthermore a has a directed path of length two to at least one vertex in I_1 , since there is an edge from O_k to I_1 . Now we conclude that $|O| = 1$, since otherwise the semicomplete digraph $T \sim$, obtained from T by using subindicator $a \rightarrow x \rightarrow j$ has at least two directed cycles, but it does not contain b , contradicting the choice of T . Let $O = \{o\}$. Similarly, it is easy to see that $b \notin I_a^2 \supseteq I$. From this we conclude, using an argument similar to the one above, that $|I| = 1$. But now T is the semicomplete digraph T_4^2 in Lemma 3.3. This contradiction proves (C).

Thus every vertex in $T - \{a, b\}$ dominates exactly one of a and b . Let

$$A = \{x \in T - \{a, b\} : b \rightarrow x \rightarrow a\} \quad \text{and} \quad B = \{x \in T - \{a, b\} : a \rightarrow x \rightarrow b\}.$$

Suppose $A, B \neq \emptyset$. Now for every $x \in A, b \in O_x^2 \supseteq B$ and for every $y \in B, b \in I_y^2 \supseteq A$. From this we conclude that $|A| = |B| = 1$: Suppose that $|A| \geq 2$. Let x be a vertex of outdegree at least one in $T(A)$. Then we have $x \notin O_x^2 \supseteq \{a, b\} \cup B$.

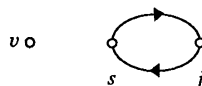


FIG. 4.6. The subindicator $v[s, j]$.

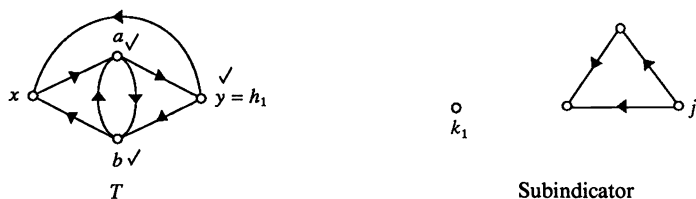


FIG. 4.7

Thus $T(O_x^2)$ contains at least two directed cycles and does not contain x , so we obtain a contradiction to the choice of T . The proof that $|B| = 1$ is similar.

Thus $|T| = 4$. Let $V(T) = \{x, a, b, y\}$ where $[a, b]$ is the double edge and $b \rightarrow x \rightarrow a, a \rightarrow y \rightarrow b$. One of x and y has outdegree one, depending on the direction of the edge between x and y ; say x has outdegree one. The other case is similar. Figure 4.7 shows T and the proper subindicator, which we can use to obtain a contradiction to the choice of T .

Thus we may assume that A or B is empty, say $B = \emptyset$. The other case is similar. Suppose that $|A| \geq 3$. Let $u \rightarrow v \rightarrow z$ be a directed path of length two in $T(A)$. Then $u \notin O_u^2 \supseteq \{z, a, b\}$, and from this we can obtain a contradiction to the choice of T . Thus $|A| = 2$ (since $|T| \geq 4$) and T is the semicomplete digraph in Fig. 4.8. This figure also shows the appropriate subindicator and the vertices that are images of j under retractions. From this we can derive a contradiction to the choice of T .

This concludes the proof in Case 3a.

Case 3b. T is a strong tournament.

By Lemma 3.2, $|T| \geq 5$.

- (I) T does not contain a vertex v such that at least one of $T(O_v)$, $T(O_v^2)$, $T(I_v)$, and $T(I_v^2)$ contains more than one directed cycle.

Proof. This follows easily, since v does not belong to any of these tournaments.

- (II) T must be one of the five tournaments in Fig. 4.9.

Suppose first that T is not two-connected. Let x be chosen such that $T - x$ is not strong and let $T_1, \dots, T_k, k \geq 2$, be the strong components of $T - x$. It is easy to see (as in Case 1) that (I) implies $|T_i| \leq 3$, for $i = 1, \dots, k$. Since T is strong, x is dominated by some vertex of T_k and dominates some vertex of T_1 . Suppose $|T_i| = |T_j| = 3$ for some $i < j$. Then we easily get, by (I), that $i = 1, j = k$ and $|T_r| = 1$ for $2 \leq r \leq k - 1$. Let $a \rightarrow b \rightarrow c \rightarrow a, u \rightarrow v \rightarrow w \rightarrow u$ denote T_1 , respectively, T_k . We may assume without loss of generality that $u \rightarrow x \rightarrow a$. Now $O_a^2 \supseteq \{x, u, v, w\}$, so $v, w \rightarrow x$ by (I) and the fact that $u \rightarrow x$. Similarly we get that $x \rightarrow b, c$. Then $O_u^2 \supseteq \{x, a, b, c, w\}$ and they induce more than one directed cycle, contradicting (I).



FIG. 4.8

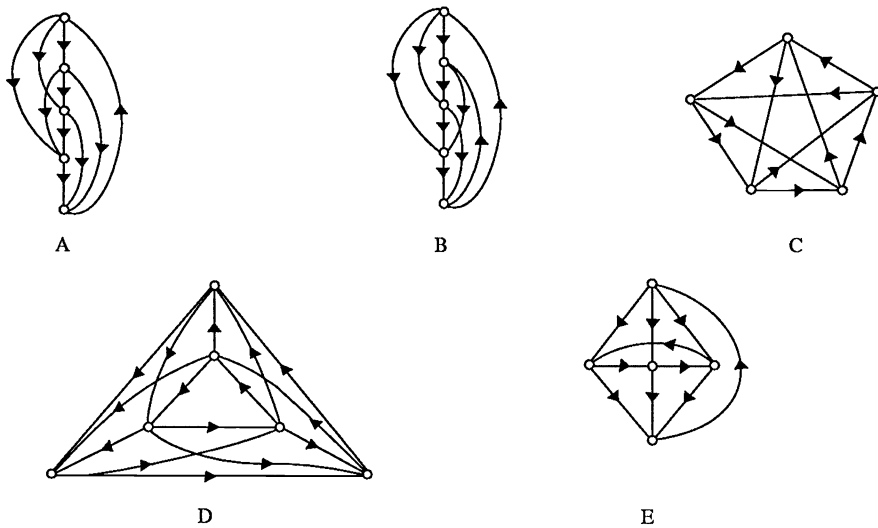


FIG. 4.9. The exceptional tournaments.

Thus at most one of the strong components of T is a directed three-cycle.

Suppose that $T_1 = a \rightarrow b \rightarrow c \rightarrow a$. Let $T_k = \{y\}$. Then we easily get, as above, that $x \rightarrow a, b, c$. If $|T| = 5$ then T is E in Fig. 4.9. Thus we may assume that $|T| \geq 6$ and therefore $k \geq 3$. Let $T_2 = \{u\}$. Then $O_a^2 \supseteq \{c, x, y, u\}$ and they induce more than one directed cycle, contradicting (I). The case where $|T_k| = 3$ is similar, and also leads only to E in Fig. 4.9.

Suppose that $T_i = a \rightarrow b \rightarrow c \rightarrow a$ for some $1 < i < k$. Let $T_1 = \{u\}$, $T_k = \{y\}$. Then $y \rightarrow x \rightarrow u$, since T is strong. As before it is easy to see that either $x \rightarrow a, b, c$ or $a, b, c \rightarrow x$. In the first case $I_c^2 \supseteq \{a, x, y, u\}$ and they induce two directed cycles. In the second case $O_a^2 \supseteq \{c, x, y, u\}$ and they induce two directed cycles. Both of these contradict (I).

Thus $T - x$ is a transitive tournament. Let $T_i = \{v_i\}$, for $i = 1, \dots, k$. Again $v_k \rightarrow x \rightarrow v_1$, since T is strong.

Suppose that $|T| \geq 6$. Then $x \rightarrow v_2$, since otherwise v_2 has directed paths of length two to all the vertices in $\{v_1, v_{k-1}, v_k, x\}$ and they induce two directed cycles, contradicting (I). Now $x \rightarrow v_3$, since otherwise v_3 has directed paths of length two to all the vertices in $\{v_1, v_2, v_k, x\}$ and they induce two directed cycles, contradicting (I). Then v_1 has directed paths of length two to all the vertices of $\{v_3, \dots, v_k, x\}$ and they induce more than one directed cycle, contradicting (I). Thus $|T| = 5$. If $(x \rightarrow v_2, v_3)$ or $(v_2, v_3 \rightarrow x)$ then T is A in Fig. 4.9. If $v_3 \rightarrow x \rightarrow v_2$ then T is B in Fig. 4.9. We cannot have that $v_2 \rightarrow x \rightarrow v_3$, since then v_2 has directed paths of length two to all the vertices in $\{x, v_1, v_3, v_4\}$ and they induce two directed cycles, contradicting (I).

Thus we may assume that T is two-connected. If $|T| = 5$ then T is C in the list, since this is the only two-connected tournament on five vertices. So, we can assume that $|T| \geq 6$. Let v be a vertex of maximum outdegree in T . Let $O = O_v$ and $I = I_v$. Then $|O| \geq 3$ and $O_v^2 \supseteq I$, by the maximality of $d^+(v)$. Let $I_1, \dots, I_k$, $k \geq 1$ (respectively, $O_1, \dots, O_q$, $q \geq 1$) be the strong components of $T(I)$ (respectively, $T(O)$). Since $T - v$ is strong, there exist vertices $r \in O_q$, $u \in I_1$ such that $r \rightarrow u$.

If $|O_1| \geq 3$ then $O_v^2 = V(T) - v$, which is strong and contains several directed cycles, contradicting (I). Thus $|O_1| = 1$. Let $O_1 = \{o_1\}$. Then $O_v^2 = V(T) - \{v, o_1\}$.

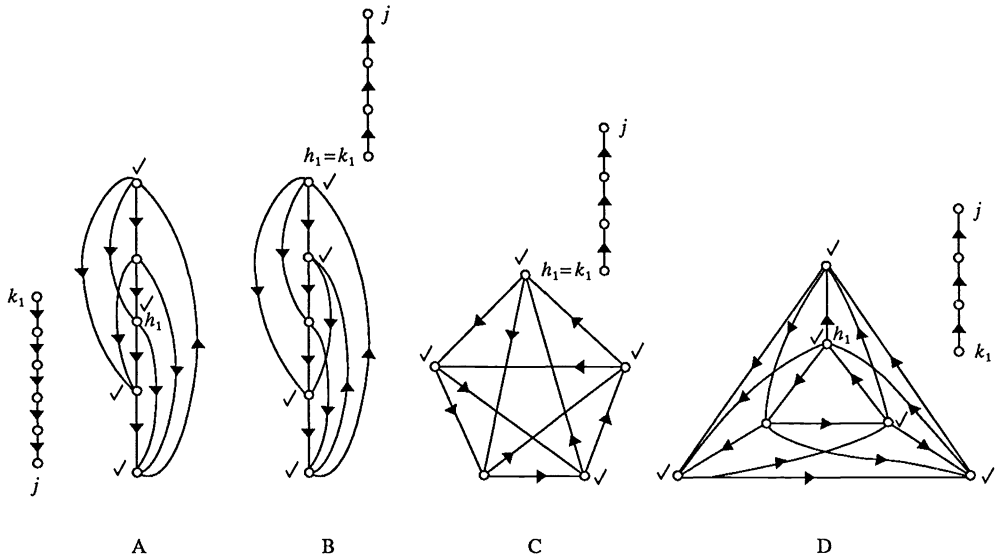


FIG. 4.10

Thus, by (I), we get that $T - \{v, o_1\}$ is not strong. This gives that $O_2 \rightarrow I_k$, or else a Hamiltonian cycle could be found in $T - \{v, o_1\}$.

Suppose that $o_1 \rightarrow z$ for some $z \in I$. Then o_1 has a directed path of length two to v and to r ($|O| \geq 3$) and o_1 also has directed paths of length two to all the vertices of $\{u, v\} \cup I_k$. Since the vertices $\{r, u, v\} \cup I_k$ induce at least two directed cycles, we obtain a contradiction to (I). Thus $I \rightarrow o_1$.

Suppose $|O| \geq 4$. Let $x \in O_2$ be chosen such that x has a directed path of length two to r . Note that if x does not have a directed path of length two to u then x and r are both in O_2 , $r \rightarrow x$ and $u \rightarrow f$ where $x \rightarrow f \rightarrow r$ is an (x, r) -path of length two. Thus $O_2^2 \supseteq \{x, f, r, u\}$ and they induce at least two directed cycles, contradicting (I). Then $O_2^2 \supseteq \{o_1, r, u, v\}$ and they induce two directed cycles, contradicting (I). Therefore $|O| = 3$ and then $|I| \leq 3$, by the maximality of O . Now $T(O)$ is a transitive triple. Let $O_i = \{o_i\}$, for $i = 1, 2, 3$. $|I_k| = 1$, since $o_2 \rightarrow I_k$ and $d^+(o_2) \leq d^+(v)$. Thus $T(I)$ is also transitive. Let $I_j = \{i_j\}$ for $j = 1, \dots, k$. All the vertices of $\{v, o_1, o_3\}$ have directed paths of length two into i_k . This shows that $k = 2$, since otherwise i_1 also has a directed path of length two into i_k and $T(\{i_1, o_1, o_3, v\})$ has two directed cycles, contradicting (I). We must have $o_3 \rightarrow i_2$, since $d^+(o_3) \geq 2$, by the two-connectivity of T . Also $o_2 \rightarrow i_1$, since $d^+(i_1) \leq 3$, by the maximality of $d^+(v)$. Now the reader can easily check that T is D in Fig. 4.9. This completes the proof of (II). If T is E in Fig. 4.9, then the T -colouring problem is NP-complete by Lemma 3.5, contradicting the choice of T . Thus T is one of A–D in Fig. 4.9. In Fig. 4.10 we indicate which subindicators to use in order to derive a contradiction to the choice of T in these cases.

We have proved that the semicomplete digraph T cannot exist and the proof of Theorem 4.1 is complete.

Acknowledgments. The authors are grateful to David Kirkpatrick and Emo Welzl for their interest and fruitful suggestions. Through E. Welzl we have learned that W. Gutjahr has solved the problem treated in this paper in the special case of a transitive tournament with added edges.

Note added in proof. In a sequel to this paper, the first two authors give several other classes of digraphs H for which the H -colouring problem is NP-hard. These include both very dense digraphs (such as the semicomplete digraphs discussed here) and very sparse digraphs. Motivated by these results, they also conjecture a sufficient condition on H which guarantees that H -colouring is NP-hard.

REFERENCES

- [1] M. ALBERTSON, P. CATLIN, AND L. GIBBONS, *Homomorphisms of 3-chromatic graphs II*, Congr. Numer., 47 (1985), pp. 19–28.
- [2] G. BLOOM AND S. BURR, *On unavoidable digraphs in orientations of graphs*, J. Graph Theory, 11 (1987), pp. 453–462.
- [3] J. A. BONDY AND U. S. R. MURTY, *Graph Theory and Applications*, American Elsevier, New York, 1976.
- [4] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability*, W. H. Freedman, San Francisco, 1979.
- [5] R. HAGGKVIST, P. HELL, D. J. MILLER, AND V. NEUMAN LARA, *On multiplicative graphs and the product conjecture*, Combinatorica, 8 (1988), pp. 71–81.
- [6] P. HELL AND J. NESETRIL, *On the complexity of H -colouring*, Computing Science Technical Report TR 86-4, Simon Fraser University, Burnaby, British Columbia, Canada, 1986.
- [7] P. HELL, J. NESETRIL, AND E. WELZL, *Recent results on directed H -colouring*, to appear.
- [8] H. A. MAURER, J. H. SUDBOROUGH, AND E. WELZL, *On the complexity of the general colouring problem*, Inform. and Control, 51 (1981), pp. 123–145.
- [9] J. W. MOON, *Topics on Tournaments*, Holt, Reinhart, Winston, New York, 1969.
- [10] J. NESETRIL AND A. PULTR, *On classes of relations and graphs determined by subobjects and factorobjects*, Discrete Math., 22 (1978), pp. 287–300.

A NEW HEURISTIC FOR PARTITIONING THE NODES OF A GRAPH*

EARL R. BARNES†, ANTHONY VANNELLI‡, AND JAMES Q. WALKER§

Abstract. There is a class of graph partitioning algorithms which improve an initial partition by interchanging nodes between the subsets of the initial partition. These algorithms tend to require long running times because usually many trials must be made before the right combination of nodes to interchange is found. In this paper we describe a fast procedure for determining sets of nodes to interchange in order to improve an initial partition.

Key words. graphs, partitioning, linear programming

AMS(MOS) subject classification. 68

1. Introduction. Let $G = \{N, E\}$ be an undirected graph having nodes $N = \{1, \dots, n\}$ and edges E . We consider the problem of partitioning N into k disjoint subsets $S_1, \dots, S_k$ of specified sizes $m_1 \geq m_2 \geq \dots \geq m_k$ satisfying $\sum_{i=1}^k m_i = n$, in such a way so as to minimize the number of edges connecting nodes in distinct subsets of the partition. Such partitioning problems arise in laying out circuits on chips [1], [2], computer program segmentation [3], design of power networks [4], and in many other areas.

A popular class of algorithms for partitioning the nodes of a graph begins with an initial partition and tries to improve it by interchanging nodes between pairs of subsets in the partition (see, for example, [5]). A shortcoming of these methods is pointed out by the following example of a nine-node graph to be partitioned into three sets containing three nodes each.

The initial partition in Fig. 1(a) can be improved by interchanging nodes 4 and 5. This leads to Fig. 1(b). The difficulty is that, in general, we do not know what sizes of node sets to interchange in order to improve a partition, and searching for the right sizes and combinations can be very time consuming. For example, there are $O(n^4)$ ways of exchanging pairs of nodes in partitioning n nodes into two equal-sized subsets.

The algorithm we propose determines the number of nodes to interchange between subsets without searching. Moreover, it performs some transfers that are more general than interchanges. For example, in Fig. 1(b) what is needed for an optimal partition is to transfer node 1 to S_3 , node 4 to S_2 , and node 7 to S_1 . Our algorithm is capable of making such transfers in one step. Of course it is a heuristic and is therefore not guaranteed to find the right transfer to make in all cases. Moreover, it is based on the assumption that the initial partition has a large number of nodes assigned correctly. We have used the algorithm successfully on a large number of examples, with starting partitions obtained by the algorithm described in [6]. Another method for obtaining initial partitions is described in [7].

* Received by the editors June 19, 1985; accepted for publication (in revised form) December 30, 1987.

† IBM T.J. Watson Research Center, Yorktown Heights, New York 10598.

‡ Department of Electrical Engineering, University of Waterloo, Waterloo, Ontario, Canada N20 L3G1. The work of this author was performed while he was a Postdoctoral Fellow in the Mathematical Sciences Department, IBM T.J. Watson Research Center, Yorktown Heights, New York 10598.

§ Department of Applied Mathematics, Yale University, New Haven, Connecticut 06520. The work of this author was performed while he was a student in the Mathematical Sciences Department, IBM T.J. Watson Research Center, Yorktown Heights, New York 10598.

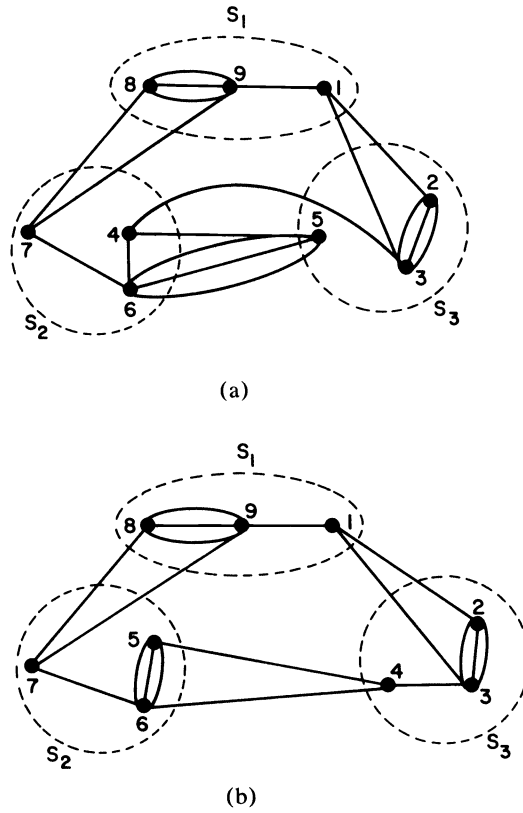


FIG. 1

2. The algorithm. Let

$$x_j = \begin{bmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{nj} \end{bmatrix}$$

be an indicator vector for the set S_j , $j = 1, \dots, k$. Thus

$$x_{ij} = \begin{cases} 1 & \text{if } i \in S_j, \\ 0 & \text{if } i \notin S_j, \end{cases} \quad j = 1, \dots, k.$$

Let a_{ij} denote the number of edges connecting nodes i and j , $i \neq j$, and let $a_{ii} = 0$, $i = 1, \dots, n$. The $n \times n$ matrix $A = (a_{ij})$ is called the adjacency matrix for G .

The number of edges with both endpoints in S_j is given by

$$(2.1) \quad \frac{1}{2} \sum_{r=1}^n \sum_{s=1}^n a_{rs} x_{rj} x_{sj} = \frac{1}{2} x_j^T A x_j.$$

Since there are m_j nodes in S_j , we have

$$\sum_{i=1}^n x_{ij} = m_j, \quad j = 1, \dots, k.$$

Also, since $S_i \cap S_j = \emptyset$,

$$\sum_{j=1}^k x_{ij} = 1, \quad i = 1, \dots, n.$$

Since the total number of edges in G is fixed, we can minimize the number of edges cut in a partition by maximizing the number of edges not cut. Thus our problem is equivalent to the 0-1 quadratic programming problem:

$$\begin{aligned} & \text{maximize} \quad \sum_{j=1}^k x_j^T A x_j \\ & \text{subject to} \quad \sum_{i=1}^n x_{ij} = m_j, \quad j = 1, \dots, k, \\ & \quad \quad \quad \sum_{j=1}^k x_{ij} = 1, \quad i = 1, \dots, n, \quad x_{ij} = 0 \text{ or } 1. \end{aligned}$$

For our purposes we replace the matrix A by a positive semidefinite matrix $A + D$ where D is a diagonal matrix. To see that this has no effect on the solution of our quadratic programming problem, note that

$$\sum_{j=1}^k x_j^T D x_j = \sum_{j=1}^k \sum_{i=1}^n d_i x_{ij}^2 = \sum_{i=1}^n d_i.$$

Thus

$$\sum_{j=1}^k x_j^T (A + D) x_j = \sum_{j=1}^k x_j^T A x_j + \sum_{i=1}^n d_i.$$

From now on we assume the diagonal of A has been modified so that A is positive semidefinite. An appropriate diagonal matrix D is described below. The method that we use for modifying the diagonal of A is due to Carter [8]. His modification attempts to minimize $\sum_{i=1}^n d_i$.

Let $A = QQ^T$ denote a Cholesky factorization of A into the product of a lower and an upper triangular matrix. Denote row r of Q by q_r . We then have

$$a_{rs} = q_r q_s^T, \quad r, s = 1, \dots, n.$$

Let $N = S_1 \cup \dots \cup S_k$ denote an initial partition. Then

$$\begin{aligned} (2.2) \quad \sum_{r=1}^n \sum_{s=1}^n \|q_r - q_s\|^2 x_{rj} x_{sj} &= \sum_{r=1}^n \sum_{s=1}^n \{ \|q_r\|^2 - 2q_r q_s^T + \|q_s\|^2 \} x_{rj} x_{sj} \\ &= 2 \sum_{r=1}^n \|q_r\|^2 m_j x_{rj} - 2x_j^T A x_j. \end{aligned}$$

Let

$$e_j = \frac{1}{m_j} \sum_{r=1}^n q_r x_{rj}$$

denote the mean of the set S_j . Then

$$\begin{aligned}
 (2.3) \quad \sum_{r=1}^n \sum_{s=1}^n \|q_r - q_s\|^2 x_{rj} x_{sj} &= \sum_{r=1}^n \sum_{s=1}^n \{ \|(q_r - e_j) - (q_s - e_j)\|^2 \} x_{rj} x_{sj} \\
 &= 2 \sum_{r=1}^n \|q_r - e_j\|^2 m_j x_{rj}.
 \end{aligned}$$

Combining this with (2.2), we see that

$$(2.4) \quad \sum_{j=1}^k \sum_{r=1}^n \{ \|q_r - e_j\|^2 - \|q_r\|^2 \} m_j x_{rj} = - \sum_{j=1}^k x_j^T A x_j.$$

This can be thought of as a linearization of the objective function in (2.1). Strictly speaking, it is not a linearization since e_j depends on x_j for each j . However, if we hold the e_j 's fixed and vary the x_j 's slightly, (2.4) remains satisfied approximately. So, with the e_j 's held fixed, we take the linear function on the left in (2.4) as an approximation of the quadratic function on the right.

THEOREM 1. *Let the q_r 's and e_j 's be chosen as in (2.4) and let (y_{rj}) be a solution of the transportation problem:*

$$\begin{aligned}
 (2.5) \quad &\text{minimize} \quad \sum_{j=1}^k \sum_{r=1}^n \{ \|q_r - e_j\|^2 - \|q_r\|^2 \} m_j y_{rj} \\
 &\text{subject to} \quad \sum_{r=1}^n y_{rj} = m_j, \quad j = 1, \dots, k, \\
 &\quad \sum_{j=1}^k y_{rj} = 1, \quad r = 1, \dots, n, \quad y_{rj} \geq 0, \quad r = 1, \dots, n, \quad j = 1, \dots, k.
 \end{aligned}$$

If the minimum is strictly less than $-\sum_{j=1}^k x_j^T A x_j$, then

$$\sum_{j=1}^k y_j^T A y_j > \sum_{j=1}^k x_j^T A x_j$$

and the partition $\bar{S}_1 \cup \dots \cup \bar{S}_k$, where

$$\bar{S}_j = \{r \mid y_{rj} = 1\}, \quad j = 1, \dots, k,$$

cuts fewer edges than the initial partition $S_1 \cup \dots \cup S_k$.

Proof. Let (y_{rj}) be a solution of (2.5). If

$$\begin{aligned}
 \sum_{j=1}^k \sum_{r=1}^n \{ \|q_r - e_j\|^2 - \|q_r\|^2 \} m_j y_{rj} &< \sum_{j=1}^k \sum_{r=1}^n \{ \|q_r - e_j\|^2 - \|q_r\|^2 \} m_j x_{rj} \\
 &= - \sum_{j=1}^k x_j^T A x_j,
 \end{aligned}$$

then

$$\min \sum_{j=1}^k \sum_{r=1}^n \{ \|q_r - \mu_j\|^2 - \|q_r\|^2 \} m_j y_{rj} < - \sum_{j=1}^k x_j^T A x_j$$

where this minimum is with respect to the means $\mu_j, j = 1, \dots, k$. This minimum occurs at

$$\bar{e}_j = \frac{1}{m_j} \sum_{r=1}^n q_r y_{rj}, \quad j = 1, \dots, k.$$

By analogy with the derivation of (2.4) we have

$$\sum_{j=1}^k \sum_{r=1}^n \{ \|q_r - \bar{e}_j\|^2 - \|q_r\|^2 \} m_j y_{rj} = - \sum_{j=1}^k y_j^T A y_j < - \sum_{j=1}^k x_j^T A x_j.$$

This proves the first assertion of the theorem. The fact that (y_{rj}) determines a better partition than (x_{rj}) follows from (2.1).

The basic idea of our algorithm is now clear.

THE ALGORITHM.

Step 1. Given an initial partition $S_j = \{r | x_{rj} = 1\}, j = 1, \dots, k$, we define

$$e_j = \frac{1}{m_j} \sum_{r=1}^n q_r x_{rj}, \quad j = 1, \dots, k$$

and solve (2.5). Denote the solution by (y_{rj}) .

Step 2. If the partition determined by (y_{rj}) is the same as a partition already computed, stop. The algorithm cannot improve the partition any further. If the partition determined by (y_{rj}) is new, take it as the initial partition and repeat Step 1.

THEOREM 2. *The algorithm converges.*

Proof. Since the number of partitions of G is finite the algorithm will eventually repeat a partition and terminate.

To complete the description of our algorithm we must explain how we choose the diagonal elements of A to make A positive semidefinite. There are many ways of doing this. For example, it suffices to choose the diagonal elements of A large enough to make A diagonally dominant. We have no strong recommendation for how to choose the diagonal of A to simplify the problem of clustering the points q_r . In our work we have used a method due to Carter [8]. This method attempts to minimize $\sum a_{rr}$ while choosing the a_{rr} large enough to make A positive definite. These methods tend to minimize the modification that must be made in A to ensure $A > 0$. Whether this helps us in the clustering problem is not clear.

Here is a brief description of Carter's algorithm.

Step 1. For $j = 1, 2, \dots, n-1$

(i) Compute $p_i = a_{ij} - \sum_{r=1}^{j-1} q_{jr} q_{ir} \quad (i = j+1, j+2, \dots, n)$

(ii) Let $q_{jj} = \left(\sum_{i=j+1}^n p_i^2 \right)^{1/4}$

(iii) $q_{ij} = \frac{p_i}{q_{jj}} \quad (i = j+1, j+2, \dots, n)$

(iv) The diagonal element is given by

$$a_{jj} = \sum_{i=1}^j q_{ji}^2$$

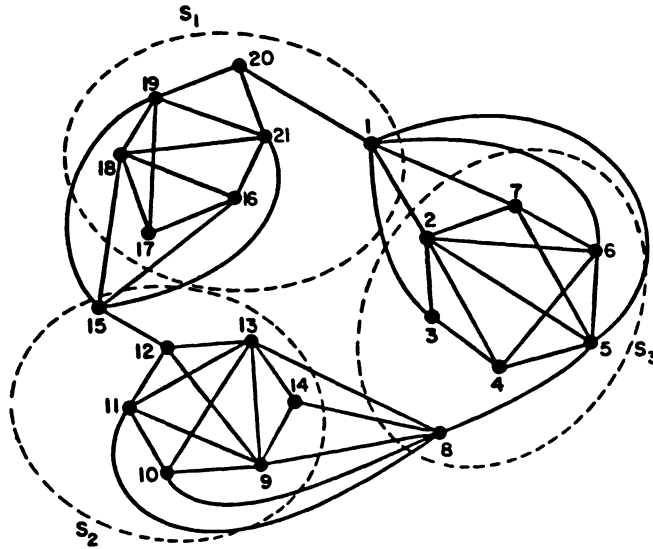


FIG. 2

fact that the q vectors cluster according to the optimal partition is clear from the matrix Q .

REFERENCES

- [1] R. L. RUSSO, P. H. ODEN, AND P. K. WOLFF, SR., *A heuristic procedure for the partitioning and mapping of computer logic blocks to modules*, IEEE Trans. Comput., 20 (1971), pp. 1455-1462.
- [2] H. R. CHARNEY AND D. L. PLATO, *Efficient partitioning of components*, Share/ACM/IEEE Design Automation Workshop, Washington, DC, July 1968.
- [3] L. W. COMEAU, *A study of user program optimization in a paging system*, ACM Symposium on Operating System Principles, Gatlinburg, TN, October 1967.
- [4] J. G. LEE, W. G. VOGT, AND M. H. MICKLE, *Optimal decomposition of large-scale networks*, IEEE Trans. System Man Cybernet., 9 (1979), pp. 369-375.
- [5] B. W. KERNIGHAM AND S. LIN, *An efficient procedure for partitioning graphs*, Bell System Tech. J., (1970), pp. 291-307.
- [6] E. R. BARNES, *An algorithm for partitioning the nodes of a graph*, SIAM J. Algebraic Discrete Methods, 3 (1982), pp. 541-550.
- [7] A. VANNELLI AND M. VIDYASAGAR, *Three approximate solution techniques for the optimal k -decomposition problem*, International Conference on Systems, Management, and Cybernetics, Delhi and Bombay, India, January 1984.
- [8] M. CARTER, *The indefinite zero-one quadratic problem*, Discrete Appl. Math., 7 (1984), pp. 23-44.

A CLASS OF POLYNOMIALLY SOLVABLE SET-COVERING PROBLEMS*

P. BERTOLAZZI† AND A. SASSANO†

Abstract. A clutter L is a collection of m subsets of a ground set $E(L) = \{x_1, \dots, x_n\}$ with the property that for every pair $A_i, A_j \in L$, A_i is neither contained nor contains A_j . A transversal of L is a subset of $E(L)$ having at least one element in common with each member of L .

The problem of finding the minimum weight transversal of a clutter L is equivalent to the well-known *set-covering problem*.

In this paper the class of *ideal* clutters that properly contains the class of clutters the members of which are the bases of a *matroid* (*matroidal* clutters) is introduced.

An ideal clutter L has the property that the number of its minimal transversals is bounded by a polynomial in m and n . The properties of ideal clutters are described, and two polynomial algorithms for recognizing them and finding their minimum weight transversal are presented.

Key words. clutters, set-covering, matroids

AMS(MOS) subject classifications. 05C70

1. Introduction. A clutter L is a collection of m subsets of a ground set $E(L) = \{x_1, \dots, x_n\}$ with the property that for every pair $A_i, A_j \in L$, A_i is neither contained nor contains A_j . A transversal of L is a subset of $E(L)$ having nonempty intersection with each member of L . The blocker $b(L)$ of L is the collection of all the minimal (with respect to set inclusion) transversals of L .

To each clutter L we can associate a *proper* $(0, 1)$ matrix M_L the rows of which are the incidence vectors in $E(L)$ of the members of L (*incidence matrix* of L).

There is a transparent relationship between the minimal transversals of a given clutter L and the feasible solutions of the following *set-covering problem* associated to the $(0, 1)$ matrix M_L :

$$(SC) \quad \min \{c^T x \mid M_L x \geq e_m, \quad x_j \in \{0, 1\}, \quad j = 1, \dots, n\}$$

where c is a positive weight vector associated to the elements of $E(L)$ and e_m denotes the vector with m ones. In particular we have that the incidence vectors of the transversals of L are the feasible solutions of the problem (SC), and that the problem of finding a minimum weight transversal of L is equivalent to (SC).

The set-covering problem is a well-known, NP-hard, optimization problem, and a partial list of its applications includes: airline crew scheduling, plant location, information retrieval, and switching circuit design. Both theoretical and algorithmic aspects of this problem have been extensively studied in the literature and, in particular, considerable attention has been addressed to the problem of characterizing those $(0, 1)$ matrices (clutters), the associated set-covering problems of which are solvable in polynomial time (polynomial in the input size mn).

To this class belong the clutters having the "Max-Flow Min-Cut" (MFMC) property [8]. A clutter has the MFMC property if all the vertices of the polyhedron $Q_{LP}(M_L) = \{x : M_L x \geq e_m, x \geq 0\}$ are integral. This is the case, for example, of the clutters the members of which are the edge sets of the *paths* joining two distinct nodes of a graph $G = (V, E)$ (s-*t* paths).

* Received by the editors December 8, 1986; accepted for publication (in revised form) February 29, 1988.

† Istituto di Analisi dei Sistemi ed Informatica del CNR, Viale Manzoni 30, 00185 Roma, Italy.

A major characteristic of a clutter L having the MFMC property is that the associated polyhedron $Q(M_L)$ can be described by a polynomial number ($m + n$) of *inequalities* (facets).

In this paper, following a “dual” approach, we introduce a new family of clutters, the *ideal* clutters, which have a polynomial number of minimal *feasible solutions*.

The same approach was followed by Peled and Simeone in their paper on *regular set-covering problems* and threshold functions [6], where they proved that, for a fairly large class of $(0, 1)$ matrices (regular matrices), the number of feasible solutions of (SC) is bounded by a polynomial in m and n . In [1] we study the clutters the incidence matrices of which are regular (regular clutters).

In this paper we investigate (§ 2) the connections between a clutter L and its blocker, describing the structure of the members of $b(L)$ in terms of properties of the members of L . In § 3 we introduce the class of *ideal* clutters that properly contains that of regular clutters. We prove that the property of being ideal is *hereditary* with respect to *contraction* and that the polyhedron $Q(M)$ defined by the incidence matrix M of an ideal clutter has a polynomial number of vertices. Moreover we observe that the clutters defined by the bases of a matroid are ideal.

In § 4 we describe two polynomial algorithms for recognizing an ideal clutter, generating all its minimal transversals and solving the associated problem (SC). Finally, in § 5, we briefly discuss the connections between ideal clutters and regular set-covering problems, reporting the main results that appeared in [1].

2. Clutters and blockers. A clutter L is a collection $\{A_1, \dots, A_m\}$ of subsets of a ground set $E(L)$ such that for every pair $A_h, A_i \in L$, A_h is neither contained nor contains A_i (Sperner family). The elements $x_j \in E(L)$ that are not contained in any $A_i \in L$ are said to be **isolated**. A **transversal** of L is a subset of $E(L)$ having nonempty intersection with each member of L .

The **blocker** $b(L)$ of a clutter L is the collection of all the minimal transversals of L . It is easy to see that $b(L)$ is also a clutter.

Remark 2.1. Observe that the complements of transversals are precisely the sets not containing any members of L . Hence the complements of the minimal transversals are precisely the maximal sets not containing any members of L .

If L is a clutter and Z is a subset of $E(L)$, we define $L \setminus Z$ to be

$$\{A_i \in L : A_i \cap Z = \emptyset\}$$

and L/Z to be the collection of minimal members of $\{A - Z : A \in L\}$. Both $L \setminus Z$ and L/Z are clutters and the associated operations are called, respectively, **deletion** and **contraction** of Z .

A **minor** L' of L is a clutter that may be obtained from L by a sequence of deletions and contractions. As proved by Seymour in [8], we have that

$$b(L \setminus Z) = b(L)/Z, \quad b(L/Z) = b(L) \setminus Z$$

and that if $Z_1 \cap Z_2 = \emptyset$ then $(L \setminus Z_1)/Z_2 = (L/Z_2) \setminus Z_1$. Finally, we denote by $\bar{A}$ the set $E(L) - A$.

Given a clutter L , a member $A \in L$ and an element $x_h \in A$, the set $\bar{A}$ can be partitioned into the following two sets:

$$P_h(L, A) = \{x_k \notin A : \exists A' \in L : A' \subseteq A - \{x_h\} \cup \{x_k\}\},$$

$$S_h(L, A) = \bar{A} - P_h(L, A).$$

In the following lemmas we present some useful properties of the sets $P_h(L, A)$.

LEMMA 2.1. *Given a clutter L , a member $A \in L$, and an element $x_i \in A$ we have that, for each $x_h \in P_i(L, A)$ and each $A' \in L$ such that $A' \subseteq A - \{x_i\} \cup \{x_h\}$,*

$$P_h(L, A') - \{x_i\} \subseteq P_i(L, A).$$

Proof. First of all observe that $x_h \in A'$ for, otherwise, A' would be a proper subset of A . Next consider an element $x_k \in P_h(L, A') - \{x_i\}$, we have that for some $A'' \in L$, $A'' \neq A$

$$A'' \subseteq A' - \{x_h\} \cup \{x_k\}.$$

Observe that $x_k \notin A$ since, otherwise, A'' would be a proper subset of A . Moreover $A'' \subseteq A - \{x_i\} \cup \{x_k\}$, and hence $x_k \in P_i(L, A)$. $\square$

LEMMA 2.2. *Let L be a clutter, let A be a member of L , and let x_i be an element of $E(L)$. If $A - x_i \in L / \{x_i\}$, then*

$$P_h(L / \{x_i\}, A - \{x_i\}) \supseteq P_h(L, A) - \{x_i\} \quad \forall x_h \in A, \quad x_h \neq x_i.$$

Proof. Suppose there exists an element $x_j \in P_h(L, A) - \{x_i\}$ not contained in $P_h(L / \{x_i\}, A - \{x_i\})$. It follows that there exists $A' \in L$ such that

$$(2.1) \quad A' \subseteq A - \{x_h\} \cup \{x_j\}$$

and, since $x_j \in S_h(L / \{x_i\}, A - \{x_i\})$, there does not exist $A'' \in L / \{x_i\}$ such that

$$(2.2) \quad A'' \subseteq (A - \{x_i\}) - \{x_h\} \cup \{x_j\}.$$

It follows, by (2.1) and (2.2), that $A' - \{x_i\} \notin L / \{x_i\}$, and hence that there exists $A''' \in L / \{x_i\}$ such that $A''' \subseteq A' - \{x_i\}$. Consequently we have that

$$A''' \subseteq (A - \{x_i\}) - \{x_h\} \cup \{x_j\},$$

contradicting (2.2). $\square$

We are now ready to state the main result of this section concerning the connections between the members of L and those of $b(L)$. It is well known [8] that, for each clutter L , each member $A \in L$, and each element $x_h \in A$, the subset $\overline{A - \{x_h\}}$ contains at least one element $B \in b(L)$. In the next theorem we provide a complete characterization of the set $\overline{A - \{x_h\}} - B$.

To this purpose we define, for a given L , a member $A \in L$, and an element $x_h \in A$, the family $\mathcal{Y}_h(L, A)$ of the maximal subsets $Y \subseteq S_h(L, A)$ with the property that there does not exist $A' \in L$ such that $A' \subseteq A - \{x_h\} \cup Y$.

THEOREM 2.1. *Given a clutter L , a member $A \in L$, a subset $Y \subseteq \bar{A}$, and an element $x_h \in A$. Then the set $B = \overline{A - \{x_h\}} - Y$ is a member of $b(L)$ if and only if $Y \in \mathcal{Y}_h(L, A)$.*

Proof. *Necessity.* Given $A \in L$, $x_h \in A$, $Y \subseteq \bar{A}$, $B = \overline{A - \{x_h\}} - Y \in b(L)$, we must show that $Y \in \mathcal{Y}_h(L, A)$. This means we must show that $K = A - \{x_h\} \cup Y$ is a maximal set not containing any members of L . But $K = \bar{B}$, and so the result follows from Remark 2.1.

Sufficiency. Since $Y \in \mathcal{Y}_h(L, A)$, $A - \{x_h\} \cup Y$ is a maximal set that does not contain any members of L . Hence, by Remark 2.1, its complement B is a minimal transversal. $\square$

The result of the above theorem allows us to derive from each pair (A, x_h) with $A \in L$ and $x_h \in A$, a number of minimal transversals, each associated to a subset $Y \in \mathcal{Y}_h(L, A)$. In the next theorem we complete the characterization of $b(L)$ by showing that every member $B \in b(L)$ can be generated in this way.

THEOREM 2.2. *Let L be a clutter and x_h an element of $E(L)$. Then every minimal transversal $B \in b(L)$, with the property that $x_h \in B$, can be expressed as*

$$B = \overline{A - \{x_h\}} - Y = \{x_h\} \cup P_h(L, A) \cup (S_h(L, A) - Y)$$

for some $A \in L$ containing x_h , and some $Y \in \mathcal{Y}_h(L, A)$.

Proof. Let A be a member of L such that $A \cap B = \{x_h\}$. Next pose $Y = S_h(L, A) - B$. We have that

$$(2.3) \quad B \subseteq \overline{A - \{x_h\}} - Y = \{x_h\} \cup P_h(L, A) \cup (S_h(L, A) - Y).$$

What we have to prove is that (2.3) holds with equality. Suppose the contrary; it follows that there exists an element $x_j \in \overline{A - \{x_h\}} - Y$ such that $x_j \notin B$. Since $x_h \in B$ and $x_j \notin S_h(L, A)$, we have that $x_j \in P_h(L, A)$, and consequently that there exists $A' \in L$ such that

$$A' \subseteq A - \{x_h\} \cup \{x_j\}.$$

It follows that $\overline{A - \{x_h\}} - \{x_j\} \notin b(L)$, but $B \subseteq \overline{A - \{x_h\}} - \{x_j\}$ since $B \cap A = \{x_h\}$, and consequently we have the contradiction that $B \notin b(L)$. $\square$

The above theorem indicates that a necessary and sufficient condition for a clutter L to have a polynomial number of minimal transversals is that, for each $A \in L$ and each $x_h \in A$, the cardinality of $\mathcal{Y}_h(A)$ be polynomial.

It is then natural to investigate, as a first step, the conditions for $\mathcal{Y}_h(L, A)$ to have a cardinality independent of m and n , and, in particular, equal to one. The following proposition answers this question.

PROPOSITION 2.1. *Let L be a clutter, let A be a member of L , and let x_h be an element of A ; then the following statements are equivalent:*

- $$(2.4) \quad \begin{aligned} & \text{(i)} \quad \mathcal{Y}_h(L, A) = \{S_h(L, A)\}, \\ & \text{(ii)} \quad P_h(L, A) \cap A' \neq \emptyset \text{ for each } A' \in L, x_h \notin A', \\ & \text{(iii)} \quad P_h(L, A) \text{ is a transversal of } L \setminus \{x_h\}, \\ & \text{(iv)} \quad \overline{A - \{x_h\}} - S_h(L, A) = \{x_h\} \cup P_h(L, A) \in b(L). \quad \square \end{aligned}$$

As a consequence, if, for a given clutter L , one of the above properties holds for each $A \in L$ and each $x_h \in A$, the number of minimal transversals of L is bounded by a polynomial in n and m .

Remark 2.2. As an example of clutter with the latter property, consider the case in which L is the collection of the bases of a matroid $M = (\mathcal{I}, E(L))$ (matroidal clutter). In this case it is easy to prove that, given a base $B \in L$ and an element $x_h \in B$, an element $x_i \notin B$ belongs to $P_h(L, B)$ if and only if the unique circuit $C(x_i, B) \subseteq B \cup \{x_i\}$ contains x_h .

By an elementary result of matroid theory we have that $B - \{x_h\} \cup S_h(L, B)$ does not contain a base $B' \in L$, and hence that $\mathcal{Y}_h(L, B) = \{S_h(L, B)\}$. It follows that Theorem 2.1 generalizes to nonmatroidal clutters the well-known result that, for each base B of a matroid M and each element $x_h \in B$, the set

$$K = \{x_h\} \cup \{x_i \notin B : x_h \in C(x_i, B)\}$$

is a cocircuit of M .

In the following section we characterize the clutters having the more general property that at least one element of $E(L)$ satisfies (2.4) in each minor $L_i = L / \{x_{i+1}, \dots, x_n\}$. In particular, we show that also for the clutters L of this larger class the number of

minimal transversals is polynomial, and hence that the associated set-covering problem is polynomially solvable.

3. Ideal clutters.

DEFINITION 3.1. Given a clutter L , an element $x_i \in E(L)$ is said to be **last** if for each member $A \in L$, with $x_i \in A$

$$P_i(L, A) \cap A' \neq \emptyset \quad \text{for each } A' \in L, \quad x_i \notin A'.$$

Notice that, by Proposition 2.1, an element $x_i \in E(L)$ is last in L if and only if $|\mathcal{Y}_i(L, A)| = 1$ for each $A \in L$ containing x_i . An isolated element $x_j \in E(L)$ is always last.

DEFINITION 3.2. Given a clutter L , let $\sigma = (x_1, x_2, \dots, x_n)$ be an ordering of the elements of $E(L)$. We say that σ is an **ideal ordering** if x_i is a last element in the minor $L_i = L / \{x_{i+1}, \dots, x_n\}$. The minor L_i is called the i th **principal minor** of L .

And finally, we have the following definition

DEFINITION 3.3. A clutter L is **ideal** if it admits an ideal ordering of the elements of $E(L)$.

Now we present the main properties of ideal clutters.

The first natural question to ask is: Is *every* last element of an ideal clutter L the n th element of some ideal ordering? We shall answer this question by stating the more general result that the property of being ideal is *hereditary* with respect to contraction.

THEOREM 3.1. *If a clutter L is ideal then the minor L/Z is ideal for each $Z \subseteq E(L)$.*

Proof. The clutter L is ideal and the elements of $E(L)$ are assumed to be indexed according to the ordering $\sigma = (x_1, \dots, x_n)$. Since every minor L/Z of L can be obtained by contracting, in any order, the elements in Z , we can prove the theorem by showing that, given an ideal clutter L and an element $x_i \in E(L)$, the minor $L/\{x_i\}$ is ideal and $\sigma' = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n)$ is its ideal ordering.

Then we must prove that x_j is last in $L_j/\{x_i\}$ for $j > i$, and that x_j is last in L_j for $j < i$. The second statement holds by assumption, and for the first statement it is enough to assume $j = n$. Now suppose x_n is not last in $L/\{x_i\}$; then we have that there exist two members A'_1 and A'_2 in $L/\{x_i\}$, with $x_n \in A'_1 - A'_2$ such that

$$(3.1) \quad P_n(L/\{x_i\}, A'_1) \cap A'_2 = \emptyset.$$

Let A_1 and A_2 be the members of L such that $A'_1 = A_1 - \{x_i\}$ and $A'_2 = A_2 - \{x_i\}$. Since x_n is last in L , we have that $P_n(L, A_1) \cap A_2 \neq \emptyset$.

Moreover, by Lemma 2.2, $P_n(L, A_1) - \{x_i\} \subseteq P_n(L/\{x_i\}, A'_1)$, and hence, by (3.1), $(P_n(L, A_1) - \{x_i\}) \cap A'_2 = \emptyset$. It follows that $x_i \in P_n(L, A_1)$. As a consequence we have that there exists a member $A_3 \in L$ such that

$$A_3 \subseteq A_1 - \{x_n\} \cup \{x_i\}.$$

Now, denoted by A'_3 the member of $L/\{x_i\}$ contained in $A_3 - \{x_i\}$, we have that $A'_3 \subseteq A'_1$, which is a contradiction. $\square$

Remark 3.1. The previous theorem implies that if L is ideal and $x_h \in E(L)$ is last in L , the minor $L/\{x_h\}$ is also ideal, and hence x_h is the n th element of an ideal ordering.

Unfortunately, it is not true that *every* minor of an ideal clutter is ideal. In fact if we *delete* the element x_3 in the ideal clutter $L = \{\{x_1, x_2\}, \{x_2, x_3, x_4\}, \{x_4, x_5\}\}$ we produce the *nonideal* minor $L \setminus \{x_3\} = \{\{x_1, x_2\}, \{x_4, x_5\}\}$.

The next theorem provides a complete characterization of the minimal transversals containing a *last* element of an ideal clutter L .

THEOREM 3.2. *Let L be a clutter and $x_h \in E(L)$ be a last element of L . Let $x_h \in B \subseteq E(L)$. Then B is a minimal transversal if and only if it can be expressed as*

$$B = \{x_h\} \cup P_h(L, A)$$

for some $A \in L$ containing x_h .

Proof. The proof follows easily from Theorem 2.2 and Proposition 2.1. $\square$

Remark 3.2. From the arguments of Remark 2.2, it follows that, if a clutter L is the collection of the bases of a matroid M (matroidal clutter), every element $x_h \in E(L)$ is last. Consequently, every matroidal clutter is ideal and the result of Theorem 3.2 yields the following characterization of the cocircuits of a matroid M : a subset $C^* \subseteq E(L)$, containing $x_h \in E(L)$, is a cocircuit of the matroid M if and only if there exists a base $B \in L$, containing x_h , such that $C^* - \{x_h\} = \{x_i \notin B : C(x_i, B) \ni x_h\}$, where $C(x_i, B)$ is the fundamental circuit defined by x_i in B .

Evidently, all the minimal transversals of a clutter L can be produced in n iterations, provided the k th iteration generates all the members of $b(L)$ containing x_k and not containing x_l for $l = k + 1, \dots, n$. Moreover, since the minimal transversals of $b(L)$ not containing x_l for $l = k + 1, \dots, n$ are the members of the minor of $b(L)$ obtained by deleting $\{x_{k+1}, \dots, x_n\}$, we have that at the k th iteration we should generate all the members of $b(L) \setminus \{x_{k+1}, \dots, x_n\} = b(L / \{x_{k+1}, \dots, x_n\}) = b(L_k)$ containing x_k . This is an easy task since L is an ideal clutter, each principal minor L_k is ideal, and x_k is last in L_k . It follows that we can apply to L_k and x_k the result of Theorem 3.2, producing all the members of $b(L_k)$ containing x_k .

In the next section we present two polynomial algorithms for recognizing ideal clutters, generating their blockers, and finding the minimum transversal, that are essentially based on the above observations.

4. Recognition and minimum weight transversal of ideal clutters. We now propose two simple algorithms, both having polynomial complexity, based on the results of the previous sections. The first algorithm (ORDER) finds, if it exists, an ideal ordering for a given clutter L or concludes that the clutter is not ideal. The second (MWT), given an ideal clutter L and a weight c_h associated to each element $x_h \in E(L)$, generates its blocker $b(L)$ and produces a transversal of L having minimum weight.

4.1. Algorithm ORDER. For a given clutter L the algorithm ORDER finds, if the clutter L is ideal, an ideal ordering σ of the elements of $E(L)$; otherwise it concludes that L is not ideal. Evidently, since every last element of a clutter L can be the n th element of an ideal ordering (Theorem 3.1), we have only to check if every principal minor of L has a last element.

procedure ORDER (input: L ; output: σ)

begin

call PPROC (input: L ; output: $\mathcal{P}(L)$);

$L_n := L$;

$k := n$

repeat

call LAST (input: L_k , $\mathcal{P}(L_k)$; output: $LASTEL$);

if $LASTEL \neq \emptyset$ *comment: $LASTEL = \emptyset \rightarrow L_k$ not ideal*

then begin

$\sigma(k) := LASTEL$;

$L_{k-1} := L_k / \{LASTEL\}$; *comment: Contraction of $LASTEL$*

$k := k - 1$;

```

    call PPROC (input:  $L_k$ ; output:  $\mathcal{P}(L_k)$ );
  end
  else write ("L is not ideal !")
until  $LASTEL = \emptyset$  or  $k = 0$ 
end.

```

The algorithm ORDER is organized in two main phases. The first (PPROC) produces the family $\mathcal{P}(L) = \{P_h(L, A) : A \in L, x_h \in A\}$. The second consists of n iterations. At the k th iteration a last element $LASTEL$ of the clutter L_k is found (LAST) and assigned to $\sigma(k)$, and, successively, the new clutter $L_{k-1} = L_k / \{LASTEL\}$ and the new family $\mathcal{P}(L_{k-1})$ are produced.

Now we describe the two procedures PPROC and LAST, and compute the relative complexities.

The procedure PPROC receives as input a clutter L and produces the family $\mathcal{P}(L)$. It considers one member $A_i \in L$ at a time and compares it with the members $A_j \in L$, $j \neq i$. For each $x_h \in A_i - A_j$, the elements $x_k \notin A_i$, such that $A_j \subseteq A_i - \{x_h\} \cup \{x_k\}$, are inserted into the set $P_h(L, A_i)$.

More precisely, the comparison is carried out as follows. For each pair $A_i, A_j \in L$, we define the two sets $C_{ij} = A_i - A_j$ and $C_{ji} = A_j - A_i$. If $C_{ji} = \{x_k\}$ we have that $A_j \subseteq A_i - \{x_h\} \cup \{x_k\}$ for each $x_h \in C_{ij}$. It follows that $x_k \in P_h(L, A_i)$ for each $x_h \in C_{ij}$. If, on the other hand, $|C_{ji}| > 1$, then A_j cannot be contained in any subset of the form $A_i - \{x_h\} \cup \{x_k\}$. In this case the comparison of A_i and A_j gives no contribution to the evaluation of $P_h(L, A_i)$ for each $x_h \in A_i$.

```

procedure PPROC (input:  $L$ ;  output:  $\mathcal{P}(L)$ )
begin
  for each  $A_i \in L$ 
    do for each  $A_j \in L$ ,  $j \neq i$ 
      do begin
         $C_{ij} = A_i - A_j$ ;
         $C_{ji} = A_j - A_i$ ;
        if  $|C_{ji}| = 1$ 
          then for each  $x_h \in C_{ij}$  do  $P_h(L, A_i) = P_h(L, A_i) \cup C_{ji}$ 
        end
      end
    end
  end

```

The number of elementary steps of the procedure PPROC is, evidently, $\mathcal{O}(nm^2)$. The procedure LAST simply applies Definition 3.1 to check if the clutter L has a last element.

```

procedure LAST (input:  $L$ ,  $\mathcal{P}(L)$ ;  output:  $LASTEL$ );
begin
   $LASTEL = \emptyset$ ;
  for each  $x_h \in E(L)$ 
    do for each  $A_i \ni x_h$ 
      do if  $(P_h(L, A_i) \cap A_j \neq \emptyset$  for all  $A_j \ni x_h)$ 
        then  $LASTEL := x_h$ ;
      end
    end
  end

```

The test performed for each $x_h \in E(L)$ and each $A_i \ni x_h$ requires $\mathcal{O}(mn)$ steps, and hence the procedure LAST can be carried out in time $\mathcal{O}(m^2n^2)$. We can now derive the complexity of the algorithm for recognizing ideal clutters. Since the contraction of an element $x_h \in E(L)$ requires no more than $\mathcal{O}(m^2n)$ elementary steps, we have that the number of steps of the procedure ORDER is $\mathcal{O}(m^2n^3)$.

4.2. Algorithm MWT. The algorithm enumerates all the minimal transversals B of an ideal clutter L , and selects a transversal $B^* \in b(L)$ with the property that

$$c(B^*) \leq c(B) \quad \forall B \in b(L)$$

where $c(B) = \sum_{x_h \in B} c_h$. We assume that an ideal ordering $\sigma = (x_1, \dots, x_n)$ of the elements of $E(L)$ is given.

The basic idea of the algorithm MWT is that all the minimal transversals of L can be enumerated in n iterations, producing, at the k th iteration, the family $\mathcal{B}_k$ of all the members $B \in b(L)$ containing x_k and not containing x_j for $j = k+1, \dots, n$. The family $\mathcal{B}_k$ can be produced by observing that its members are the members of the minor $b(L) \setminus \{x_{k+1}, \dots, x_n\}$ containing x_k . Moreover $b(L) \setminus \{x_{k+1}, \dots, x_n\}$ is the blocker of the k th principal minor L_k and the element x_k is *last* in L_k . It follows, by Theorem 3.2, that each member $B \in \mathcal{B}_k$ can be expressed as $B = \{x_k\} \cup P_k(L_k, A)$, for some $A \in L_k$ containing x_k .

Observe that the family $\mathcal{B}_k$ contains at most $d_k = |\{A \in L_k : x_k \in A\}| \leq m$ members. It follows that we can establish the following upper bound on the cardinality of $b(L)$:

$$|b(L)| \leq \sum_{k=1}^n d_k \leq mn.$$

By the above remarks, the k th iteration of the algorithm MWT consists of two main logical phases. In the first the family $\mathcal{B}_k$ is generated and, possibly, B^* is updated. In the second the $(k-1)$ th principal minor L_{k-1} is produced by *contracting* x_k in L_k . The latter task can also be carried out quite easily since the minor L_{k-1} can be defined, in terms of L_k , in the following way:

$$(4.1) \quad L_{k-1} = \{(A_i - \{x_k\}) : A_i \in L_k \text{ and } x_k \notin P_h(L_k, A_i) \forall x_h \in A_i\}.$$

In fact, if $A_i - \{x_k\} \notin L_{k-1}$ then there exists $A_j \in L_k$ $A_j \neq A_i$, such that $A_j - \{x_k\} \subseteq A_i - \{x_k\}$. Consequently, since $A_j \subsetneq A_i$, we have that $x_k \in A_j - A_i$, and since $A_i \subsetneq A_j$, there exists $x_h \in A_i - A_j$. Therefore, $A_j \subseteq A_i - \{x_h\} \cup \{x_k\}$; hence $x_k \in P_h(L_k, A_i)$.

Conversely, if $x_k \in P_h(L_k, A_i)$ for some $x_h \in A_i$, then $A_i - \{x_k\} = A_i \notin L_{k-1}$. Thus an $A_i \in L_k$ satisfies $A_i - \{x_k\} \notin L_{k-1}$ if and only if $x_k \in P_h(L_k, A_i)$ for some $x_h \in A_i$. Equivalently, $A_i \in L_k$ satisfies $A_i - \{x_k\} \in L_{k-1}$ if and only if $x_k \notin P_h(L_k, A_i)$ for all $x_h \in A_i$.

Both the family $\mathcal{B}_k$ and the principal minor L_{k-1} can be produced in $\mathcal{O}(m^2)$ elementary steps if the subsets $C_{ij} = A_i - A_j$ are available for each pair $A_i, A_j \in L_k$. More precisely, for each $A_i \in L_k$, we have that

$$P_k(L_k, A_i) = \{x_h \notin A_i : \{x_h\} = C_{ji} \text{ for some } A_j \ni x_k\},$$

and hence, if $x_k \in A_i$ we can produce the minimal transversal $x_k \cup P_k(L_k, A_i)$, update B^* and add $A_i - \{x_k\}$ to L_{k-1} . Conversely, if $x_k \notin A_i$, we have only to decide if $A_i - \{x_k\}$ belongs to L_{k-1} and, by (4.1), this is the case if and only if there does not exist a member $A_j \in L_k$ such that $C_{ji} = \{x_k\}$.

Finally observe that the sets C_{ij} can be computed, at the first iteration, in $\mathcal{O}(m^2n)$ elementary steps and can be updated in $\mathcal{O}(m^2)$ steps, at the k th iteration, by simply removing the last element x_k from the sets C_{ij} such that $A_i, A_j \in L_k$.

It follows that the total complexity of the algorithm MWT is $\mathcal{O}(m^2n)$.

The algorithm receives as input the clutter L , the weight vector c and the ideal ordering $\sigma = (x_1, \dots, x_n)$ and produces the blocker $b(L)$, the transversal with minimum weight B^* and its weight MIN .

procedure MWT (**input**: L, c, σ ; **output**: $b(L), B^*, MIN$);

begin

for each $A_i \in L$

do for each $A_j \in L, j \neq i$

do $C_{ji} := A_j - A_i$;

$MIN := \infty$

$b(L) := \emptyset$

for $k := n$ **downto** 1

do begin

for each $A_i \in L_k$

do if $x_k \in A_i$

then begin

$B := \{x_k\}$;

$c(B) := c_k$;

for each $A_j \in L_k, j \neq i$

do if $x_k \notin A_j$ **and** $C_{ji} = \{x_i\}$

then begin

$B := B \cup \{x_i\}$;

$c(B) := c(B) + c_i$;

end;

$b(L) := b(L) \cup \{B\}$;

if $c(B) < MIN$

then begin

$MIN := c(B)$;

$B^* := B$

end;

$L_{k-1} := L_{k-1} \cup \{A_i - \{x_k\}\}$

end

else *comment: $x_k \notin A_i$*

if $C_{ji} \neq \{x_k\}$ **for all** $A_j \in L_k, j \neq i$

then $L_{k-1} := L_{k-1} \cup \{A_i\}$;

for each $C_{ji} : A_i, A_j \in L_k$

do $C_{ji} := C_{ji} - \{x_k\}$

end

end.

Example 4.1. In this example we apply the algorithm MWT to the ideal clutter

$$L = \{ \{x_1, x_2\}, \{x_2, x_4\}, \{x_1, x_4, x_5\}, \{x_1, x_3\} \}.$$

More precisely, we show how the minimal transversals of L can be enumerated, skipping the obvious details concerning the evaluation of the cost function and the choice of the transversal having minimum weight.

To make the procedure easier to follow we give a pictorial representation of the successive iterations by listing on the left the incidence matrix M_{L_k} of the principal minor L_k and on the right the family $\bigcup_{i=1}^k \mathcal{B}_k$. We also list the values of $P_k(L_k, A_i)$ for each $A_i \in L_k$ containing the last element x_k of L_k .

We start from the matrix $M_L = M_{L_5}$, ordered according to the ideal ordering $\sigma = (x_1, x_2, x_3, x_4, x_5)$:

$$M_{L_5} = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{matrix} & \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \end{pmatrix} \end{matrix}, \quad \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} B_1(0) \end{matrix} & \begin{pmatrix} 1 & 1 & 0 & 1 \end{pmatrix} \end{matrix}, \quad P_5(L_5, A_3) = \{2, 3\}, \quad B_1(0) = \{1\}.$$

In the first iteration we produce all the members of $b(L)$ containing x_5 (B_1) and, successively, we contract the element x_5 producing the principal minor L_4 :

$$M_{L_4} = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{matrix} & \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \end{matrix}, \quad \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} B_1 \\ B_2 \\ B_3 \end{matrix} & \begin{pmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix} \end{matrix}, \quad P_4(L_4, A_2) = \{1\}, \quad P_4(L_4, A_3) = \{2, 3\}.$$

In the second iteration all the members of $b(L)$ containing x_4 and not containing x_5 (B_2, B_3) are produced and the minor L_3 is generated. Observe that the sets $A_1 - \{x_4\}$ and $A_4 - \{x_4\}$ are not included in L_3 , according to (4.1), since $x_4 \in P_3(L_4, A_4)$ and $x_4 \in P_1(L_4, A_1)$:

$$M_{L_3} = \begin{matrix} & \begin{matrix} 1 & 2 & 3 \end{matrix} \\ \begin{matrix} A_2 \\ A_3 \end{matrix} & \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \end{matrix}, \quad \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} B_1 \\ B_2 \\ B_3 \end{matrix} & \begin{pmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix} \end{matrix}.$$

Observe that, since x_3 is an *isolated* element in L_3 , it is last in L_3 but no member of $b(L)$ containing x_3 and not containing x_4 and x_5 exists, and hence no minimal transversal is added to $b(L)$ in the third iteration. Moreover the contraction of the isolated element x_3 is, simply, the removal of the third column of M_{L_3} :

$$M_{L_2} = \begin{matrix} & \begin{matrix} 1 & 2 \end{matrix} \\ \begin{matrix} A_2 \\ A_3 \end{matrix} & \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \end{matrix}, \quad P_2(L_2, A_2) = \{1\}, \quad M_{b(L)} = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} B_1 \\ B_2 \\ B_3 \\ B_4 \end{matrix} & \begin{pmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix} \end{matrix}.$$

At the 4th iteration all the members containing x_2 and not containing x_3, x_4, x_5 are generated (B_4) and the last element x_2 is contracted, producing the *empty* clutter. The matrix $M_{b(L)}$ is the incidence matrix of the blocker $b(L)$.

5. Regular clutters. Our interest in ideal clutters was originally motivated by the results on a special class of polynomially solvable set-covering problems presented in a paper by Peled and Simeone dealing with threshold functions, the so-called *regular set-covering problems* [6].

A problem (SC) is said to be *regular* if there exists an ordering of the variables $\sigma = (x_1, \dots, x_n)$ such that, for every feasible solution x with $x_i = 1$, $x_j = 0$, $j < i$, the vector $x + e_j - e_i$ is also a feasible solution, where e_i is the i th unit vector. A matrix M associated to a regular set-covering problem is said to be regular and its columns are assumed to be indexed according to the ordering σ .

In [6], Peled and Simeone proposed an algorithm (*Hop-Skip-Jump*) which, in $\mathcal{O}(mn^3)$ elementary steps produces all the feasible solutions of a regular set-covering problem (SC) and observed that a regular matrix M can be recognized in $\mathcal{O}(m^2n^2)$. In a recent survey [5], Hammer and Simeone, extending the results in [6], proved that the problem (SC) defined on a regular matrix M can be solved in time $\mathcal{O}(mn \lg_2 m)$. They also report that, following a different approach, Crama [2] devised an $\mathcal{O}(mn^2)$ algorithm for generating all the feasible solutions of a regular set-covering problem. More recently [7] Peled and Simeone have improved the bound of Hammer and Simeone [5], showing that it is possible to find an optimal solution for a regular set-covering problem in $\mathcal{O}(mn)$ steps.

In [1] we show that every clutter L the incidence matrix M_L of which is regular (*regular clutter*) is *ideal*.

As a consequence, all properties and algorithms stated in the previous sections for ideal clutters can be specialized to regular clutters.

In particular, based on the general result stated by Theorem 3.2 and on some special properties of regular clutters, we improve the polynomial bounds reported above by providing an algorithm which, in $\mathcal{O}(mn)$ elementary steps, *both* generates the blocker $b(L)$ and finds the minimum weight transversal of a regular clutter L . In addition, we prove that all the *minors* of a regular clutter are regular and that a clutter is regular if and only if its blocker is regular.

Acknowledgments. The authors are deeply indebted to the referees who gave detailed and helpful advice and made numerous suggestions for improvements.

REFERENCES

- [1] P. BERTOLAZZI AND A. SASSANO, *Regular clutters*, Theoret. Comput. Sci., 54 (1987), pp. 1–11.
- [2] Y. CRAMA, *Dualization of a regular Boolean function*, Technical Report, Rutgers University, New Brunswick, NJ, May 1986.
- [3] P. L. HAMMER, E. L. JOHNSON, AND U. N. PELED, *Regular 0-1 programs*, Cahiers Centre Etudes Rech. Opér., 16 (1974), pp. 267–276.
- [4] P. L. HAMMER, U. N. PELED, AND M. A. POLLATSCHEK, *An algorithm to dualize a regular switching function*, IEEE Trans. Comput., 28 (1979), pp. 238–243.
- [5] P. L. HAMMER AND B. SIMEONE, *Order relations of variables in 0-1 programming*, in Proc. School on Combinatorial Optimization, C. Ribeiro, G. Laporte, and S. Martello, eds., Rio de Janeiro, 1985.
- [6] U. N. PELED AND B. SIMEONE, *Polynomial-time algorithms for regular set-covering and threshold synthesis*, Discrete Appl. Math., 12 (1985), pp. 57–69.
- [7] ———, private communication, October 1986.
- [8] P. D. SEYMOUR, *The matroids with the max-flow min-cut property*, J. Combin. Theory Ser. B, 23 (1977), pp. 189–222.

TOTAL DOMINATION AND IRREDUNDANCE IN WEIGHTED INTERVAL GRAPHS*

ALAN A. BERTOSSI† AND ALESSANDRO GORI†

Abstract. In an undirected graph, a subset X of the nodes is a total dominating set if each node in the graph is a neighbor of some node in X . In contrast, X is an irredundant set if the closed neighborhood of each node in X is not contained in the union of closed neighborhoods of the other nodes in X . This paper gives $O(n \log n)$ and $O(n^4)$ time algorithms for finding, respectively, a minimum weighted total dominating set and a minimum weighted maximal irredundant set in a weighted interval graph, i.e., one that represents n intersecting intervals on the real line, each having a (possibly negative) real weight.

Key words. dominating set, total dominating set, irredundant set, interval graph, min-tree, shortest path

AMS(MOS) subject classifications. 05C70, 49B99, 68R10, 68Q25

1. Introduction. Let $G(N, E)$ be an undirected graph. The *open neighborhood* of a node x is $N(x) = \{y \in N : [x, y] \in E\}$, while its *closed neighborhood* is $N[x] = N(x) \cup \{x\}$. In general, let $N(X)$ and $N[X]$ denote, respectively, $\bigcup_{x \in X} N(x)$ and $N(X) \cup X$.

A subset X of the nodes *dominates* another subset Y if $Y \subseteq N(X)$. X is a *dominating set* if $N[X] = N$, while it is a *total dominating set* if in addition $X \subseteq N(X)$ [COC80]. In words, X is a dominating set if each node of $N - X$ is adjacent to at least one node in X , while it is also total if in addition each node in X is adjacent to some other node in X . As an example, Fig. 1(a) shows the “Capricorn” graph. It is easy to realize that $\{1, 4, 6\}$ is a dominating set whereas $\{2, 4\}$ is not; moreover, $\{1, 2, 5, 6\}$ is a total dominating set.

A node $x \in X$ is *redundant* in X if $N[x] \subseteq N[X - \{x\}]$. X is an *irredundant set* if it contains no redundant node. An irredundant set is *maximal* if it is not properly contained within another irredundant set. For instance, the set $\{2, 4\}$ is an irredundant set for the Capricorn graph, whereas $\{2, 4, 5\}$ is not, since node four is redundant; furthermore, $\{2, 4\}$ is also maximal, since adding any other node to it would induce redundancies.

The problems of determining dominating, total dominating, or irredundant sets have obvious applications in the location of facilities in a network [ROB78]. In particular, the notion of redundancy is relevant in the context of communication networks, since any redundant node in a set can be removed from the set without affecting the totality of nodes that may receive communication from some node in the set [BOL79], [LAS85].

The problem of finding dominating sets along with several closely related variants has been remarkably investigated (e.g., see [HUJ84], [JOH84] for extensive references). In particular, finding a minimum cardinality total dominating set is NP-complete, even for special classes of graphs, such as *bipartite* or *chordal graphs* [BER84], [CHA84], [LAS84]. Moreover, finding a minimum cardinality maximal irredundant set is also NP-complete for chordal graphs [LAS83], but can be solved in linear time for *trees* [BER85] and *partial k-trees* [WIM87]. In this paper, we consider the weighted versions of these last two problems on a particular subclass of chordal graphs, known as *interval graphs*. Such graphs are one of the most useful discrete mathematical structures for modeling problems arising in the real world. In fact, they have found applications in

* Received by the editors December 1, 1986; accepted for publication (in revised form) February 9, 1988.

† Dipartimento di Informatica, Università di Pisa, Corso Italia 40, 56100 Pisa, Italy.

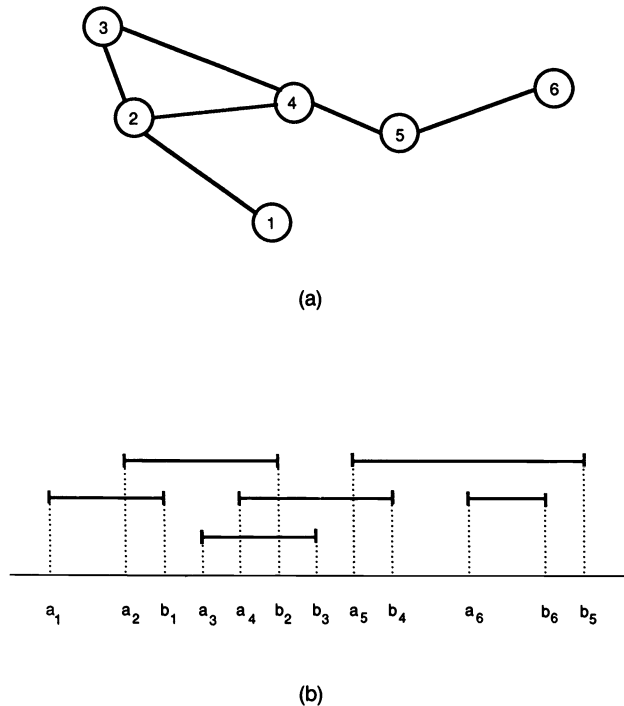


FIG. 1. (a) The "Capricorn" graph; (b) an interval family for it.

archaeology, biology, psychology, traffic control, job scheduling, and storage information retrieval (e.g., see [GOL80], [ROB78], [GOL85]).

More formally, an *interval family* is a set of intervals on the real line. A graph is an *interval graph* if there is a one-to-one correspondence between the nodes of the graph and the intervals of an interval family such that two nodes of the graph are joined by an edge if and only if their corresponding intervals overlap. An interval graph is *weighted* if a real (possibly negative) number w_i (the *weight*) is associated to each node i . For instance, Fig. 1(b) shows an interval family for the Capricorn graph, which thus results to be an interval graph.

Investigating the algorithmic complexity of total domination in interval graphs was mentioned in [LAS84] as a relevant open question. Successively, an $O(n + e)$ time algorithm for finding a minimum cardinality total dominating set in an n -node, e -edge interval graph has been devised (see [KEI86] and, for a more comprehensive exposition, [RAM88]). Independently, an $O(n^2)$ time algorithm has been proposed for the same problem that can be easily modified so as to work in the weighted case too [BER86]. In this paper, in § 3, we improve upon both these results, by giving an $O(n \log n)$ time algorithm for finding a minimum weighted total dominating set in an interval graph. We also present in § 4 an $O(n^4)$ time algorithm for determining a minimum weighted maximal irredundant set in an interval graph.

2. Background. Let $I = \{1, \dots, n\}$ be an interval family such that the i th interval is equal to $[a_i, b_i]$, $i = 1, \dots, n$. Without loss of generality, we assume that each interval contains both its endpoints and that no two intervals share a common endpoint. For the sake of simplicity, the interval graph G corresponding to the interval family I will be denoted as $G(I)$ and we shall deal directly only with the intervals instead of the nodes.

We assume that the interval family I is augmented with two extreme intervals, say zero and $n + 1$, in such a way that $b_0 < a_1$, $b_n < a_{n+1}$, and $w_0 = w_{n+1} = 0$. We also assume that the intervals are indexed by increasing a_i 's, namely, $a_0 < a_1 < \dots < a_n < a_{n+1}$. We generically denote with $e_1, \dots, e_{2n+4}$ the resulting $2n + 4$ endpoints, sorted by increasing order.

To better understand the algorithms presented in this paper, we first need to review some known results. As shown in [MAN84], the problem of finding a dominating set having minimum overall weight in an interval graph $G(I)$ can be solved by reducing it to a shortest path problem on an appropriate directed graph H .

Let us temporarily assume that there is no negative weight and consider the augmented interval family $I' = I \cup \{0, n + 1\}$. The nodes of H correspond to the intervals in I' . There is an arc (i, j) in H with length w_i if and only if $j \in P_i \cup Q_i$, where

$$P_i = \{k : a_i < a_k < b_i < b_k\}, \quad \text{and}$$

$$Q_i = \{k : a_k > b_i \text{ and there is no } h \text{ with } b_i < a_h < b_h < a_k\}.$$

We may readily verify that any shortest path from node zero to node $n + 1$ in H corresponds to a minimum dominating set for $G(I')$. The minimum dominating set for $G(I)$ is obtained from such a path by deleting nodes zero and $n + 1$.

Since H is acyclic, a shortest path can be found in $O(n^2)$ time [LAW76]. However, an $O(n \log n)$ time algorithm exists that uses a data structure known as *min-tree*. It consists of a complete binary tree with $n + 2$ leaves (corresponding to intervals in I') that is almost exactly balanced in such a way that the leftmost set of leaves is at one level and the rightmost set of leaves at one level less. A value is assigned to each leaf of the tree, while each interior node maintains the minimum of the values of its sons. An example is shown in Fig. 2.

Let us define a *block* to be a set of intervals with consecutive indices, e.g., $\langle i, i + 1, \dots, i + d \rangle$. Given a block of leaves, it is possible to find in $O(\log n)$ time the minimum value in the block along with the index of the leaf in which this minimum is located [AHO74], [MAN84]. A new value may be inserted in a leaf, and updating the min-tree then also takes $O(\log n)$ time. Moreover, constructing such a min-tree takes $O(n)$ time.

Manacher and Smith's algorithm for finding a minimum dominating set sequentially scans the interval endpoints $e_1, \dots, e_{2n+2}$ backward, from right to left (lines 5–14). As soon as a right endpoint b_i is encountered (line 6), the set Q_i is determined (line 7). Since Q_i is clearly a block, it is implemented by a pair $\langle Q_i[\text{lower}], Q_i[\text{upper}] \rangle$. The algorithm uses a min-tree T_{DIST} in which the value $\text{DIST}[i]$ associated to a leaf i is the distance in a shortest path of H between i and $n + 1$. As soon as a left endpoint a_i is

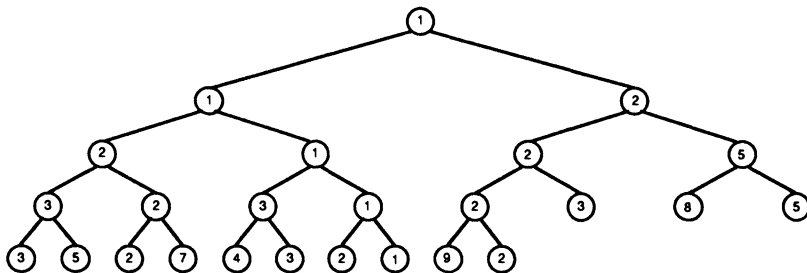


FIG. 2. A min-tree with 13 leaves.

encountered (line 8), the leaf k having smallest value in the block $\langle i + 1, Q_i[\text{upper}] \rangle$ is determined and $\text{DIST}[k]$ is added to w_i in order to get $\text{DIST}[i]$ (lines 9–11).

```

1. procedure MINIMUM-DOMINATING-SET;
2.   Construct a min-tree  $T_{\text{DIST}}$  with leaf values  $\text{DIST}[0], \dots, \text{DIST}[n+1]$ ;
3.    $\text{DIST}[n+1] := 0$  and UPDATE ( $T_{\text{DIST}}$ );
4.    $\text{NEXT}[n+1] := \text{"end-of-list"}$ ;  $\text{AVAIL}[\text{lower}] := \text{AVAIL}[\text{upper}] := n+1$ ;
5.   for  $s$  from  $2n+2$  to 1 by  $-1$  do
6.     if  $e_s$  is a right endpoint  $b_i$  then
7.        $Q_i[\text{lower}] := \text{AVAIL}[\text{lower}]$ ;  $Q_i[\text{upper}] := \text{AVAIL}[\text{upper}]$ ;
8.     else //  $e_s$  is a left endpoint  $a_i$  //
9.        $k := \text{MINEL}(T_{\text{DIST}}, i+1, Q_i[\text{upper}])$ ;
10.       $\text{NEXT}[i] := k$ ;
11.       $\text{DIST}[i] := w_i + \text{DIST}[k]$  and UPDATE ( $T_{\text{DIST}}$ );
12.       $\text{AVAIL}[\text{lower}] := i$ ;  $\text{AVAIL}[\text{upper}] := \min \{ \text{AVAIL}[\text{upper}],$ 
         $Q_i[\text{lower}] - 1 \}$ ;
13.    endif
14.  endfor.
```

The output of the algorithm is a minimum dominating set the weight of which is $\text{DIST}[0]$ and the elements of which are those in the NEXT list beginning with $\text{NEXT}[0]$. The subroutine *MINEL* (T, i, j) selects the leaf of the min-tree T having minimum value in the block $\langle i, j \rangle$. In case of ties, it chooses the rightmost leaf in the block. Let

$$C_i = \{ h : a_i < a_h < b_h < b_i \}.$$

The correctness of the algorithm is based on the fact that for the block

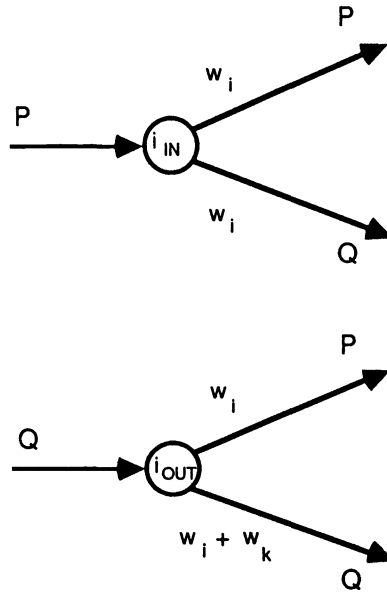
$$\langle i + 1, Q_i[\text{upper}] \rangle = C_i \cup P_i \cup Q_i,$$

MINEL does not select any interval k which either is in C_i or is “sandwiched” (i.e., with $\text{NEXT}[k] \in C_i \cup P_i$). Indeed, in both cases, either $\text{DIST}[k]$ is not the minimum value in the block (since $w_k > 0$) or it is (because $w_k = 0$), but k is not the rightmost leaf [MAN84, Lemma 5b]. Since *MINEL* and *UPDATE* both take $O(\log n)$ time, the overall $O(n \log n)$ time bound follows.

In the case there are negative weights, each $w_i < 0$ is temporarily set to zero. Then the above procedure is applied and a minimum weighted dominating set for $G(I)$ is obtained by deleting intervals zero and $n + 1$ and by adding all the intervals having negative weights [MAN84].

3. Total domination. As we may easily prove by extending the results given in [BER86], the problem of finding a minimum weighted total dominating set can also be reduced to a shortest path problem on an acyclic directed graph H' and thus solved in $O(n^2)$ time. Temporarily assuming again there are only nonnegative weights, the nodes of H' are obtained from those of H by splitting each node i into two nodes, say i_{IN} and i_{OUT} . As for the arcs, they are defined as follows:

- (i) There is an arc $(i_{\text{IN}}, j_{\text{IN}})$ with length w_i if and only if $j \in P_i$;
- (ii) There is an arc $(i_{\text{IN}}, j_{\text{OUT}})$ with length w_i if and only if $j \in Q_i$;
- (iii) There is an arc $(i_{\text{OUT}}, j_{\text{IN}})$ with length w_i if and only if $j \in P_i$;
- (iv) There is an arc $(i_{\text{OUT}}, j_{\text{OUT}})$ with length $w_i + w_k$ if and only if $j \in Q_i$, $k \in C_i$ and $w_k = \min \{ w_h : h \in C_i \}$.

FIG. 3. Nodes and arcs in H' .

As illustrated in Fig. 3, only “ P -arcs” enter into an “IN-node,” while both P - and “ Q -arcs” can leave from it. In contrast, only Q -arcs enter into an “OUT-node.” This is because a total dominating set cannot contain any *isolated* interval, that is, one that does not overlap with any other interval in the set. Since P -arcs are induced by interval overlapping, but Q -arcs are not, no two consecutive Q -arcs may appear in H' , unless an interval properly contained within another (indeed, that having minimum weight) is taken into account as in (iv).

As before, our algorithm scans the interval endpoints $e_1, \dots, e_{2n+2}$ backward. However, we use three min-trees T_{IN} , T_{OUT} , and T_{INC} . The leaves of T_{IN} (T_{OUT}) correspond to IN-nodes (OUT-nodes) of H' , and the value $IN[i]$ ($OUT[i]$) associated to a generic leaf i is the shortest path distance between i_{IN} (i_{OUT}) and $n+1_{OUT}$ (lines 24–30 and 31–42, respectively). T_{INC} , instead, is used to determine, for each i , a $k \in C_i$, if any, such that $w_k = \min \{w_h : h \in C_i\}$. The value $INC[i]$ initially associated to each leaf i is w_i (lines 6–8). Since C_i is not a block, the block $\langle i+1, Q_i[\text{lower}] - 1 \rangle = C_i \cup P_i$ is considered as soon as the right endpoint b_i is encountered (lines 13–16). The algorithm acts in such a way that leaves in P_i have value equal to infinity at this time. This is accomplished by setting $INC[i]$ to ∞ as soon as b_i is scanned, since hereafter interval i cannot be properly included in any interval yet to be encountered (line 12).

1. **procedure** *MINIMUM-TOTAL-DOMINATING-SET*;
2. Construct three min-trees T_{IN} , T_{OUT} and T_{INC} with leaf values, respectively, $IN[0], \dots, IN[n+1]$, $OUT[0], \dots, OUT[n+1]$, and $INC[0], \dots, INC[n+1]$;
3. $IN[n+1] := 0$ and *UPDATE* (T_{IN}); $OUT[n+1] := 0$ and *UPDATE* (T_{OUT});
4. $NEXT[n+1_{IN}] := NEXT[n+1_{OUT}] := \text{“end-of-list”}$;
5. $AVAIL[\text{lower}] := AVAIL[\text{upper}] := n+1$;
6. **for** i **from** 0 **to** $n+1$ **do**

```

7.      INC [i] :=  $w_i$  and UPDATE ( $T_{\text{INC}}$ ); CONT [i] := "empty";
8.  endfor
9.  for  $s$  from  $2n+2$  to 1 by  $-1$  do
10.    if  $e_s$  is a right endpoint  $b_i$  then
11.       $Q_i[\text{lower}] := \text{AVAIL} [\text{lower}]; Q_i[\text{upper}] := \text{AVAIL} [\text{upper}];$ 
12.      INC [i] :=  $\infty$  and UPDATE ( $T_{\text{INC}}$ );
13.      if  $i+1 < Q_i[\text{lower}]$  then
14.         $k := \text{MINEL} (T_{\text{INC}}, i+1, Q_i[\text{lower}]-1);$ 
15.        if INC [k]  $\neq \infty$  then CONT [i] := k endif
16.      endif
17.    else //  $e_s$  is a left endpoint  $a_i$  //
18.      if  $i+1 < Q_i[\text{lower}]$  then
19.         $p := \text{MINEL} (T_{\text{IN}}, i+1, Q_i[\text{lower}]-1);$ 
20.      else
21.         $p := \text{"empty"}$ 
22.      endif
23.       $q := \text{MINEL} (T_{\text{OUT}}, Q_i[\text{lower}], Q_i[\text{upper}]);$ 
24.      if  $p = \text{"empty"}$  or  $\text{OUT} [q] \leq \text{IN} [p]$  then
25.        IN [i] := OUT [q] +  $w_i$  and UPDATE ( $T_{\text{IN}}$ );
26.        NEXT [ $i_{\text{IN}}$ ] :=  $q_{\text{OUT}}$ ;
27.      else
28.        IN [i] := IN [p] +  $w_i$  and UPDATE ( $T_{\text{IN}}$ );
29.        NEXT [ $i_{\text{IN}}$ ] :=  $p_{\text{IN}}$ ;
30.      endif
31.      if  $p = \text{"empty"}$  then
32.        OUT [i] :=  $\infty$  and UPDATE ( $T_{\text{OUT}}$ );
33.      else
34.         $k := \text{CONT} [i];$ 
35.        if  $k = \text{"empty"}$  or  $\text{IN} [p] \leq \text{OUT} [q] + w_k$  then
36.          OUT [i] := IN [p] +  $w_i$  and UPDATE ( $T_{\text{OUT}}$ );
37.          NEXT [ $i_{\text{OUT}}$ ] :=  $p_{\text{IN}}$ ;
38.        else
39.          OUT [i] := OUT [q] +  $w_i + w_k$  and UPDATE ( $T_{\text{OUT}}$ );
40.          NEXT [ $i_{\text{OUT}}$ ] :=  $q_{\text{OUT}}$ ;
41.        endif
42.      endif
43.      AVAIL [lower] :=  $i$ ; AVAIL [upper] := min {AVAIL [upper],
44.         $Q_i[\text{lower}]-1$ };
45.    endif
46.  endfor.

```

The output of the algorithm is a minimum total dominating set the weight of which is $\text{IN} [0]$ and the elements of which are those in the NEXT list beginning with NEXT [0_{IN}]; of course, elements zero and $n+1$ are not to be included in such a set; moreover, if two consecutive OUT-nodes, say i_{OUT} and j_{OUT} , appear in the NEXT list, then also CONT [i] belongs to the optimum solution.

The correctness of the algorithm follows from Lemma 5b of [MAN84]. In particular, the subroutine *MINEL* selects the proper leaf having minimum value also in the block $\langle i+1, Q_i[\text{lower}]-1 \rangle = C_i \cup P_i$ (line 22), excluding, as before, selection of an interval

p that is in C_i , as we may readily check. Notice that the “ $\leq$ ” condition in line 24 allows us to select leaf q (the rightmost one) in case of ties.

The overall time complexity is $O(n \log n)$. Of course, in the presence of negative weights, the same maneuver seen in the previous section can be performed, and the actual minimum total dominating set is thus found with the same computational effort.

4. Irredundance. In this section, we show that the problem of finding a minimum weighted maximal irredundant set of an interval graph can also be reduced to a shortest path problem on an appropriate acyclic directed graph. To do this, we first need the following result.

LEMMA. *Let $G(I)$ be an interval graph. Then a subset X of intervals has redundancies if and only if we have the following:*

- (1) *There are $i, j \in X$ such that $a_i < a_j < b_j < b_i$, or*
- (2) *There are $i, j \in X$ such that $N[i] \subseteq N[j]$, or*
- (3) *There are $i, j, h \in X$ such that $a_i < a_h < a_j$, $b_i < b_h < b_j$, and $N[h] \subseteq N[i] \cup N[j]$.*

Proof. Verifying one direction of the lemma is trivial. So assume X to have redundancies. If either condition (1) or (2) is verified, we are done. If none of them holds, assume h to be redundant. By assumption, intervals in I , and hence in X , are indexed by increasing left endpoints. Let i (j) be that interval in X which immediately precedes (follows) h in the ordering. Note that both i and j must exist, because otherwise either condition (1) or (2) should be verified. Since h is redundant, the case $b_i < a_h < b_h < a_j$ cannot occur. Thus h overlaps with at least one interval between i and j . Without loss of generality, assume $a_i < a_h < b_i$ (the case $a_j < b_h < b_j$ can be dealt with similarly). Each interval in $N[h]$ including one point p , $a_h < p < b_i$, is dominated by i . Thus consider the remaining intervals in $N[h]$. Each such interval k must contain a point q , $b_i < q < b_h$. Since there is no interval in X properly contained within another interval in X and h is redundant, $b_i < a_k < b_k < a_j$ cannot occur. Thus, k overlaps with j and $N[h] \subseteq N[i] \cup N[j]$. $\square$

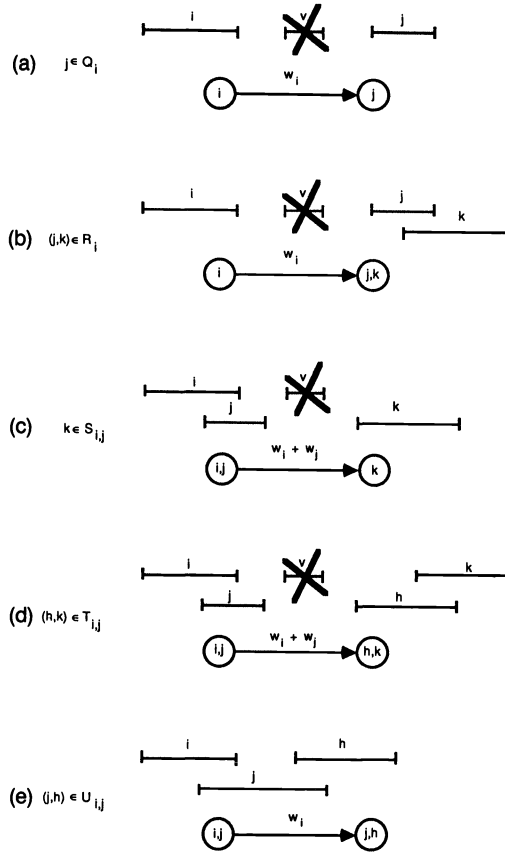
Let $J = \{(i, j) : i, j \in I, a_i < a_j < b_i < b_j, N[i] \not\subseteq N[j] \text{ and } N[j] \not\subseteq N[i]\}$. By the lemma, we can define a directed graph D having node set equal to $I' \cup J$. There is an arc (x, y) in D if and only if

$$y \in Q_x \cup R_x \cup S_x \cup T_x \cup U_x,$$

where:

- (a) for $i \in I'$, $Q_i = \{j : a_j > b_i \text{ and there is no } v \text{ with } b_i < a_v < b_v < a_j\}$,
- (b) for $i \in I'$, $R_i = \{(j, k) : a_j > b_i, N[j] \not\subseteq N[i] \cup N[k] \text{ and there is no } v \text{ with } b_i < a_v < b_v < a_j \text{ and } N[j] \not\subseteq N[v] \cup N[k]\}$,
- (c) for $(i, j) \in J$, $S_{i,j} = \{k : a_k > b_j, N[j] \not\subseteq N[i] \cup N[k] \text{ and there is no } v \text{ with } b_j < a_v < b_v < a_k \text{ and } N[j] \not\subseteq N[i] \cup N[v]\}$,
- (d) for $(i, j) \in J$, $T_{i,j} = \{(h, k) : a_h > b_j, N[j] \not\subseteq N[i] \cup N[h], N[h] \not\subseteq N[j] \cup N[k] \text{ and there is no } v \text{ with } b_j < a_v < b_v < a_h \text{ for which } N[j] \not\subseteq N[i] \cup N[v] \text{ and } N[h] \not\subseteq N[v] \cup N[k]\}$,
- (e) for $(i, j) \in J$, $U_{i,j} = \{(j, h) : a_h > b_i, \text{ and } N[j] \not\subseteq N[i] \cup N[h]\}$.

Examples of arcs in D are given in Fig. 4. The length of the arcs is clearly equal to w_i in cases (a), (b), and (e), while it is equal to $w_i + w_j$ in cases (c) and (d). Note that in case (e) there is an arc between nodes (i, j) and (j, k) , which share the *same* j . This

FIG. 4. Examples of arcs in D .

allows us to consider sequences of (two or more) pairwise overlapping intervals. A complete example is exhibited in Fig. 5.

THEOREM. *All the paths in D between zero and $n + 1$ correspond to maximal irredundant sets of $G(I)$, and vice versa.*

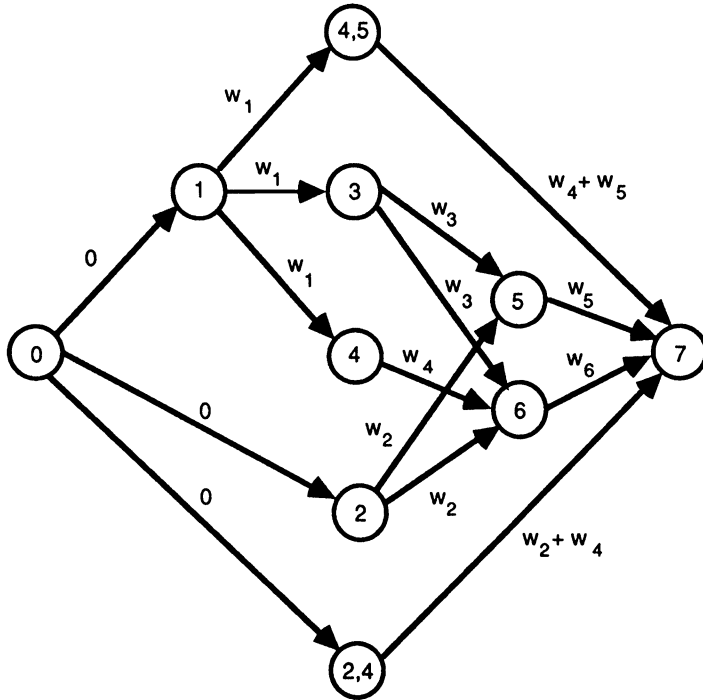
Proof. By construction, any path p in D between zero and $n + 1$ corresponds to an irredundant set $X \subseteq I$. Indeed, for any three (consecutive) intervals in p , conditions (1)–(3) of the lemma have been tested. To prove maximality, let us add to X any interval $v \in I - X$ and show that the resulting set $X \cup \{v\}$ has redundancies.

Clearly, if there is a j in X such that $a_j < a_v < b_v < b_j$ (or $a_v < a_j < b_j < a_v$) then v (j) is redundant. Moreover, this is true also if $N[v] \subseteq N[j]$ (or $N[j] \subseteq N[v]$). Therefore, let i, j be two consecutive intervals appearing in p such that $a_i < a_v < a_j$, $N[v] \not\subseteq N[i]$, $N[v] \not\subseteq N[j]$, $N[i] \not\subseteq N[v]$, and $N[j] \not\subseteq N[v]$ (of course, $i = 0$ and/or $j = n + 1$ can occur). Five cases may come up.

(a) If $(i, j) \in J$, then v is “sandwiched” by i and j , since $a_i < a_v < a_j < b_i < b_v < b_j$, and thus it is redundant (Fig. 6(a)).

(b) If $j \in Q_i$, then clearly the case $b_i < a_v < b_v < a_j$ cannot occur; moreover, neither can there exist an interval $w \in N(v)$ with $b_i < a_w < b_w < a_j$. Thus v is redundant (Fig. 6(b)).

(c) Assume there is a k in p for which $(j, k) \in R_i$. If $b_i < a_w < b_w < a_j$ for a $w \in N[v]$, then, by definition of R_i , $N[j] \subseteq N[i] \cup N[w]$ and thus j is redundant (Fig. 6(c)). Otherwise, v is redundant.

FIG. 5. The directed graph D for the Capricorn graph.

(d) Let $j \in S_{h,i}$ for an h in p . Again, if there is a $w \in N[v]$ for which $b_i < a_w < b_w < a_j$, then i is redundant, since $N[i] \subseteq N[h] \cup N[w]$, by definition of $S_{h,i}$ (Fig. 6(d)). If no such w exists, v itself is redundant.

(e) Finally, when the case $(j, k) \in T_{h,i}$ holds for h and k in p , then either i or j , or both, are redundant (if there is $w \in N[v]$ with $b_i < a_w < b_w < a_j$; see Fig. 6(e)), or v itself is redundant (otherwise).

Conversely, let X be a maximal irredundant set and assume its elements to be ordered by increasing a_i 's. Suppose X does not correspond to any path in D . Thus there exist two consecutive intervals in X , say i and j , for which neither $(i, j) \in J$ nor is there an arc joining them.

Since $(i, j) \notin J$, then i and j do not overlap, i.e., $b_i < a_j$. Moreover, since $j \notin Q_i$, there exists a $v \in I - X$ with $b_i < a_v < b_v < a_j$. Thus $N[i] \subsetneq N[v]$, $N[j] \subsetneq N[v]$ and $N[v] \subsetneq N[i] \cup N[j]$. Four cases may appear: in each of them we shall reach a contradiction.

(i) If both i and j do not overlap with any other interval in X , then $X \cup \{v\}$ is irredundant.

(ii) If only interval i overlaps with another interval in X , say h , then $N[i] \subsetneq N[h] \cup N[v]$, since $j \notin S_{h,i}$. Thus $X \cup \{v\}$ is again irredundant.

(iii) If only interval j overlaps with another interval in X , say k , then $N[j] \subsetneq N[v] \cup N[k]$, since $(j, k) \notin R_i$. As before, $X \cup \{v\}$ results to be irredundant.

(iv) If intervals i and j overlap, respectively, with h and k , then $N[i] \subsetneq N[h] \cup N[v]$ and $N[j] \subsetneq N[v] \cup N[k]$, since $(j, k) \notin T_{h,i}$. Even in this case, $X \cup \{v\}$ comes out to be irredundant. $\square$

As a consequence of the above theorem, any minimum weighted maximal irredundant set can be found by means of the following algorithm.

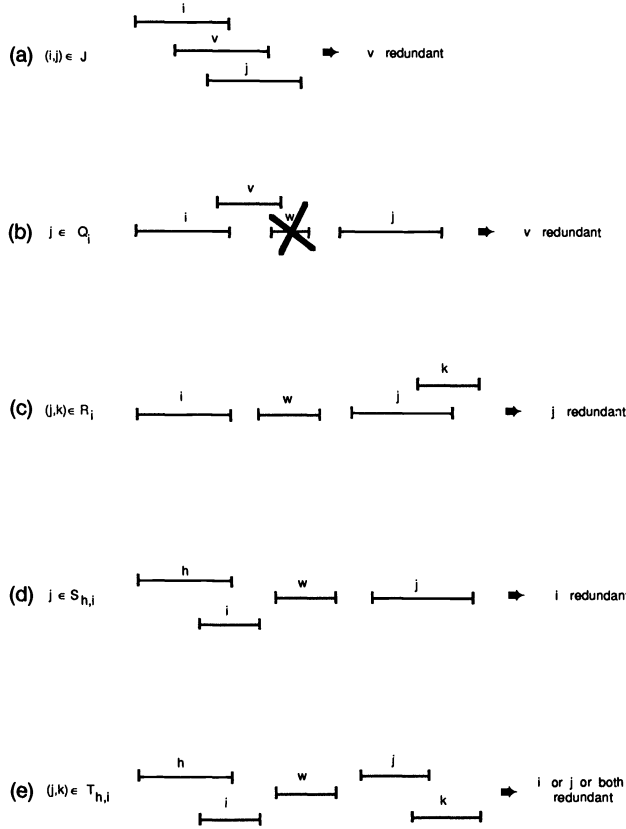


FIG. 6. Examples of redundancies.

1. **procedure** *MINIMUM-MAXIMAL-IRREDUNDANT-SET*;
2. Construct the directed graph D ;
3. Find a shortest path $p = 0, x_1, \dots, x_k, n+1$ in D ;
4. The minimum maximal irredundant set X for $G(I)$ is given by all intervals i and $j \in I$ such that either $i = x_h$ or $(i, j) = x_h$, $1 \leq h \leq k$. The weight of X is equal to the length of p .

The time required to build up D is $O(n^4)$, since it is dominated by that needed to compute all the $T_{h,i}$'s and $U_{h,i}$'s. Indeed, let us assume each $N[i]$ to be represented by an ordered list. Then verifying $N[h] \subseteq N[i] \cup N[j]$ for given h, i, j , takes $O(n)$ time. Since there are $O(n^3)$ such triples, we can set up in $O(n^4)$ time a Boolean tri-dimensional array B whose generic entry $B[h, i, j]$ is equal to one if and only if $N[h] \subseteq N[i] \cup N[j]$. Then, all the R_i 's and $S_{i,j}$'s can be computed in $O(n^4)$ time. Moreover, since $(h, k) \in T_{i,j}$ if and only if both $(h, k) \in R_j$ and $h \in S_{i,j}$, computing all the $T_{h,i}$'s also requires $O(n^4)$ time.

Since D contains $O(n^2)$ nodes and is acyclic, as we may easily verify, a shortest path can be found in $O(n^4)$ time [LAW76]. Hence, the overall running time required to find a minimum weighted maximal irredundant set of an interval graph is $O(n^4)$.

REFERENCES

- [AHO74] A. AHO, J. HOPCROFT, AND J. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

- [BER84] A. A. BERTOSSI, *Dominating sets for split and bipartite graphs*, Inform. Process. Lett., 19 (1984), pp. 37–40.
- [BER85] M. BERN, E. L. LAWLER, AND A. WONG, *Why certain subgraph computations require only linear time*, in Proc. 26th Annual IEEE Symposium on Foundations of Computer Science, Portland, OR, 1985, pp. 117–125.
- [BER86] A. A. BERTOSSI, *Total domination in interval graphs*, Inform. Process. Lett., 23 (1986), pp. 131–134.
- [BOL79] B. BOLLOBÁS AND E. J. COCKAYNE, *Graph-theoretic parameters concerning domination, independence, and irredundance*, J. Graph Theory, 3 (1979), pp. 241–249.
- [CHA84] G. J. CHANG AND G. L. NEMHAUSER, *The k -domination and k -stability problems on sun-free chordal graphs*, SIAM J. Algebraic Discrete Methods, 5 (1984), pp. 332–345.
- [COC80] E. J. COCKAYNE, R. M. DAWES, AND S. T. HEDETNIEMI, *Total domination in graphs*, Networks, 10 (1980), pp. 211–219.
- [GOL80] M. C. GOLUMBIC, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York, 1980.
- [GOL85] M. C. GOLUMBIC, ED., *Interval graphs and related topics*, Discrete Math., 55 (special issue: July 1985), pp. 113–243.
- [HUJ84] M. HUJTER, *Bibliography on the minimum dominating set problem*, RUTCOR, Rutgers University, New Brunswick, NJ, 1984.
- [JOH84] D. S. JOHNSON, *The NP-completeness column: an ongoing guide*, J. Algorithms, 5 (1984), pp. 147–160.
- [KEI86] J. M. KEIL, *Total domination in interval graphs*, Inform. Process. Lett., 22 (1986), pp. 171–174.
- [LAS83] R. LASKAR AND J. PFAFF, *Domination and irredundance in split graphs*, manuscript, 1983.
- [LAS84] R. LASKAR, J. PFAFF, S. M. HEDETNIEMI, AND S. T. HEDETNIEMI, *On the algorithmic complexity of total domination*, SIAM J. Algebraic Discrete Methods, 5 (1984), pp. 420–425.
- [LAS85] R. LASKAR AND S. T. HEDETNIEMI, *Irredundance in graphs: a survey*, Congress. Numer., 48 (1985), pp. 183–194.
- [LAW76] E. L. LAWLER, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart & Winston, New York, 1976.
- [MAN84] G. K. MANACHER AND C. J. SMITH, *Efficient algorithms for new problems on interval graphs and interval models*, manuscript, 1984.
- [RAM88] G. RAMALINGAM AND C. PANDU RANGAN, *A unified approach to domination problems in interval graphs*, Inform. Process. Lett., to appear.
- [ROB78] F. S. ROBERTS, *Graph Theory and Its Applications to Problems of Society*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1978.
- [WIM87] T. V. WIMER, *Linear algorithms on k -terminal graphs*, Ph.D. thesis, Dept. of Computer Science, Clemson University, Clemson, SC, August 1987.

THE DIAMETER OF A CYCLE PLUS A RANDOM MATCHING*

B. BOLLOBÁS† AND F. R. K. CHUNG‡

Abstract. How small can the diameter be made by adding a matching to an n -cycle? In this paper this question is answered by showing that the graph consisting of an n -cycle and a random matching has diameter about $\log_2 n$, which is very close to the best possible value. It is also shown that by adding a random matching to graphs with certain expanding properties such as expanders or Ramanujan graphs, the resulting graphs have near optimum diameters.

Key words. diameter, random graphs, expanders

AMS(MOS) subject classification. 05C

1. Introduction. The following problem frequently comes up in connection with network optimization: For given integers n and k , find a graph on n vertices with maximum degree k , having diameter as small as possible.

The complementary problem of the preceding problem can be described as follows:

For given integers k and D , find a graph, with bounded degree k and diameter at most D , having as many vertices as possible.

Known solutions for these problems fall into two types; one type is to construct explicitly such good graphs [3], [4], [8], [17] and the other is to take the probabilistic approach by analyzing the random regular graphs [6], [7], [10]. As it turns out, random graphs generally outperform those graphs constructed by various methods (with a few exceptions when n or D is small). In fact, random graphs have diameters very close to the optimum value, while the best constructions have much larger diameters [1], [10].

Although random graphs are easy to analyze probabilistically, the memory required for storing the edges is proportional to n^2 , whereas systematic constructions tend to use much less memory. For instance, some r -regular expander graphs only need to store k numbers [11]. This is particularly crucial in certain network problems involving routing and distributing computing. The result in this paper can be viewed as a “halfway” solution which blends a good construction with a small amount of randomness. In particular, our “hybrid” graphs only require cn memory to record all the edges (instead of cn^2 for random graphs), while the diameters are near optimal.

We will show that the graph obtained by adding a random matching to the n -cycle C_n has diameter very close to the optimum value, thus settling a problem of Farley and Hedetniemi [13]. We also prove a general theorem which asserts that by adding a random matching to k -regular graphs with certain expanding properties (detailed in § 3) the resulting graphs have diameter about $\log_k n$, which is the order of the best possible value. The best-known expanding graphs were constructed by Lubotzky, Phillips, and Sarnak [18] and have diameter $2 \log_{k-1} n$. Adding a matching result, with high probability, reduces the diameter by a factor of 2.

The paper is organized as follows. In § 2 we obtain sharp diameter bounds for a cycle plus a random matching. In § 3, we discuss several generalizations, including a general theorem which implies that adding a random matching to a k -regular graph with certain expanding properties results in a graph with diameter very close to the optimum value. In § 4, we conclude with various remarks and questions.

* Received by the editors May 5, 1987; accepted for publication (in revised form) January 18, 1988.

† Department of Mathematics, Cambridge University, Cambridge CB2 1ST, United Kingdom.

‡ Bell Communications Research, Morristown, New Jersey 07960.

2. The diameter of a cycle plus a random matching. Let C_n denote an n -cycle with vertices $v_1, v_2, \dots, v_n$, where v_i is adjacent to v_{i+1} and all indices are then modulo n . A matching on C_n is just a partition of $\{1, \dots, n\}$ into disjoint pairs (plus a singleton if n is odd). It is clear that C_n has diameter $\lfloor n/2 \rfloor$. We will show that by adding a matching to C_n the diameter can be reduced dramatically. We remark that it is known that any graph on n vertices with maximum degree 3 has diameter at least $\log_2 n - 2$ (see [2], [10]) and the best constructions for degree 3 graphs has diameter $1.47 \log_2 n$ (see [12], [15]). First, we consider the probability space of all graphs formed by adding a matching to C_n . We assume that any two such graphs have the same probability. We will prove such hybrid graphs have diameter about the same as random graphs of degree 3.

THEOREM 1. *Let G be a graph formed by adding a random matching to an n -cycle. Then with probability tending to 1 as n goes to infinity, G has diameter $D(G)$ satisfying*

$$\log_2 n - c \leq D(G) \leq \log_2 n + \log_2 \log n + c$$

where c is a small constant (at most 10).

Proof. Let C denote the n -cycle in G and M denote a random matching in G . For a vertex x , we define

$$S_i(x) = \{y: d(x, y) = d_G(x, y) = i\},$$

$$B_l(x) = \bigcup_{i \leq l} S_i(x).$$

Now pick a fixed vertex x . Draw the chord (an edge in M) incident to x . This determines $S_1(x)$. Then we add the neighbors of $S_1(x)$ one by one to determine $S_2(x)$ and proceed to determine $S_i(x)$. Call a chord incident to a vertex in $S_i(x)$ *inessential* if the other vertex in $S_i(x)$ is within distance $3 \log_2 n$ (in C) of the vertices determined so far. Since $|B_i(x)| \leq 3 \cdot 2^i$, the probability of an edge being inessential (at level i) is at most

$$\frac{18 \cdot 2^{i+1} \log_2 n}{n}.$$

Hence the probability that in the sequence of chords chosen in $B_l(x)$ at least two chords are inessential is at most

$$\binom{3 \cdot 2^{l+1}}{2} \left(\frac{18 \cdot 2^{l+1} \log_2 n}{n} \right)^2 = O(n^{-6/5} (\log_2 n)^2)$$

for $l = \lfloor \log_2 n/5 \rfloor$. Therefore the probability that for every vertex x at most one of the chords in $B_l(x)$ is inessential is at least $1 - O(n^{-1/5} (\log n)^2)$. From now on we will be only interested in graphs satisfying the property mentioned above, and will only consider conditional probabilities on this event, say, event A .

For a fixed vertex x , consider those vertices y in $S_i(x)$ for which there is a unique path from x to y of length i , say $x_0 = x, x_1, \dots, x_i = y$, such that (i) if x_{i-1} is adjacent to y on the cycle C then $B_i(x)$ has no vertex within $3 \log_2 n$ of x_i on the side opposite to x_{i-1} (see Fig. 1); (ii) if $x_{i-1}y$ is a chord then $B_i(x) - \{x_i\}$ has no vertex on C within distance $3 \log_2 n$ of x_i (see Fig. 2).

Denote the set of y 's in (i) by $C_i(x)$ and the set of y 's in (ii) by $D_i(x)$. Thus $C_i(x) \cup D_i(x) \subseteq S_i(x)$. Now, if A holds then

$$|C_i(x)| \geq 2^{i-2} \quad \text{and} \quad |D_i(x)| \geq 2^{i-3} \quad \text{for } i \leq \frac{1}{5} \log_2 n.$$

Now we consider i between $\frac{1}{5} \log_2 n$ and $\frac{3}{5} \log_2 n$. The probability of a chord being inessential is at most

$$\frac{18 \cdot 2^{i+1} \log_2 n}{n} < n^{-1/6}$$

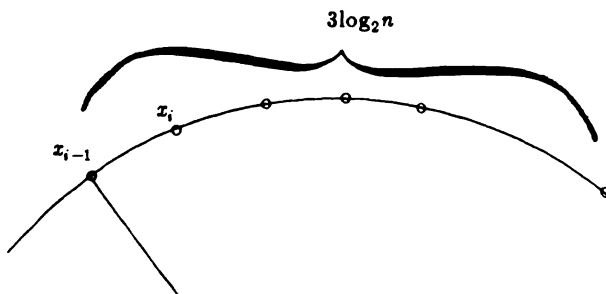


FIG. 1

for n large. Since there are at most 2^i chords leaving $S_i(x)$ for $\frac{1}{5} \log_2 n \leq i \leq \frac{3}{5} \log_2 n$, the probability that there are at least $2^i n^{-1/10}$ inessential chords leaving $S_i(x)$ is at most

$$\binom{2^i}{2^i n^{-1/10}} n^{-1/6 \cdot 2^i n^{-1/10}} \leq (n^{1/10} n^{-1/6})^{2^i n^{-1/10}} \leq (n^{-1/20})^{n^{1/10}} < n^{-5}$$

for large n . Therefore, with probability $1 - O(n^{-2})$ for every x and every i satisfying $\frac{1}{5} \log_2 n \leq i \leq \frac{3}{5} \log_2 n$, at most $2^i n^{-1/10}$ inessential chords leave $S_i(x)$. Call this event B .

For y in $C_i(x)$, a new neighbor of y in C is a “potential” element of $C_{i+1}(x)$ and a new neighbor, which is the end-vertex of the chord from y , is a “potential” element of $D_{i+1}(x)$. (Here “potential” means that the vertices in question become elements of $C_{i+1}(x)$ or $D_{i+1}(x)$, unless the corresponding edge is inessential.) Also, if $y \in D_i(x)$, then the two new neighbors of y on C are potential elements of $C_{i+1}(x)$. Hence if A and B both hold, then for $3 \leq i \leq \frac{1}{5} \log_2 n$ and for any x we have

$$|C_i(x)| \geq 2^{i-2} \quad \text{and} \quad |D_i(x)| \geq 2^{i-3}$$

and for $\frac{1}{5} \log_2 n \leq i \leq \frac{3}{5} \log_2 n$ we have

$$|C_{i+1}(x)| \geq |C_i(x)| + 2|D_i(x)| - 2^{i+1} n^{-1/10},$$

$$|D_{i+1}(x)| \geq |C_i(x)| - 2^{i+1} n^{-1/10}.$$

Therefore, for $3 \leq i \leq \frac{3}{5} \log_2 n$, we have

$$|C_i(x)| \geq 2^{i-3} \quad \text{and} \quad |D_i(x)| \geq 2^{i-4}.$$

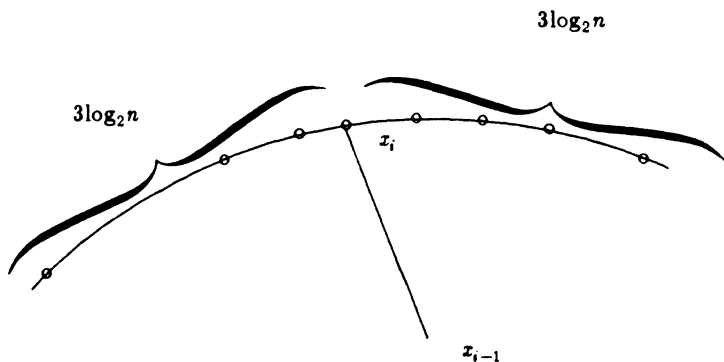


FIG. 2

Now set $i_0 = \lceil \frac{1}{2}(\log_2 n + \log_2 \log n + c) \rceil$. We want to estimate the conditional probability (on A and B) for two points x and y having distance at least $2i_0 + 1$ in G . Let us choose chords leaving $C_{i_0}(x)$ one by one. At each choice the probability of not choosing the other end vertex in $C_{i_0}(y)$ is at most $1 - (2^{i_0-3}/n)$.

Since we have to make at least $|C_{i_0}(x)|/2 \geq 2^{i_0-4}$ such choices, we have

$$\begin{aligned} \text{Prob}(d(x, y) > 2i_0 + 1 | A \cap B) &\leq \left(1 - \frac{2^{i_0-3}}{n}\right)^{2^{i_0-4}} \\ &\leq \exp(-2^{2i_0-7}/n) \\ &\leq \exp(-(\log n)2^{c-7}) \\ &\leq n^{-4} \end{aligned}$$

if $c \geq 9$.

We are now ready to consider the probability that $D(G) > 2i_0 + 1$.

$$\begin{aligned} \text{Prob}(D(G) > 2i_0 + 1) &\leq (1 - P(A)) + (1 - P(B)) + \sum_{x, y} \text{Prob}(d(x, y) > 2i_0 + 1 | A \cap B) \\ &\leq c_1(n^{-1/5}(\log n)^2) + c_2(n^{-2}) + n^{-2} = o(1). \end{aligned}$$

Therefore almost all G have diameter at most

$$2\lceil \frac{1}{2}(\log_2 n + \log_2 \log n + 9) \rceil \leq \log_2 n + \log_2 \log n + 10.$$

The proof of Theorem 1 is complete.

3. Several generalizations. The proof in § 2 can be easily carried over to the following generalizations or variations of Theorem 1.

PROPOSITION 1. *If we add a random matching to a graph on n vertices which is a disjoint union of large cycles (say at least $100\sqrt{n}$ each), the resulting graph has diameter D satisfying*

$$\log_2 n - c \leq D(G) \leq \log_2 n + \log_2 \log n + c$$

with probability tending to 1 as n goes to infinity, where c is a small constant (at most 10).

PROPOSITION 2. *Suppose T is a complete binary tree on $2^k - 1$ vertices. If we add two random matchings of size 2^{k-1} to the leaves of T , then the resulting graph has diameter D satisfying*

$$\log_2 n - c' \leq D(G) \leq \log_2 n + \log_2 \log n + c$$

with probability tending to 1 as n goes to infinity, where c and c' are small constants at most 10.

All the results in this paper are included in the following general version.

THEOREM 2. *Suppose H is a graph on n vertices with bounded degree k satisfying the property that for any $x \in V(H)$, the i th neighborhood $N_i(x)$ of x (i.e., $N_i(x) = \{y: d_H(x, y) = i\}$) contains at least $c_1 k(k-1)^{i-2}$ vertices for $i \leq (\frac{1}{2} + \epsilon) \log_{k-1} n$, where ϵ and c_1 denote some fixed positive values. Then by adding a random matching to H the resulting graph G has diameter $D(G)$ satisfying*

$$\log_k n - c \leq D(G) \leq \log_k n + \log_k \log n + c$$

with probability tending to 1 as n goes to infinity, where c is a constant depending on ϵ and c_1 .

Proof. The proof is very similar to the proof of Theorem 1. We will sketch the idea without giving all the details. Let G denote the graph formed by adding a random matching M to H . We define, for each vertex x , $S_i(x)$ and $B_i(x)$ as before (in the proof of Theorem 1). The definition of a chord being *inessential* stays the same except that $\log_2$ is replaced by $\log_k$ and 3 is replaced by $k + 1$.

It is easy to see that for $l = \lfloor \log_k n/5 \rfloor$, the probability that, for every vertex x , at most one of the chords in $B_l(x)$ is inessential is at least $1 - O(n^{-1/5}(\log_k n)^2)$. Similarly, for $\frac{1}{2} \log_k n \leq i \leq (\frac{1}{2} + \epsilon) \log_k n$, the probability that at least $k^i n^{-1/10}$ inessential chords leaving $S_i(x)$ is at most n^{-5} . Now we bound the conditional probability (on A and B). We define $C_i(x)$ and $D_i(x)$ the same way, except that we replace C by H and require that $B_i(x)$ have no vertex with distance $(k + 1) \log_k n$ of x_i in $G - \{x_i, y\}$. Again we have $|C_i(x)| \geq c_1 k^{i-2}$ and $|D_i(x)| \geq c_1 k^{i-3}$ for $i \leq \frac{1}{3} \log_k n$.

For i between $\frac{1}{3} \log_k n$ and $(\frac{1}{2} + \epsilon) \log_k n$, we have

$$\begin{aligned} |C_{i+1}(x)| &\geq (k-1)|C_i(x)| + k|D_i(x)| - k^{i+1}n^{-1/10}, \\ |D_{i+1}(x)| &\geq |C_i(x)| - k^{i+1}n^{-1/10}. \end{aligned}$$

Therefore, for $3 \leq i \leq (\frac{1}{2} + \epsilon) \log_k n$ we have

$$|C_i(x)| \geq c_1 k^{i-3} \quad \text{and} \quad |D_i(x)| \geq c_1 k^{i-4}.$$

Now choose $i_0 = \lceil \frac{1}{2} \log_k n + \log_k \log n + c \rceil$. The probability of two vertices x and y of distance $> 2i_0 + 1$ is at most

$$\left(1 - \frac{c_2 k^{i_0-3}}{n}\right)^{k^{i_0-4}} \leq n^{-4}.$$

Thus the probability that $D(G) > 2i_0 + 1$ is no more than $O(n^{-1/5}(\log n)^2) + O(n^{-2}) + n^{-2}$. Therefore almost all G have diameter

$$\log_k n + \log_k \log n + 10.$$

This concludes the proof of Theorem 2.

One natural question is which k -regular graphs satisfy the expanding property $N_i(x) \geq ck^{i-1}$ for every vertex x (as described in Theorem 2)? Of course, random graphs have such an expanding property. In the past few years much progress has been made on various explicit constructions of so-called expander graphs [1], [14], [16], [18], [19]. All these expander graphs have various expanding properties for different applications. In particular, a graph is an expander graph if it has relatively small second largest eigenvalues for its adjacency matrix [18].

Let us denote by λ the second largest (in absolute value) eigenvalue of the adjacency matrix of a k -regular graph G . (Of course, the largest eigenvalue is k .) Tanner [20] proved that for any set X of vertices, the number of neighbors $N(X)$ of X is at least

$$N(X) \geq \frac{k^2 |X|}{(k^2 - \lambda^2) |X|/n + \lambda^2}.$$

Clearly if $|\lambda| < k - \epsilon$, then G satisfies the expanding property required in Theorem 2.

Recently, Lubotzky, Phillips, and Sarnak [17] constructed graphs with λ satisfying $|\lambda| \leq 2\sqrt{k-1}$, which is the best possible value. They call these graphs Ramanujan graphs. It is easy to see that Ramanujan graphs satisfy $N_i(x) \geq (k-1)^{i-1}/2$ for each vertex x . Ramanujan graphs have diameter $2 \log_{k-1} n + c$, while the lower bound for the diameter is $\log_{k-1} n$. By adding a matching to a Ramanujan graph, the resulting graph has diameter $(1 + o(1)) \log_k n$.

4. Concluding remarks. Many problems concerning the diameter of graphs remain open. We mention several of them here.

(1) Find explicit constructions for graphs with n vertices and degree at most k having diameter $(1 + o(1)) \log_{k-1} n$.

(2) For given integers n, k, t , let $f(n, k, t)$ denote the minimum value over all diameters of graphs which are formed by deleting (any choice of) t edges from a graph with n vertices and degree at most k . The problem is to determine $f(n, k, t)$ and to characterize the optimal graphs.

(3) Find efficient algorithms for determining the diameter of a graph. The best-known algorithms require $O(n^{2.38})$ time or $O(ne)$ time (see [10]). In particular, for planar graphs is there an $o(n^2)$ algorithm?

REFERENCES

- [1] N. ALON, *Eigenvalues and expanders*, Combinatorica, 6 (1986), pp. 83–96.
- [2] J. C. BERMOND AND B. BOLLOBÁS, *The diameter of graphs—a survey*, in Proc. Congr. Numer., 32, 1981, pp. 3–27.
- [3] J. C. BERMOND, C. DELORME, AND G. FARHI, *Large graphs with given degree and diameter III*, Proc. Coll. Cambridge, 1981; Ann. Discrete Math., 13 (1981), pp. 23–32.
- [4] J. C. BERMOND, J. BOND, M. PAOLI, AND C. PEYRAT, *Graphs and interconnection networks: diameter and vulnerability*, in Proc. Tenth British Combinatorics Conf., 1983.
- [5] B. BOLLOBÁS, *Extremal Graph Theory*, Academic Press, London, 1978.
- [6] ———, *The diameter of random graphs*, Trans. Amer. Math. Soc., 267 (1981), pp. 41–52.
- [7] B. BOLLOBÁS AND W. F. DE LA VEGA, *The diameter of random graphs*, Combinatorica, 2 (1982), pp. 125–134.
- [8] N. G. DE BRUIJN, *A combinatorial problem*, Nederl. Akad. Wetensch. Proc., 49 (1946), pp. 758–764.
- [9] F. R. K. CHUNG AND M. R. GAREY, *Diameter bounds for altered graphs*, J. Graph Theory, 8 (1984), pp. 511–534.
- [10] F. R. K. CHUNG, *Diameters of communication networks*, in Mathematics of Information Processing, AMS Short Course Lecture Notes (1984), pp. 1–18.
- [11] ———, *Diameters and eigenvalues*, preprint.
- [12] ———, *Diameters of graphs: Old problems and new results*, preprint.
- [13] A. M. FARLEY AND S. T. HEDETNIEMI, *Broadcasting in grid graphs*, in Proc. 9th S. E. Conference on Combinatorics, Graph Theory and Computing, 1978; Congr. Numer., 21 (1978), pp. 275–288.
- [14] O. GABBER AND Z. GALIL, *Explicit construction of linear sized superconcentrators*, J. Comput. System Sci., 22 (1981), pp. 407–420.
- [15] M. JERRUM AND S. SKYUM, *Families of fixed degree graphs for processor interconnection*, Internat. Report, Department of Computer Science, Univ. of Edinburgh, 1983.
- [16] W. LELAND AND M. SOLOMON, *Dense trivalent graphs for processor interconnection*, IEEE Trans. Comput., 31 (1982), pp. 219–222.
- [17] A. LUBOTZKY, R. PHILLIPS, AND P. SARNAK, *Ramanujan graphs*, preprint.
- [18] G. A. MARGULIS, *Explicit constructions of concentrators*, Problemy Peredachi Informatsii, 9 (1978), pp. 71–80. (In Russian.) Problems of Inform. Transmission, 9 (1975), pp. 325–332. (In English.)
- [19] N. PIPPENGER, *Superconcentrators*, SIAM J. Comput., 6 (1977), pp. 298–304.
- [20] R. M. TANNER, *Explicit construction of concentrators from generalized n -gons*, SIAM J. Algebraic Discrete Methods, 5 (1984), pp. 287–293.

UNIQUE FINITE DIFFERENCE MEASUREMENT*

PETER C. FISHBURN†, HELEN M. MARCUS-ROBERTS‡, AND FRED S. ROBERTS§

Abstract. This paper considers a variety of combinatorial and number-theoretic questions that arise from considerations of uniqueness in the theory of measurement. Given $n + 1$ objects linearly ordered from worst (smallest) to best (largest), let d_i stand for the difference between objects i and $i + 1$. Three types of assumptions that allow different inequality and equality comparisons between certain subsets of differences are considered. The three types of assumptions arise from problems concerning the uniqueness of finite algebraic difference measurement and finite subjective probability measurement. If the number of equality comparisons is sufficient to imply that the d_i in $d = (d_1, d_2, \dots, d_n)$ are unique up to multiplication by a positive constant, then d is said to be unique. Results on the number of unique d 's and relationships among the components of such d 's are obtained for each of the three types.

Key words. measurement theory, difference measurement, uniqueness in measurement, integer sequences, linear equations

AMS(MOS) subject classifications. primary 05A99; secondary 10A35, 92A25, 92A90

1. The measurement problems. The aim of this paper is to explore a variety of combinatorial and number-theoretic problems that arise from considerations of uniqueness in finite difference measurement. We begin § 1 with a few remarks on measurement theory that will focus our concerns. In § 2 we explain the formulation used in the rest of the paper and outline the main results.

The reader who wishes to skip the measurement motivation on the first reading can use Lemma 1 below as the definition of a unique sequence $d = (d_1, \dots, d_n)$. Then he/she can start with the definitions of Types A, B, and C in the middle of § 2. In this case, it is necessary to keep in mind that we consider two sequences equivalent if we can obtain one from the other by multiplication by a positive constant and we usually write a sequence in the equivalent form of a sequence of integers with $\min d_i$ as small as possible.

In studying the properties of a scale of measurement, measurement theorists [14], [17]–[19], [24] often study the homomorphic mappings of one empirical relational system $(X, <_1, <_2, \dots, <_m)$ into another relational system on the reals. Here, X is a set and $<_i$ is an n_i -ary relation on X . The homomorphism is considered to define a scale of measurement, and the properties of this scale are captured by considering the uniqueness of the homomorphism, or, equivalently, the properties of the class of admissible transformations of scale. In many cases, the uniqueness properties of measurement homomorphisms are well known. However, in two important cases in measurement theory, these properties are not completely understood. These cases are algebraic difference measurement and subjective probability measurement.

In *algebraic difference measurement* [2], [6], [14], [15], [19], [23], [24], [26], [27], we have a system $(X, <_1)$, where $<_1$ is a quaternary relation on X or a binary relation on $X \times X$, and we seek a real-valued function f on X so that for all x, y, u, v in X ,

$$(1) \quad (x, y) <_1 (u, v) \Leftrightarrow f(x) - f(y) < f(u) - f(v).$$

* Received by the editors August 18, 1987; accepted for publication February 16, 1988.

† AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

‡ Department of Mathematics and Computer Science, Montclair State College, Upper Montclair, New Jersey 07043.

§ Department of Mathematics, Rutgers University, New Brunswick, New Jersey 08903. This research was supported by the National Science Foundation under grant IST-86-04530 to Rutgers University.

In *subjective probability measurement* [1], [3]–[5], [7], [13], [16], [19], [22], [23], we have a system $(X, <_1)$, where X is a Boolean algebra of subsets of a point set and $<_1$ is a binary relation on X thought of as “less likely than.” We seek a finitely additive probability measure p on X such that for all A, B in X ,

$$(2) \quad A <_1 B \Leftrightarrow p(A) < p(B).$$

Most interesting uniqueness theorems in the literature of measurement theory require X to be infinite and conclude that the homomorphism f is rather precisely determined. Among the important types of uniqueness results are those which say that f is unique, in which case we call it an *absolute scale*; those which say that f is unique up to multiplication by a positive constant, in which case we call it a *ratio scale*; and those which say that f is unique up to a positive affine transformation $\alpha f + \beta$, $\alpha > 0$, in which case we call it an *interval scale*.

In algebraic difference measurement, a sample uniqueness theorem (see [19]) is that under a certain solvability axiom, f satisfying (1) is unique up to a positive affine transformation, i.e., defines an interval scale. (For other uniqueness theorems leading to the same conclusion, see [6] or [14].) In subjective probability measurement, a sample result (see [19]) is that if there is a so-called n -fold almost uniform partition for every n , then p satisfying (2) is unique, i.e., defines an absolute scale. (For other uniqueness theorems leading to the same conclusion, see [1], [16], or [22].)

It is easy to give examples where these results fail without some special assumption. For instance, if $X = \{0, 1, 3\}$ and $(x, y) <_1 (u, v)$ holds if and only if $x - y < u - v$, then $f(x) = x$ and $g(0) = 0$, $g(1) = 2$, and $g(3) = 8$ both satisfy (1). But g is not equal to $\alpha f + \beta$ for any $\alpha > 0$ and β , so f is not an interval scale. On the other hand, if $X = \{0, 1, 2\}$ and $<_1$ is defined in the same way as above, then $f(x) = x$ satisfies (1) and is an interval scale, since any other function g satisfying (1) has the property that $g(0) < g(1) < g(2)$ and $g(1) - g(0) = g(2) - g(1)$.

Similarly, if X consists of all subsets of $\{a, b\}$ and $<_1$ is defined by $\emptyset <_1 \{a\} <_1 \{b\} <_1 \{a, b\}$, then the finitely additive probability measures $p(\{a, b\}) = 1$, $p(\{b\}) = \frac{2}{3}$, $p(\{a\}) = \frac{1}{3}$, $p(\emptyset) = 0$ and $p'(\{a, b\}) = 1$, $p'(\{b\}) = \frac{3}{4}$, $p'(\{a\}) = \frac{1}{4}$, $p'(\emptyset) = 0$ both satisfy (2). It follows that p is not an absolute scale. On the other hand, suppose we define $\sim'_1$ by

$$A \sim'_1 B \Leftrightarrow [\text{not } A <_1 B \text{ and not } B <_1 A].$$

Then if X is as above and $\emptyset <_1 \{a\} \sim'_1 \{b\} <_1 \{a, b\}$, then $p(\emptyset) = 0$, $p(\{a\}) = p(\{b\}) = \frac{1}{2}$, $p(\{a, b\}) = 1$ defines the only finitely additive probability measure satisfying (2).

The paper [20] surveys some related uniqueness questions in measurement theory, and poses the questions: Under what conditions does a function satisfying (1) define an interval scale? Under what conditions does a function satisfying (2) define an absolute scale? The latter question is considered in [8] under the assumption that X is finite. The purpose of the present study is to consider the former question under the assumption that X is finite. Results from [8] will be mentioned, since one of the special cases of uniqueness resulting from algebraic difference measurement will be similar to the situation of subjective probability.

Note that algebraic difference measurement is closely related to another kind of measurement called *absolute difference measurement* [6], [11], [12], [14], [26], [28]. Here, we have a relational system $(X, <_1^*, <_2^*)$, where X is a set, $<_1^*$ is a binary relation on X , and $<_2^*$ is a quaternary relation on X or a binary relation on $X \times X$. We seek a

real-valued function f on X so that for all x, y, u, v in X ,

$$(3) \quad x \prec_1^* y \Leftrightarrow f(x) < f(y)$$

and

$$(4) \quad (x, y) \prec_2^* (u, v) \Leftrightarrow |f(x) - f(y)| < |f(u) - f(v)|.$$

(Tversky and Krantz [28] show how to obtain a function f satisfying (4) without assuming we know $\prec_1^*$; see [14].) Such a function f satisfying (3) and (4) is called a higher-ordered metric by Siegel [25]. Algebraic difference measurement and absolute difference measurement are equivalent in the following sense. Given $\prec_1$, we can define $\prec_1^*$ and $\prec_2^*$ from it so that a function f satisfies (1) if and only if it satisfies (3) and (4). Conversely, given $\prec_1^*$ and $\prec_2^*$, we can define $\prec_1$ from them so that a function f satisfies (1) if and only if it satisfies (3) and (4). Hence, the uniqueness results for absolute difference measurement are equivalent to those for algebraic difference measurement. For the purposes of this paper, it is more convenient to think of the former. It will be useful to consider the binary relation $\sim_2^*$ defined by

$$(x, y) \sim_2^* (u, v) \Leftrightarrow [\text{not } (x, y) \prec_2^* (u, v) \text{ and not } (u, v) \prec_2^* (x, y)].$$

The relation $\sim_2^*$ is the relation of “equal absolute differences.” We shall observe that a function f satisfying (3) and (4) is unique up to a positive affine transformation, and therefore defines an interval scale, if and only if there are “enough” pairs of differences in the equal differences relation $\sim_2^*$.

Consider a function f satisfying (3) and (4). If $f(x) = f(y)$, then x and y are “perfect substitutes” for each other [19] in the relations $\prec_1^*$ and $\prec_2^*$ and we can disregard x or y . Thus, we can assume that $\prec_1^*$ is a linear order. In the next section, we formulate the problem of specifying the uniqueness of the function f as a problem of specifying the uniqueness of the differences between successive $f(x)$.

2. The difference formulation of the problems. Suppose $X = \{x_1, x_2, \dots, x_{n+1}\}$, $n \geq 1$, suppose that $\prec_1^*$ and $\prec_2^*$ are given, with the former a linear order and the latter a quaternary relation, and suppose that there is a real-valued f on X so that (3) and (4) hold. Since $\prec_1^*$ is a linear order, the $f(x)$ are all different. It is convenient to assume that we have labeled elements of X so that

$$f(x_1) < f(x_2) < \dots < f(x_{n+1}).$$

Let $d_i = f(x_{i+1}) - f(x_i)$. Thus, $d_i > 0$, all i . Each relation $(x, y) \prec_2^* (u, v)$ translates into an inequality

$$(5) \quad \sum_{i \in \Psi} d_i < \sum_{i \in \Delta} d_i,$$

while each relation $(x, y) \sim_2^* (u, v)$ translates into an equality

$$(6) \quad \sum_{i \in \Psi} d_i = \sum_{i \in \Delta} d_i,$$

where Ψ and Δ are nonempty disjoint subsets of $\{1, \dots, n\}$. If $\Psi \cap \Delta \neq \emptyset$ initially, then we cancel common d_i . Note that f satisfying (3), (4) is unique up to a positive affine transformation if and only if the corresponding sequence $d = (d_1, d_2, \dots, d_n)$ is unique up to multiplication by a positive constant. We shall seek to determine conditions under which this sequence of differences is unique up to such multiplication, and if it is we shall call d *unique*. Thus, $(d_1, \dots, d_4) = (0.1, 0.2, 0.3, 0.4)$ and $(d_1, \dots, d_4) = (1, 2, 3, 4)$

are equivalent under this type of uniqueness. We shall usually write d as a sequence of integers with $\min d_i$ as small as possible.

It turns out to be interesting to consider not just algebraic or absolute difference measurement, but several variations as well. In all, then, three types of difference measurement will be considered. We describe these by the difference comparisons (5) and (6) they allow, i.e., by restrictions on the sets Ψ and Δ . The one motivated by algebraic difference measurement or absolute difference measurement is based on the relation $\prec_2^*$ and allows comparisons between any $|f(x_i) - f(x_j)|$ and $|f(x_k) - f(x_l)|$ for $i < j \leq k < l$. A more restrictive form of difference measurement is based on a ternary relation that only permits comparisons between $|f(x_i) - f(x_j)|$ and $|f(x_j) - f(x_k)|$ for $i < j < k$. For the same reason that absolute difference measurement is equivalent to algebraic difference measurement, this corresponds to having a ternary relation $\prec_2^{**}$ on X and having a real-valued function f on X satisfying

$$(x, y, z) \in \prec_2^{**} \Leftrightarrow f(x) - f(y) < f(y) - f(z).$$

A less restrictive form of difference measurement allows comparisons between the “sum” of differences $|f(x_{i+1}) - f(x_i)|$ for all i in a set Ψ and a similar sum for all i in a second set Δ . The d_i satisfying such comparisons correspond to the $p(\{a_i\})$ arising in comparisons (2) if $X = \{a_1, \dots, a_n\}$. For a reason to be given shortly, the corresponding uniqueness problem is equivalent to the uniqueness problem for subjective probability measurement with which we are concerned.

In terms of the d_i comparisons allowed, we define a sequence d as belonging to one of the following types:

Type A (ternary)— d is the solution of equations or inequalities (5) or (6), where $\Psi = \{i, \dots, j-1\}$ and $\Delta = \{j, \dots, k-1\}$, for $1 \leq i < j < k \leq n+1$;

Type B (quaternary)— d is the solution of equations or inequalities (5) or (6), where $\Psi = \{i, \dots, j-1\}$ and $\Delta = \{k, \dots, l-1\}$, for $1 \leq i < j \leq k < l \leq n+1$;

Type C (additive)— d is the solution of equations or inequalities (5) or (6), where $\Psi, \Delta \subseteq \{1, \dots, n\}$, with $\Psi \cap \Delta = \emptyset$.

The unique d 's of type B correspond to the structures unique up to a positive affine transformation. The unique type C's correspond to the finitely additive probability measures p which satisfy subjective probability measurement with all $p_i > 0$ and which are uniquely determined (i.e., define an absolute scale). This is because, in the case of subjective probability measurement, uniqueness up to multiplication by a positive constant combined with the conventional $\sum p_i = 1$ normalization for probability gives us an absolute scale.

To illustrate the differences between the unique solutions of the three types, note that the sequence $d = (1, 2, 3, 4)$ is a unique type C. That it is a type C follows because it is the solution to the equations

$$d_3 = d_1 + d_2, \quad d_4 = d_1 + d_3, \quad d_2 + d_3 = d_1 + d_4.$$

It is the unique solution to this set of equations. (Recall that “unique” means “unique up to multiplication by a positive constant.”) We cannot obtain this d , or any permutation thereof, as the unique solution of equations or inequalities permitted by type B. It is, of course, a solution to type B inequalities, such as

$$d_1 < d_2, \quad d_2 < d_3, \quad d_3 < d_4.$$

However, the solutions to such inequalities are not unique up to multiplication by a positive constant.

One unique d for type B is $(2,1,1,3)$. It arises from the equations

$$d_1 = d_2 + d_3, \quad d_2 = d_3, \quad d_1 + d_2 = d_4,$$

and it is a unique solution. The same d cannot be obtained as a unique solution of type A equations or inequalities, although it has permutations that are obtained as unique type A solutions. For instance, the permutation $(1,1,2,3)$ is obtained as the unique solution to the equations

$$d_1 = d_2, \quad d_3 = d_1 + d_2, \quad d_4 = d_2 + d_3.$$

We begin our technical development in the next section with a fundamental lemma which says that, for each type, d is unique if and only if it satisfies $n - 1$ linearly independent equations of the form (6) allowed by that type. This was just illustrated by our three-equation sets for $n = 4$.

We now outline the rest of the paper. Much of our ensuing analysis is based on the distinction between regular and irregular unique d 's. A unique d is *regular* if it can be constructed inside-out from a pair of adjacent 1's by adding one term at a time where the value equals the sum of one or more terms already in place and is justified by one equation allowed by the type under consideration. Otherwise d is *irregular*. For example, $61124 = (6,1,1,2,4)$ is a type B regular. It is built up as follows:

$$\begin{array}{ll} 11 & (d_2 = d_3) \\ 112 & (d_4 = d_2 + d_3) \\ 1124 & (d_5 = d_2 + d_3 + d_4) \\ 61124 & (d_1 = d_4 + d_5). \end{array}$$

On the other hand, 61124 is not a type A regular since 6 is not the sum of contiguous terms *immediately to its right*. It is, however, a type A *irregular* since 6 is obtained from the type A equation $d_1 + d_2 = d_3 + d_4 + d_5$. Type C regulars and irregulars are studied in [8] and type A regulars in [9], where they are called two-sided generalized Fibonacci sequences.

Section 4 is concerned with counting the number of regular and irregular (unique) d 's for $n = 1, 2, \dots$ for each of our three types. We have had more success with the regular counts and prove that they are $n^{n(1+o(1))}$, $n^{2n(1+o(1))}$, and $2^{n^2(1+o(1))/2}$ for types A, B, and C, respectively. Here, $o(x_n)$ denotes a function of n that goes to zero when it is divided by x_n , and so $o(1)$ denotes a function of n that approaches zero as n gets large. We note the recurrence

$$\alpha_{n+1} = 2n\alpha_n - (n-1)^2\alpha_{n-1}$$

for $n \geq 2$ ($\alpha_1 = \alpha_2 = 1$), where α_n is the number of regular type A's of length n .

Beyond the first few n , it turns out that the irregular d 's outnumber the regulars. This occurs first for type C at $n = 4$ and for types A and B at $n = 6$. The ratio of number of irregulars to number of regulars for types A and C goes to infinity, but we have been unable to prove the same thing for type B. Upper bounds for the number of irregulars are $n^{2n(1+o(1))}$, $n^{3n(1+o(1))}$, and $3^{n^2(1+o(1))}$ for types A, B, and C, respectively. It follows that type C *regulars* far outnumber all type B unique d 's when n is large.

Section 5 considers specific positive integers $k_1, k_2, \dots, k_m$ and studies regular d 's that contain these integers. If $m \geq 5$, it turns out that there might be no type A regular that contains $k_1, k_2, \dots, k_m$ in any order. However, such d 's always exist for $m \leq 4$ and

for all m for types B and C. The shortest regular type A or B d that contains $k_1 = k$ has $\lceil \log_2 k \rceil + 2$ terms. Moreover, the difference between the shortest type A and type B regulars that contain a permutation of k_1, k_2, k_3, k_4 can be arbitrarily large. Related results and open problems are noted in this section.

Section 6 discusses three extremization problems for irregular d 's. The first is to make the smallest d_i as small as possible relative to the sum of all d_j . The minimum value for the expression $\min d_i / \sum d_j$ is $(\frac{1}{2})^{n-1}$ for regulars (consider the solutions 11248...). For type C irregulars we can get the ratio arbitrarily smaller than $(\frac{1}{2})^{n-1}$ at large n . However, we know of no type A or B irregular with a ratio smaller than $(\frac{1}{2})^{n-1}$ and conjecture that this is best possible for these types.

The second extremization problem is to find the maximum over all unique d 's of length n of the ratio $\text{pin } d_i / \min d_i$, where $\text{pin } d_i$, for *penultimate minimum* d_i , is defined as e_2 when $e_1 e_2 \cdots e_n$ is a nondecreasing permutation of $d_1 d_2 \cdots d_n$. For regulars, $\text{pin}/\min = 1$. For irregulars,

$$\max (\text{pin}/\min) \geq 2^{\lfloor n/2 - 1 \rfloor} \quad \text{for types A and B,}$$

$$\max (\text{pin}/\min) \geq 2^{n-4} + 2^{n-6} + 2^{(n-4)/2} \quad \text{for type C, } n \geq 5.$$

The third extremization problem considers maximization at each position: for each $j \leq \lceil n/2 \rceil$, maximize $d_j / \min d_i$. Because unique d 's for type C permute into other unique d 's, this reduces for type C to $\max (\max d_j / \min d_i)$. This max exceeds $(2.12)^{n-2}$ for $n \geq 5$, which compares to the max of 2^{n-2} for regulars. For type A we prove that $\max (d_j / \min d_i) \geq 2^{n-3}$ for $2 \leq j \leq \lceil n/2 \rceil$, and for type B,

$$\max (d_j / \min d_i) \geq 2^{n-3} + 2^{n-j-2} - 2 \quad \text{for } 2 \leq j \leq \lceil n/2 \rceil.$$

Section 7 returns to our conjecture that $\min d_i / \sum d_j \geq (\frac{1}{2})^{n-1}$ for all type A and B unique d 's with n terms. We show that this conjecture is equivalent to each of the following for all $n \geq 3$ and all unique d with $\min d_i = 1$ throughout:

$$\sum d_i \leq 2^{n-1} \quad (\text{restatement}),$$

$$\max d_i \leq 2^{n-2},$$

$$\max d_i < 2^{n-2} \quad \text{when } \max d_i \neq \text{sum of all other } d_i.$$

The paper concludes with a brief discussion that highlights interesting open problems.

The new results in the paper pertain mostly to types A and B. Results for type C either come directly from [8] or follow from [8] by proof modifications, and are included here partly for comparison to the other types. In addition, some results mentioned in §§ 4 and 5 for regular type A d 's are proved in [9].

3. Unique differences.

LEMMA 1. Suppose $n \geq 2$, S_1 and S_2 are disjoint sets of ordered pairs (Ψ, Δ) of nonempty disjoint subsets of $\{1, \dots, n\}$, and there exist real $d_i > 0$ for $i = 1, \dots, n$ such that

$$\sum_{i \in \Psi} d_i < \sum_{i \in \Delta} d_i \quad \text{for all } (\Psi, \Delta) \in S_1,$$

$$\sum_{i \in \Psi} d_i = \sum_{i \in \Delta} d_i \quad \text{for all } (\Psi, \Delta) \in S_2.$$

Then the $d_i > 0$ are unique up to multiplication by a positive constant if and only if S_2 contains $n - 1$ pairs for which the corresponding equations $\sum_{i \in \Psi} z_i = \sum_{i \in \Delta} z_i$ are linearly independent.

Proof. The proof is similar to the proof of Lemma 1 in [8]. If there are $n - 1$ linearly independent equations of the designated type, they must have a solution of the following form: for some $a \in \{1, \dots, n\}$, $z_i = \lambda_i z_a$ for all $i \neq a$. The uniqueness of positive d_i then follows from the hypotheses of the lemma. (Recall that we assumed there *is* a solution.)

Because of homogeneity ($\sum_{\Psi} z_i - \sum_{\Delta} z_i = 0$) there can be at most $n - 1$ linearly independent equations for S_2 . Suppose the largest independent set has $n - 2$ members, so that S_2 solution sets have the following form: there are $a \neq b$ so that $z_i = \lambda_i z_a + \sigma_i z_b$, for all $i \neq a, b$. Suppose $(d_1, \dots, d_n)$ is a particular solution satisfying the hypotheses. Then leave d_a fixed but change d_b slightly to d'_b and redefine the other d_i as $d'_i = \lambda_i d_a + \sigma_i d'_b$. Then the new d must satisfy the S_2 equations and will also satisfy the S_1 inequalities if the change in d_b is small. However, the new d is not equal to the initial d times a positive constant. A similar conclusion holds if the maximum set of linearly independent equations for S_2 has fewer than $n - 2$ members. $\square$

Lemma 1 applies, of course, to each of types A, B, and C according to the comparisons they allow for $S_1 \cup S_2$. We note also that S_2 can produce many apparently different subsets of $n - 1$ linearly independent equations. However, our hypothesis that there is a $d_i > 0$ solution guarantees that all such subsets are mutually equivalent.

Much of the following will deal with regular d 's. Recall that a regular d "begins" with two adjacent equal d_i , which we always set at 1, and builds outward one term at a time by any sequence of left/right moves so that each new term is an allowable sum of terms in the partial sequence constructed so far. We hasten to add that this definition applied to type C is *not* the same as the definition of regular p used in [8]. (These two are different because any permutation of a type C solution is another type C solution.) The definition given above is used uniformly for all three types in the present paper.

As before, a unique d for a given type is *irregular* if it is not regular.

For $n < 4$, there is no distinction between types A and B and both have only regular d 's. At $n = 2$, the only unique type A or B solution is 11, with defining equation $d_1 = d_2$. At $n = 3$, the unique type A or B solutions are 111, 112, 211, with defining equations

$$\begin{aligned} d_1 = d_2, \quad d_2 = d_3; \\ d_1 = d_2, \quad d_1 + d_2 = d_3; \\ d_2 = d_3, \quad d_1 = d_2 + d_3; \end{aligned}$$

respectively. Type C also has the irregular 121 at $n = 3$. The defining equations are $d_1 = d_3$, $d_2 = d_1 + d_3$.

It is useful to list all type A and B unique d 's for $n = 4$ and $n = 5$. Type C lists can be composed from the lists in [8].

$n = 4$. There are two type A irregulars, 2213 and its *dual* (reverse) 3122, and 14 type A regulars:

1112, 1113, 1122, 1123, 1124, 2114, their duals, and the self-dual 1111 and 2112.

All of these are, of course, unique type B's as well. In addition, there are four other type B regulars—1121, 2113, and their duals.

$n = 5$. Table 1 lists all unique type B solutions at $n = 5$ up to duality; the self-dual d 's are in parentheses. These were checked systematically by computer. All type A unique solutions are shown in boldface. Type A irregulars in the list of type B regulars are indicated with an asterisk.

The first irregulars for types A and B occur at $n = 4$: 2213 and 3122. There are many more irregulars for these types at $n = 5$, including 22233, where the smallest d_i in

TABLE 1

Type A and B unique d 's at $n = 5$. Type A's are shown **boldface**; self-duals are in parentheses; type A irregulars in the type B regular list have an asterisk. Duals of the nonself-duals are not shown.

Type B regulars				Type B irregulars			
(11111)	11223	13112	22114	Some permutation is a regular			
11112	11224	14112	22115	11312	12122	13122	21214
11113	11225	(21112)	22116	11321	12213	13123	21315
11114	11226	21113	(31113)	11322	12231	13214	42115
11121*	11231*	21114	31114*	11422	12314		
11122	11232	21115	31115	No permutation is a regular			
11123	11233	21122	31116	Two different components			
11124	11234	21123	31123*	Three different components			
11125	11235	21124	31124	22233			
11131	11236	21126	31125	Four different components			
11132	11237	21132	31127				
11133	11241	21133	32114	14133	22213	31223	33224
11134	11242	21134	32115*	21322	22215	33114	33244
11135	11243	21135	32116	22132	22433	33115	44116
11136	11244	21137	32117	22133			
(11211)*	11246	21142	41124	Five different components			
11212	11247	21143	41125	22134	22316	24123	31228
11213*	11248	21144	41126	22135	22415	31224	42215
11214	12112	21145	41128	22136	22417	31225	42235
11215	12113	21146	42116*	22138	23216	31226	43115
11221*	12114	21148	42117	22314			
11222	12115	22113	42118				
				41326	43126		

minimum integer format exceeds 1. (We say that a solution is in *minimum integer format* if all components are integers and no positive multiple of the solution containing all integers has a smaller integer.) Another $n = 5$ irregular is 41326, in which all d_i are different. At $n = 6$, the irregulars outnumber the regulars, there is a d in the minimum integer format whose $\min d_i$ equals 4, namely 556448, and there are many d 's with all d_i different, including (6,2,4,3,1,8), (4,3,1,6,2,12), (5,3,2,6,4,12), (8,6,2,1,3,12), (8,6,2,4,3,9), and (8,3,1,4,6,10). All of these are type A as well as type B. On the other hand, (7,6,1,2,3,9) is a type B unique solution, according to the five linearly independent type B equations

$$d_1 = d_2 + d_3,$$

$$d_1 + d_2 + d_3 = d_4 + d_5 + d_6,$$

$$d_2 = d_3 + d_4 + d_5,$$

$$d_2 + d_3 + d_4 = d_6,$$

$$d_3 + d_4 = d_5.$$

However, it is not a type A unique d . The fourth equation is not an admissible type A equation, and we can show that there are no five linearly independent type A equations that define d .

4. Counting unique solutions. Let α_n , β_n , and γ_n be the number of *regular* d 's of length n for types A, B, and C, respectively, and let α_n^* , β_n^* , and γ_n^* be the corresponding number of *irregular* d 's. Table 2 shows these counts for small n . The numbers for $n \leq 5$ come from the lists of the preceding section and from [8]. The counts of type A and B regulars for $n = 6$ (626 and 2433) were obtained by allowable left- and right-end additions to the regulars for $n = 5$, and the partial irregular counts for α_6^* and β_6^* were generated largely by what we call "insertions" and "splits" on the d 's for $n = 5$, along with "end additions." For example, $(1,1,2,7,4,3)$ can be obtained for type B from 11243 at $n = 5$ by inserting $7 (=4 + 3)$ between 112 and 43, but because insertion separates 4 from 112, $(1,1,2,7,4,3)$ is not of type A. Splits are illustrated by first doubling all d_i for an $n = 5$ solution and then splitting one of the numbers in two:

even split: $41326 \rightarrow (8,2,6,4,12) \rightarrow (8,2,3,3,4,12)$,

odd split: $41326 \rightarrow (8,2,6,4,12) \rightarrow (8,2,6,4,1,11)$.

An odd split requires one new nontrivial equation, such as $d_6 = d_3 + d_4 + d_5$ ($11 = 6 + 4 + 1$) in the latter case. An end addition just means adding at the end an appropriate sum of d_i 's appearing earlier.

Table 2 shows that even for small n there can be a rich variety of unique d 's, and that we do not have to go very far before the irregulars overtake the regulars. This happens first at $n = 4$ for type C and at $n = 6$ for the others.

We now turn to counts for all n , beginning with α_n for type A regulars.

THEOREM 1. *With $\alpha_1 = \alpha_2 = 1$, for all $n \geq 2$,*

$$\alpha_{n+1} = 2n\alpha_n - (n-1)^2\alpha_{n-1},$$

$$\alpha_n \sim (n-1)!k_1\sqrt{e/\pi}e^{2\sqrt{n}}/(2n^{1/4}),$$

where

$$k_1 = \frac{1}{e} - \int_{y=0}^1 \frac{e^{-1/(1-y)}}{1-y} dy = 0.148495 \dots$$

Remarks. A complete proof of Theorem 1 is given in [9]. The recurrence holds because each d for α_n has n different left-end additions and n different right-end additions, and that additions at both ends of the d 's for α_{n-1} precisely cover all duplications of separate left and right additions for α_n . The asymptotic result comes from the recurrence by use of the saddle point method.

TABLE 2
Counts for small n .

n	Type A			Type B			Type C		
	α_n	α_n^*	Total	β_n	β_n^*	Total	γ_n	γ_n^*	Total
2	1	0	1	1	0	1	1	0	1
3	3	0	3	3	0	3	3	1	4
4	14	2	16	18	2	20	18	57	75
5	85	63	148	172	94	266	185	7231	7416
6	626	≥ 1311	≥ 1937	2433	≥ 2862	≥ 5295			

Although regular counts for types B and C are more complex, we do know their general behaviors.

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

$$\alpha_n = n^{n(1+o(1))}, \quad \beta_n = n^{2n(1+o(1))}, \quad \gamma_n = 2^{(n^2/2)(1+o(1))}.$$

Proof. First, we note that $\alpha_n \leq n^{n-2}$. This follows for $n \leq 5$ by enumeration (see Table 1) and for $n > 5$ by using induction on n and the observation that $\alpha_{n+1} \leq 2n\alpha_n$. The latter holds since there are at most n possible additions to the right ends of type A regulars of length n , and similarly for left ends. (It also follows by Theorem 1.) Next, we shall show that $\alpha_n \geq (n-1)!$. For by the recurrence in Theorem 1, it is easy to show by induction that $\alpha_{n+1}/\alpha_n > n^2/(n+1)$. But then by the recurrence in Theorem 1,

$$\alpha_{n+1} = n\alpha_n + [n\alpha_n - (n-1)^2\alpha_{n-1}] > n\alpha_n.$$

It now follows by induction on n that $\alpha_n \geq (n-1)!$. Finally, since $(n-1)! = n^{n(1+o(1))}$, the desired conclusion for α_n follows.

Let δ_n be the number of *nondecreasing* type C regulars at n . It is known [8] that $\delta_n = 2^{(n^2/2)(1+o(1))}$. Since $\delta_n \leq \gamma_n \leq n!\delta_n$ and

$$n! < n^n = 2^{n \log_2 n} = 2^{(n^2/2)o(1)},$$

the stated result for γ_n follows.

We conclude with type B regulars. Consider a type B regular that has $d_1 = d_2 = 1$ and is constructed left to right. Then,

$$d_k = d_i + d_{i+1} + \cdots + d_j,$$

for $1 \leq i \leq j \leq k-1$. Hence, the number of different d_k is at most the number of such i, j , i.e.,

$$k-1 + \binom{k-1}{2} = \frac{k(k-1)}{2}.$$

It follows that the number of such regulars is less than or equal to

$$\prod_{k=3}^n (k(k-1)/2) \leq n[(n-1)!]^2/2^{n-2}.$$

Now an arbitrary type B regular d can be constructed outwardly from a pair of 1's located in any one of $n-1$ adjacent positions, and there are no more than 2^{n-2} left/right sequences outward from (1,1) to construct d . Moreover, for a given choice of location for (1,1) and of a sequence of moves outward, the number of d 's is at most $n[(n-1)!]^2/2^{n-2}$, by an argument analogous to the one above. It follows that

$$\beta_n \leq (n-1)2^{n-2} \{n[(n-1)!]^2/2^{n-2}\} \leq (n!)^2 < n^{2n} = n^{2n(1+o(1))}.$$

To obtain a lower bound on β_n , begin with $d_1 = d_2 = 1$ and continue through position $k+1$, adding a maximum possible d_i in each place, and getting $11248 \cdots 2^{k-1}$. It is easily checked that there are exactly $1 + 1 + 2 + 3 + \cdots + k = (k^2 + k + 2)/2 > k^2/2$ different sums produced by contiguous strings $d_i + d_{i+1} + \cdots + d_j$ for $1 \leq i \leq j \leq k+1$. Then, using only these sums in positions $k+2$ through n , we conclude that

$$\beta_n \geq (k^2/2)^{n-k-1} = (2/k^2)(k/\sqrt{2})^{2(n-k)}.$$

Let $k \approx n^{1-1/(\log \log n)}$, where $\approx$ means approximately equal. Then

$$\begin{aligned} \log (k/\sqrt{2})^{2(n-k)} &\approx 2(n - n^{1-1/(\log \log n)}) \left[\log (n^{1-1/(\log \log n)}) - \frac{1}{2} \log 2 \right] \\ &= 2n(\log n) \left\{ 1 - \frac{1}{\log \log n} - \frac{n^{1-1/(\log \log n)}}{n} + \frac{n^{1-1/(\log \log n)}}{n \log \log n} \right. \\ &\quad \left. - \frac{\log \sqrt{2}}{\log n} + \frac{(\log \sqrt{2})n^{1-1/(\log \log n)}}{n \log n} \right\} \\ &= 2n(\log n)[1 + o(1)] \end{aligned}$$

since $n^{1-1/(\log \log n)}/n \rightarrow 0$. Hence

$$\begin{aligned} \log \beta_n &\geq \log \left[\left(\frac{2}{k^2} \right) \left(\frac{k}{\sqrt{2}} \right)^{2(n-k)} \right] \\ &\approx 2n(\log n)[1 + o(1)] + \log 2 - 2 \left(1 - \frac{1}{\log \log n} \right) \log n \\ &\geq 2n(\log n) \left[1 + o(1) - \left(1 - \frac{1}{\log \log n} \right) / n \right] \\ &= 2n(\log n)[1 + o(1)]. \end{aligned}$$

It follows that $\beta_n \geq n^{2n(1+o(1))}$. This and the upper bound derived above yield the desired result. $\square$

Upper bounds on irregular counts are noted next. The bounds in the following theorem hold for irregulars plus regulars, but we omit the regular parts since they are orders of magnitude less than the new bounds.

THEOREM 3. For all $n \geq 2$,

$$\alpha_n^* \leq n^{2n(1+o(1))}, \quad \beta_n^* \leq n^{3n(1+o(1))}, \quad \gamma_n^* \leq 3^{n^2(1+o(1))}.$$

Proof. As described in the definition of type A, each type A equation corresponds to a triple (i, j, k) with $1 \leq i < j < k \leq n+1$. Hence there are $\binom{n+1}{3} = (n^3 - n)/6$ type A equations. Since Lemma 1 tells us that $n-1$ of these are needed to obtain a unique d ,

$$\alpha_n + \alpha_n^* \leq \binom{(n^3 - n)/6}{n-1} \leq \frac{(n^3/6)^{n-1}}{(n-1)!} = \frac{n^{3n(1+o(1))}}{n^{n(1+o(1))}} = n^{2n(1+o(1))}.$$

For type B there are $\binom{n+1}{3}$ of the type A equations plus another $\binom{n+1}{4}$ for $1 \leq i < j < k < l \leq n+1$, for a total of $\binom{n+2}{4}$ type B equations. Using Lemma 1, we find that

$$\beta_n + \beta_n^* \leq \binom{\binom{n+2}{4}}{n-1} \leq \frac{n^{4n(1+o(1))}}{n^{n(1+o(1))}} = n^{3n(1+o(1))}.$$

Each type C equation can be written as $\sum x_i d_i = 0$, with $x_i = 0, +1$, or -1 . There are at most 3^n such equations. But multiplying all x_i by -1 gives an equivalent equation, so there are at most $3^n/2$ type C equations. It follows that

$$\gamma_n + \gamma_n^* \leq \binom{3^n/2}{n-1} \leq \frac{(3^n/2)^{n-1}}{(n-1)!} = 3^{n^2(1+o(1))}. \quad \square$$

The final section of [8] highlights difficulties of precise assessment of γ_n^* in the probability setting. Even if we could estimate the number of $(n - 1)$ -member subsets of equations of a given type that are linearly independent *and* have positive solutions, we would still face the problem that many different subsets can have the same solution. Despite this, Theorems 2 and 3 do tell us that $\gamma_n/(\beta_n + \beta_n^*) \rightarrow \infty$.

We have noted that the irregulars overtake the regulars at small n for each type. We now prove that α_n^*/α_n and γ_n^*/γ_n become arbitrarily large as n gets large. We lack a proof of this for β_n^*/β_n . We have not even verified that $\beta_n^* > \beta_n$ for large n , although this is surely true.

THEOREM 4. $\alpha_n^*/\alpha_n \rightarrow \infty$ and $\gamma_n^*/\gamma_n \rightarrow \infty$.

Proof for Type A. The results of Theorem 1 will be combined with the claim that

$$\alpha_{n+2}^* \geq \left[\left(\frac{17}{16} \right) n^2 - n \right] \alpha_n^*$$

for all $n \geq 6$. To verify the claim for n even, we consider double left extensions of each irregular d counted in α_n^* : $yx d_1 d_2 \cdots d_n$. There are n regular extensions for x ($x = d_1, x = d_1 + d_2, \dots, x = d_1 + \cdots + d_n$), but we count only the last $n - 1$ of these to avoid turning d into a regular solution via $x = d_1 = 1$. Then there are $n + 1$ regular extensions for y ($y = x, \dots, y = x + d_1 + \cdots + d_n$). Hence, there are at least $(n - 1)(n + 1)\alpha_n^* = (n^2 - 1)\alpha_n^*$ regular-regular extensions for xy that guarantee type A irregulars at $n + 2$. All of these have $y \geq x$.

There are also regular-irregular extensions where x is taken as before with $x = d_1 + \cdots + d_j$ and y is defined by the type A equation

$$x + y = (d_1 + \cdots + d_j) + (d_{j+1} + \cdots + d_k),$$

so that $y = d_{j+1} + \cdots + d_k$. To avoid duplication with extensions in the preceding paragraph, we use these only when $y < x$, i.e., when $d_{j+1} + \cdots + d_k < d_1 + \cdots + d_j$. By duality, at least half of the d 's counted in α_n^* have the property that

$$d_1 + \cdots + d_{(n/2)+1} > d_{(n/2)+2} + \cdots + d_n.$$

For each of these, y can be defined in $n/2 - 1$ ways to get $y < x$ by the preceding equation when $j = n/2 + 1$ and $k \leq n$; y can be defined in $n/2 - 2$ ways for $y < x$ when $j = n/2 + 2$ and $k \leq n$; and so forth. Hence, the total number of irregulars at $n + 2$ produced by this regular-irregular extension procedure, all of which have $y < x$, is at least

$$\left(\frac{\alpha_n^*}{2} \right) \left[\left(\frac{n}{2} - 1 \right) + \left(\frac{n}{2} - 2 \right) + \cdots + 1 \right] = \alpha_n^* n(n - 2)/16.$$

The double left extensions of the preceding two paragraphs guarantee

$$\alpha_{n+2}^* \geq \alpha_n^* [n^2 - 1 + n(n - 2)/16] \geq \left[\left(\frac{17}{16} \right) n^2 - n \right] \alpha_n^*,$$

which proves the claim for n even. When n is odd, the argument is similar. One shows that at least half of the d 's counted in α_n^* have

$$d_1 + \cdots + d_{\lfloor n/2 \rfloor} + d_{\lfloor n/2 \rfloor + 1} > d_{\lfloor n/2 \rfloor + 2} + \cdots + d_n.$$

For each of these, y can be defined in $\lfloor n/2 \rfloor - 1$ ways to get $y < x$, using $j = \lfloor n/2 \rfloor + 1$ and $k < n$; y can be defined in $\lfloor n/2 \rfloor - 2$ ways to get $y < x$, using $j = \lfloor n/2 \rfloor + 2$ and $k <$

n and so forth. Thus, one obtains at least

$$\left(\frac{\alpha_n^*}{2}\right) \left[\left(\left\lfloor \frac{n}{2} \right\rfloor - 1 \right) + \left(\left\lfloor \frac{n}{2} \right\rfloor - 2 \right) + \cdots + 1 \right] = \alpha_n^* (n-1)(n-3)/16$$

irregulars of length $n+2$ using the regular-irregular procedure, and all have $y < x$. It follows that

$$\alpha_{n+2}^* \geq \alpha_n^* \left[n^2 - 1 + \frac{(n-1)(n-3)}{16} \right] \geq \left[\left(\frac{17}{16} \right) n^2 - n \right] \alpha_n^*.$$

This proves the claim.

From the recurrence in Theorem 1, it is straightforward to show by induction that $\alpha_{n+1}/\alpha_n < (\sqrt{n} + 1)^2$. Hence,

$$\begin{aligned} \alpha_{n+2} &< \alpha_n (\sqrt{n} + 1)^2 (\sqrt{n+1} + 1)^2 \\ &= \alpha_n [n^2 + o(n^2)]. \end{aligned}$$

Since $[(\frac{17}{16})n^2 - n]/[n^2 + o(n^2)] \rightarrow \frac{17}{16}$, it follows that for all large n ,

$$\alpha_{n+2}^*/\alpha_n^* \geq (1.05)\alpha_{n+2}/\alpha_n.$$

Hence, with large n fixed,

$$\frac{\alpha_{n+2N}^*}{\alpha_{n+2N}} \geq (1.05)^N \frac{\alpha_n^*}{\alpha_n} \rightarrow \infty \quad \text{as } N \rightarrow \infty. \quad \square$$

Proof for Type C. For $n \geq 5$, recall that δ_n is the number of *nondecreasing* regular type C d 's with n terms. Each of these begins with a pair of 1's. As noted in the Appendix of [8], $\delta_5 = 27$. Let δ_n^* be the number of *strictly increasing* irregular type C d 's with n terms that begin with irregular $(1,2,3,6,10)$ at $n=5$ and proceed by *regular* extension to the right. It follows from the fact that every integer from 1 to 22 inclusive is the sum of one or more integers in $\{1,2,3,6,10\}$ that the integer sequences counted in δ_n^* are precisely identified by the inequalities $11 \leq d_6 \leq 22$, $d_6 + 1 \leq d_7 \leq d_6 + 22$, $d_7 + 1 \leq d_8 \leq d_6 + d_7 + 22$, $\dots$, $d_{n-1} + 1 \leq d_n \leq d_6 + \dots + d_{n-1} + 22$.

Clearly, $\gamma_n \leq n!\delta_n$ and, because the sequences counted in δ_n^* are strictly increasing, $\gamma_n^* \geq n!\delta_n^*$. Therefore $\gamma_n^*/\gamma_n \geq \delta_n^*/\delta_n$. We claim that $\delta_n^*/\delta_n \rightarrow \infty$, so $\gamma_n^*/\gamma_n \rightarrow \infty$ also.

To verify the claim, consider the nondecreasing regular right extensions of $(1,1,2,4,8)$, the regular type C at $n=5$ with maximum sum, and compare the increasing regular right extensions of $(1,2,3,6,10)$. The number of such extensions of $(1,1,2,4,8)$ through n is

$$\begin{aligned} \sum_{d_6=8}^{16} \sum_{d_7=d_6}^{d_6+16} \cdots \sum_{d_n=d_{n-1}}^{\sum_{i=6}^{n-1} d_i + 16} 1 &= \sum_{x_6=0}^8 \sum_{x_7=x_6}^{x_6+16} \cdots \sum_{x_n=x_{n-1}}^{\sum_{i=6}^{n-1} x_i + 16} 1 \\ &\leq \int_{x_6=0}^9 \int_{x_7=x_6}^{x_6+17} \cdots \int_{x_n=x_{n-1}}^{\sum_{i=6}^{n-1} x_i + 17} dx_n \cdots dx_7 dx_6 \\ &= A. \end{aligned}$$

The number for (1,2,3,6,10) is

$$\begin{aligned}
 \delta_n^* &= \sum_{d_6=11}^{22} \sum_{d_7=d_6+1}^{d_6+22} \cdots \sum_{d_n=d_{n-1}+1}^{\sum_{i=6}^{n-1} d_i + 22} 1 \\
 &\geq \sum_{d_6=11}^{22} \sum_{d_7=d_6}^{d_6+21} \cdots \sum_{d'_n=d'_{n-1}}^{d_6+d'_7+\cdots+d'_{n-1}+21} 1 \\
 &= \sum_{y_6=0}^{11} \sum_{y_7=y_6}^{y_6+21} \cdots \sum_{y_n=y_{n-1}}^{\sum_{i=6}^{n-1} y_i + 21} 1 \\
 &\geq \int_{y_6=0}^{11} \int_{y_7=y_6}^{y_6+21} \cdots \int_{y_n=y_{n-1}}^{\sum_{i=6}^{n-1} y_i + 21} dy_n \cdots dy_7 dy_6 \\
 &= B,
 \end{aligned}$$

where $d'_i = d_i - (i - 6)$ for $i \geq 7$. It then follows from the analysis in the proof of Theorem 2 in [8] that the ratio B/A goes to ∞ as n increases. The same thing is true for each of the other 26 nondecreasing regulars for δ_5 versus (1,2,3,6,10), but even more so since the sum of the five terms in each is less than 16. Finally, since δ_n is the sum of the numbers of nondecreasing regular right extensions through n of the 27 d 's in δ_5 , $\delta_n^*/\delta_n \rightarrow \infty$. $\square$

5. The appearance of specific terms in regular solutions. We turn next to the appearance of specified positive integers in *regular* type A and type B d 's. We shall assume that $\min d_i = 1$ in all cases. We shall also assume throughout this section that $m, k, k_1, k_2, \dots$ are positive integers.

DEFINITION. $r(k_1, \dots, k_m)$ is the length of a *shortest* type A regular d that contains some permutation of $k_1, \dots, k_m$. $s(k_1, \dots, k_m)$ is the length of a *shortest* type B regular d that contains some permutation of $k_1, \dots, k_m$. If there is no regular d of the given type that contains a permutation of $k_1, \dots, k_m$, then $r(k_1, \dots, k_m)$ or $s(k_1, \dots, k_m)$ is undefined.

We might also consider shortest d 's that contain $k_1, \dots, k_m$ in the given order as a subsequence, but we will not do that here.

The following results, which apply exclusively to type A, are proved in [9].

THEOREM 5. $r(k_1, \dots, k_m)$ is always defined for $m \leq 4$ but can be undefined for $m \geq 5$.

THEOREM 6. $r(k) = \lceil \log_2 k \rceil + 2$ for all $k \geq 2$.

LEMMA 2. $r(k, k+1, k+2, k+3) \geq \lfloor k/3 \rfloor + 6$ for all $k \geq 4$.

THEOREM 7. If $k_1 \leq k_2 \leq k_3 \leq k_4$, then $r(k_1, k_2, k_3, k_4) - r(k_2, k_3, k_4)$ can be arbitrarily large.

We comment briefly on the proofs of these results before proving results that involve type B. To prove Theorem 5, we note that given $k_1 \leq k_2 \leq k_3 \leq k_4$,

$$k_2, k_1, 1, \dots, 1, k_3, k_4$$

with k_4 1's is a type A regular. Then we note that there is no type A regular that contains a permutation of 4,5,6,7,8. A similar construction, but with terms larger than 1 in the middle, is used to prove Lemma 2. Next, one observes that

$$2^t + 2, 2^t + 1, 1, 1, 1, 2, 4, 8, \dots, 2^t$$

is a type A regular that contains $k + 1$, $k + 2$, and $k + 3$ when $k + 1 = 2^t$. This fact and Lemma 2 imply that

$$r(k, k + 1, k + 2, k + 3) - r(k + 1, k + 2, k + 3)$$

can be arbitrarily large, which is a summary of the proof of Theorem 7.

As for Theorem 6, one shows that for $k \geq 2$, the largest possible term in a type A regular solution of length n is 2^{n-2} , which shows that $r(k) \geq \lceil \log_2 k \rceil + 2$. The other inequality, i.e., $r(k) \leq \lceil \log_2 k \rceil + 2$, follows when k is not a power of 2 from the type A regular

$$k, 2^{v_1}, \dots, 2^{v_2}, 2^{v_1}, 1, 2^{u_1}, 2^{u_2}, \dots, 2^{u_q},$$

where

$$k = 2^{v_1} + 2^{v_2} + \dots + 2^{v_t}, \quad 0 \leq v_1 < \dots < v_t$$

and

$$\{u_1, \dots, u_q\} = \{0, 1, \dots, v_1\} \setminus \{v_1, \dots, v_t\},$$

with $u_1 < \dots < u_q$. The desired inequality follows when k is a power of 2 by using the preceding sequence from 2^{v_1} on. Of course, $1, 1, 2, 4, \dots, 2^{v_1}$ would be just as good.

The following for type B correspond to Theorems 5 and 6.

THEOREM 8. $s(k_1, \dots, k_m)$ is always defined.

THEOREM 9. $s(k) = \lceil \log_2 k \rceil + 2$ for all $k \geq 2$.

Proofs. Let $k = \max k_i$. Then $1, \dots, 1, k_1, k_2, \dots, k_m$ with k 1's is a type B regular. Theorem 9 follows from Theorem 6 because $s(k) \geq \lceil \log_2 k \rceil + 2$ as in the proof of Theorem 6 and because $s(k) \leq r(k)$ (every type A regular is a type B regular). $\square$

We do not have a type B correspondent to Theorem 7, and presently do not know whether $s(k_1, k_2, k_3, k_4) - s(k_2, k_3, k_4)$ can exceed 1 when $k_1 \leq k_2 \leq k_3 \leq k_4$. We do, however, have the following comparison between types A and B.

THEOREM 10. $r(k_1, k_2, k_3, k_4) - s(k_1, k_2, k_3, k_4)$ can be arbitrarily large.

Proof. We show that

$$s(k, k + 1, k + 2, k + 3) \leq \lceil \log_2 k \rceil + 7,$$

which in conjunction with Lemma 2 proves that

$$r(k, k + 1, k + 2, k + 3) - s(k, k + 1, k + 2, k + 3)$$

can be arbitrarily large. To prove the desired inequality, let $S(k)$ denote the type A regular used in the proofs of Theorems 6 and 9: $(k, 2^{v_1}, \dots, 2^{u_q})$ or $(2^{v_1}, 1, 1, 2, \dots, 2^{v_1-1})$ when $k = 2^{v_1}$. Then $[k + 3, 1, k + 2, k + 1, 1, S(k)]$ is easily seen to be a type B regular with $5 + \lceil \log_2 k \rceil + 2$ terms. $\square$

Although we have been unable to match the conclusion of Theorem 10 for k_1, k_2, k_3 , we do know that $r(k_1, k_2, k_3)$ can exceed $s(k_1, k_2, k_3)$.

THEOREM 11. $r(4, 5, 6) > s(4, 5, 6)$.

Proof. Since $5, 4, 1, 1, 2, 6$ is a type B regular, $s(4, 5, 6) \leq 6$. However, the same sequence is not a type A regular and no type A regular with six terms contains all of 4, 5, and 6. If it did, then its three smallest terms would have to be 1, 1, 2 in that order or its dual as 1, 1, 1 is insufficient for 4. But then 4, 5, and 6 cannot be added to get a type A regular. $\square$

Remark. By a similar proof, $r(2, 4, 5, 6) > s(2, 4, 5, 6)$, $r(1, 2, 4, 5, 6) > s(1, 2, 4, 5, 6)$, and $r(1, 1, 2, 4, 5, 6, \dots, 6) > s(1, 1, 2, 4, 5, 6, \dots, 6)$, so we can have $r(k_1, \dots, k_m) > s(k_1, \dots, k_m)$ for all $m \geq 3$.

Among the things we do not know about r and s , the following seem attractive for further study.

CONJECTURE 1. $r(k_1, k_2) = s(k_1, k_2)$.

CONJECTURE 2. If $k_1 \leq k_2$, then $r(k_1, k_2) \leq r(k_2) + 1$.

CONJECTURE 3. If $k_1 \leq k_2$, then $s(k_1, k_2) \leq s(k_2) + 1$.

Since $s(k_1, k_2) \leq r(k_1, k_2)$ (every type A regular is a type B regular) and $r(k_2) = s(k_2)$ (Theorems 6 and 9), Conjecture 2 implies Conjecture 3.

All three conjectures have been systematically verified for $k_1 \leq k_2 \leq 16$. In addition, we have proved several special cases of Conjecture 3 that we note here without proof. Let

$$P(k) = \{v_1, v_2, \dots, v_{t(k)}\}$$

when

$$k = 2^{v_1} + \dots + 2^{v_{t(k)}}, \quad 0 \leq v_1 < \dots < v_{t(k)},$$

and call $P(k)$ *consecutive* when $v_{i+1} = v_i + 1$ for $i = 1, \dots, t(k) - 1$. Then, given $k_1 < k_2$ and $l = k_2 - k_1$, the conclusion of Conjecture 3 holds if any of the following hold:

- (a) $P(k_1)$ is consecutive, $P(k_1) \subseteq P(k_2)$;
- (b) $P(k_1)$ is consecutive, $P(k_1) \subseteq P(l)$;
- (c) $P(k_1)$ is consecutive, $P(k_1) \cap P(k_2) = \emptyset$;
- (d) $P(k_1)$ is consecutive, $P(k_1) \cap P(l) = \emptyset$, $k_1 < l$;
- (e) $P(k_1)$ and $P(l)$ are consecutive, $P(k_1) \cap P(l) = \emptyset$, $l < k_1$;
- (f) $P(k_2)$ is consecutive, $P(k_1) \cap P(k_2) = \emptyset$;
- (g) $P(l)$ is consecutive, $P(l) \subseteq P(k_1)$, $v_{t(l)} = v_{t(k_1)}$;
- (h) l is a power of 2, $k_1 \leq l$.

Thus, if there is a counterexample to Conjecture 3, it must violate all of (a)–(h) and have $k_2 \geq 17$.

6. Three extremization problems. Theorem 3 in [8] says that there are irregular d 's of n terms of type C such that, for large n , $(\min d_i / \sum d_j)$ can be an arbitrarily small fraction of $(\frac{1}{2})^{n-1}$, the smallest such ratio for type C regulars. However, we have been unable to show that $(\min d_i / \sum d_j)$ can be less than $(\frac{1}{2})^{n-1}$ for type A and type B irregulars of n terms. Therefore, we state two more conjectures.

CONJECTURE 4. $(\min d_i / \sum d_j) \geq (\frac{1}{2})^{n-1}$ for all type A unique d 's with n terms.

CONJECTURE 5. $(\min d_i / \sum d_j) \geq (\frac{1}{2})^{n-1}$ for all type B unique d 's with n terms.

Table 1 shows that the conjectures hold at $n = 5$, and, of course, we have not found a counterexample among the several thousand d 's at $n = 6$ counted for Table 2 or thousands more generated by computer. Equivalent versions of the conjectures are noted in the next section.

The second extremization problem for irregulars that was mentioned in § 1 is to maximize $(\min d_i / \min d_j)$ over all unique d 's of a given type that have n terms. Let $M_A(n)$, $M_B(n)$, and $M_C(n)$ denote the maxima of $\min/\min$ at n for types A, B, and C, respectively.

THEOREM 12.

$$M_A(n) \geq 2^{\lfloor n/2 - 1 \rfloor} \quad \text{for all } n \geq 2,$$

$$M_B(n) \geq 2^{\lfloor n/2 - 1 \rfloor} \quad \text{for all } n \geq 2.$$

THEOREM 13. $M_C(n) \geq 2^{n-4} + 2^{n-6} + 2^{(n-4)/2}$ for all $n \geq 5$.

Proofs. Theorem 13 is equivalent to Theorem 4 in [8]. For Theorem 12 it suffices to construct a type A unique d for even $n \geq 2$ for which $\min/\min = 2^{n/2-1}$. The result is

trivial for $n = 2$. Let $n = 2m$ and for $m \geq 2$ consider the n -term sequence

$$2^{m-2}x, 2^{m-3}x, \dots, 4x, 2x, x, x, 1, x+1, 2x+2, 4x+4, \dots, 2^{m-2}x+2^{m-2}.$$

Apart from indeterminate x , this is a type A sequence built regularly inside-out away from $x, x, 1$. Our final type A equation that is used to determine $x = x/1 = \text{pin/min}$ equates the sum of all left terms through x, x to the sum of all right terms, starting at 1. The equation is

$$2^{m-1}x = (2^{m-1} - 1)x + 2^{m-1} \quad \text{or} \quad x = 2^{m-1} = 2^{n/2-1}. \quad \square$$

Our third extremization problem is to maximize $d_j/\min d_i$ for each $j \leq \lceil n/2 \rceil$. Let $N_A(j, n)$ and $N_B(j, n)$ be the maxima of $d_j/\min d_i$ over all n -term unique d 's for types A and B, respectively. By Table 1,

$$N_A(1, 5) = N_B(1, 5) = 8,$$

$$N_A(2, 5) = N_B(2, 5) = 4,$$

$$N_A(3, 5) = N_B(3, 5) = 4.$$

The type B irregular $(2, 10, 1, 2, 3, 6)$ shows that $N_B(2, 6) \geq 10$, but it appears impossible to have $N_A(2, 6)$ larger than 8.

THEOREM 14. For all $n \geq 6$,

$$N_A(1, n) \geq 2^{n-2}, \quad N_B(1, n) \geq 2^{n-2},$$

$$N_A(j, n) \geq 2^{n-3} \quad \text{for } 2 \leq j \leq \lceil n/2 \rceil,$$

$$N_B(j, n) \geq 2^{n-3} + 2^{n-j-2} - 2 \quad \text{for } 2 \leq j \leq \lceil n/2 \rceil.$$

Proof for Type A. The bounds for $j = 1$ or 2 come from $2^{n-2}, 2^{n-3}, \dots, 2, 1, 1$. For $j = 3$, we use

$$2^{n-2} - 1, 1, 2^{n-3}, 2^{n-4}, \dots, 4, 2, 2,$$

which is an n -term type A irregular. Two of its equations are $d_1 = d_2 + d_3 + \dots + d_{n-1}$ and $d_1 + d_2 = d_3 + \dots + d_n$, and it verifies $N_A(3, n) \geq 2^{n-3}$. To get $N_A(4, n) \geq 2^{n-3}$, we double all terms in the preceding sequence and then split $d_2 = 2$ into $d'_2 = d'_3 = 1$ to get the $(n+1)$ -term sequence

$$2^{n-1} - 2, 1, 1, 2^{n-2}, 2^{n-3}, \dots, 8, 4, 4,$$

so $N_A(4, n+1) \geq 2^{n-2}$, or $N_A(4, m) \geq 2^{m-3}$ with $m = n+1$. Since the preceding step for $j = 3$ presumes $3 \leq \lceil n/2 \rceil$, our procedure gives $N_A(4, m) \geq 2^{m-3}$ whenever $3 \leq \lceil (m-1)/2 \rceil$, which includes all m such that $4 \leq \lceil m/2 \rceil$. Hence, $N_A(4, n) \geq 2^{n-3}$ whenever $4 \leq \lceil n/2 \rceil$. Another doubling and even split of one 2 gives $N_A(5, n) \geq 2^{n-3}$, and continuation of the process yields the desired result. $\square$

Proof for Type B. To verify the final line of Theorem 14, consider the n -term sequence

$$y_{j-1}, \dots, y_2, y_1, d_j, 1, z_1, z_2, \dots, z_{n-j-1}.$$

Two type B equations used for this are

$$d_j + 1 = \sum z_i \quad \text{and} \quad y_1 + d_j = 1 + \sum z_i,$$

which imply $y_1 = 2$. We also use

$$y_2 = y_1, \quad y_3 = y_2 + y_1, \quad \dots, \quad y_{j-1} = y_{j-2} + \dots + y_1,$$

$$z_1 = \sum y_i \text{ (the only equation not of type A),}$$

$$z_2 = z_1 + 1, \quad z_3 = z_2 + z_1 + 1, \quad \dots, \quad z_{n-j-1} = z_{n-j-2} + \dots + z_1 + 1.$$

These give $y_i = 2^{i-1}$ for $i \geq 2$, $z_1 = 2^{j-1}$, and

$$z_i = (2^{j-1} + 1)2^{i-2} \quad \text{for } i \geq 2.$$

Hence, $\sum z_i = 2^{n-j-2}(2^{j-1} + 1) - 1$, so by $d_j + 1 = \sum z_i$, we obtain

$$d_j = 2^{n-3} + 2^{n-j-2} - 2. \quad \square$$

As mentioned earlier, all permutations of type C unique d 's are also type C unique d 's. In this case, therefore, we define $N_C(n)$ as the maximum of $\max d_i / \min d_i$ over all n -term unique d 's for type C. It can be seen from [8] that $N_C(4) = 4$, $N_C(5) = 10$, $N_C(6) \geq 28$, $N_C(7) \geq 74$, and $N_C(n) \geq 192$. In general, $N_C(n)$ grows more rapidly than 2^{n-2} .

THEOREM 15. $N_C(n) > (2.12)^{n-2}$ for all $n \geq 5$.

Proof. The first few $n \geq 5$ can be checked by computation. In the proof, we build sequences $d(n) = d_1, \dots, d_n$. The sequences $d(n)$ will have certain values d_i fixed for all n sufficiently large. We use the notation d_i^* for a term that will be fixed for all subsequent n . We start with the following seven-term type C irregular from [8], whose terms remain fixed for all n :

$$d_1^* = 1, \quad d_2^* = 5, \quad d_3^* = 14, \quad d_4^* = 18, \quad d_5^* = 36, \quad d_6^* = 44, \quad d_7^* = 74.$$

For $8 \leq n < 14$, we extend $d(7)$ regularly to the right by summing all previous terms and obtain

$$d_8 = 192, \quad d_9 = 2(192), \quad \dots, \quad d_{13} = 2^5(192).$$

For $n \geq 14$, $d_8, \dots, d_{14}$ become fixed as $d_8^*, \dots, d_{14}^*$, according to the formula $(d_8^*, \dots, d_{14}^*) = (\sum_{i=1}^7 d_i^*)(d_1^*, \dots, d_7^*)$. Thus,

$$d_8^* = 192, \quad d_9^* = 5(192), \quad \dots, \quad d_{14}^* = 74(192).$$

For $15 \leq n < 21$, we extend $d(14)$ regularly to the right by summing all preceding terms and obtain

$$d_{15} = (192)(193), \quad d_{16} = 2(192)(193), \quad \dots, \quad d_{20} = 2^5(192)(193).$$

For $n \geq 21$, $d_{15}, \dots, d_{21}$ become fixed as $d_{15}^*, \dots, d_{21}^*$ according to the formula $(d_{15}^*, \dots, d_{21}^*) = (\sum_{i=1}^{14} d_i^*)(d_1^*, \dots, d_{14}^*)$. Thus,

$$d_{15}^* = (192)(193), \quad d_{16}^* = 5(192)(193), \quad \dots, \quad d_{21}^* = 74(192)(193).$$

For $22 \leq n < 28$, we extend $d(21)$ regularly to the right by summing all preceding terms and obtain

$$d_{22} = (192)(193) + (192)(192)(193) = (192)(193)^2,$$

$$d_{23} = 2(192)(193)^2, \quad \dots, \quad d_{27} = 2^5(192)(193)^2.$$

And so on.

Note that if $n = 7k + i + 1$, $i = 0, 1, \dots, 5$, then for $d(n)$,

$$\max d_i = d_n = 2^i(192)(193)^{k-1}.$$

If $n = 7k + 7$, then for $d(n)$,

$$\max d_i = d_n = 74(192)(193)^{k-1}.$$

Thus, to show that $N_C(n) > (2.12)^{n-2}$ for all $n \geq 8$, it suffices to show for all $k \geq 1$ that

$$2^i(192)(193)^{k-1} > (2.12)^{7k+i-1} \quad \text{for } i = 0, 1, \dots, 5,$$

and

$$74(192)(193)^{k-1} > (2.12)^{7k+5}.$$

Taking natural logarithms, we see that each of these has the form

$$c + (0.002778)k > 0$$

with $c > 0$; therefore they always hold. $\square$

7. The max conjecture. Our inability to understand some of the finer structure of type A and B unique d 's is most clearly illustrated by Conjectures 4 and 5. In the interest of encouraging others to work on them, we note some equivalents. For convenience here we assume that $\min d_i = 1$ for every d .

THEOREM 16. *The following three assertions are mutually equivalent for either type A or type B with $\min d_i = 1$:*

$$(A1) \quad \sum d_i \leq 2^{n-1} \text{ for all } n \geq 3 \text{ and all unique } d.$$

$$(A2) \quad \max d_i \leq 2^{n-2} \text{ for all } n \geq 3 \text{ and all unique } d.$$

$$(A3) \quad \max d_i < 2^{n-2} \text{ for all } n \geq 3 \text{ and all unique } d \text{ for which } \max d_i \text{ is not the sum of all other } d_i.$$

Proof. If (A1) fails at n with $\sum d_i > 2^{n-1}$, then (A2) fails at $n + 1$ when we extend d by taking $d_{n+1} = d_1 + \dots + d_n$. If (A2) fails at n with $\max d_i > 2^{n-2}$, then (A1) fails at n since $\max d_i$ can never exceed the sum of all other d_i . Hence (A1) and (A2) are equivalent.

If (A1) or (A2) fails for d at n , continue d regularly with

$$d_{n+1} = d_1 + \dots + d_n \geq 2^{n-1} + 1, \dots, d_{n+m} = d_1 + \dots + d_{n+m-1} \geq 2^{n+m-2} + 2^{m-1}$$

until $2^m > d_1$. Then take

$$d_{n+m+1} = d_2 + \dots + d_{n+m} \geq 2^{n+m-1} + 2^m - d_1 > 2^{n+m-1}.$$

This violates (A3) at $n + m + 1$. Conversely, if (A3) fails at n with $\max d_i \geq 2^{n-2}$, then the sum of all other d_i exceeds 2^{n-2} , so $\sum d_i > 2^{n-1}$ and (A1) fails. $\square$

In connection with (A1) and (A3) it may also be noted that there are many cases for which $\sum d_i = 2^{n-1}$ and $\max d_i$ is less than 2^{n-2} . Examples include $(2, 2, 1, 3)$, $(4, 1, 3, 2, 6)$, and $(8, 3, 1, 4, 6, 10)$, each of which is a type A unique d .

8. Discussion. Although measurement theory has become a ripe field for interesting mathematical analysis, very little consideration has been given to uniqueness in finite measurement structures. The present paper along with [8], [9] attempts to correct this imbalance and, in the process, to reveal that uniqueness in finite structures is a rich

source of intriguing combinatorial and number-theoretic problems. The results in [8], [9], and related papers are summarized in the survey [10]. Another recent paper that explores other aspects of the uniqueness of finite measurement structures is [21].

Our research has answered a number of questions about uniqueness in finite difference measurement, but it has also uncovered a host of other problems that remain to be solved. Principal among these is the “max” conjecture of the preceding section, whose resolution could contribute substantially to our understanding of sets of $n - 1$ linearly independent equations of types A and B that lead to uniqueness. We also emphasize Conjectures 1, 2, and 3 since they raise interesting issues about the structures of regular solutions.

Numerous other possibilities for further exploration should be evident. One of these is the type B gap in Theorem 4 (does $\beta_n^*/\beta_n \rightarrow \infty?$); another concerns more precise assessments of the number of irregulars for types A, B, and C. In addition, many of the theorems in the paper might be sharpened by giving more precise bounds for various inequalities or by demonstrating that present bounds cannot be improved.

Acknowledgments. The authors thank Suh-ryung Kim and Barry Tesman for their helpful comments.

REFERENCES

- [1] D. DAVIDSON AND P. SUPPES, *A finitistic axiomatization of subjective probability and utility*, *Econometrica*, 24 (1956), pp. 264–275.
- [2] G. DEBREU, *Stochastic choice and cardinal utility*, *Econometrica*, 26 (1958), pp. 440–444.
- [3] Z. DOMOTOR, *Probabilistic relational structures and their applications*, Tech. Report 144, Inst. for Mathematical Studies in the Social Sciences, Stanford Univ., Stanford, CA, 1969.
- [4] T. L. FINE, *Theories of Probability*, Academic Press, New York, 1973.
- [5] P. C. FISHBURN, *Weak qualitative probability on finite sets*, *Ann. Math. Statist.*, 40 (1969), pp. 2118–2126.
- [6] ———, *Utility Theory for Decision Making*, John Wiley, New York, 1970.
- [7] ———, *The axioms of subjective probability*, *Statist. Sci.*, 1 (1986), pp. 335–345.
- [8] P. C. FISHBURN AND A. M. ODLYZKO, *Unique subjective probability on finite sets*, AT&T Bell Laboratories, Murray Hill, NJ, 1987.
- [9] P. C. FISHBURN, A. M. ODLYZKO, AND F. S. ROBERTS, *Two-sided generalized Fibonacci sequences*, AT&T Bell Laboratories, Murray Hill, NJ, 1988.
- [10] P. C. FISHBURN AND F. S. ROBERTS, *Uniqueness in finite measurement*, in *Applications of Combinatorics and Graph Theory to the Biological and Social Sciences*, F. S. Roberts, ed., Springer-Verlag, New York, to appear.
- [11] R. FRISCH, *Sur un problème d'économie pure*, *Norsk. Math. Forenings Skrif.*, 1 (1926), pp. 1–40.
- [12] O. HÖLDER, *Die Axiome der Quantität und die Lehre vom Mass*, *Ber. Verh. Kgl. Sächsis. Ges. Wiss. Leipzig, Math.-Phys. Classe*, 53 (1901), pp. 1–64.
- [13] C. H. KRAFT, J. W. PRATT, AND A. SEIDENBERG, *Intuitive probability on finite sets*, *Ann. Math. Statist.*, 30 (1959), pp. 408–419.
- [14] D. H. KRANTZ, R. D. LUCE, P. SUPPES, AND A. TVERSKY, *Foundations of Measurement*, Vol. I, Academic Press, New York, 1971.
- [15] W. KRISTOF, *A foundation of interval scale measurement*, *Research Bull. RB 67-22*, Educational Testing Service, Princeton, NJ, 1967.
- [16] R. D. LUCE, *Sufficient conditions for the existence of a finitely additive probability measure*, *Ann. Math. Statist.*, 38 (1967), pp. 780–786.
- [17] L. NARENS, *Abstract Measurement Theory*, MIT Press, Cambridge, MA, 1985.
- [18] J. PFANZAGL, *Theory of Measurement*, John Wiley, New York, 1968.
- [19] F. S. ROBERTS, *Measurement Theory, with Applications to Decisionmaking, Utility, and the Social Sciences*, Addison-Wesley, Reading, MA, 1979.

- [20] ———, *Issues in the theory of uniqueness in measurement*, in *Graphs and Orders*, I. Rival, ed., D. Reidel, Amsterdam, 1985, pp. 415–444.
- [21] F. S. ROBERTS AND Z. ROSENBAUM, *Tight and loose value automorphisms*, *Discrete Appl. Math.*, to appear.
- [22] L. J. SAVAGE, *The Foundations of Statistics*, John Wiley, New York, 1954.
- [23] D. SCOTT, *Measurement models and linear inequalities*, *J. Math. Psych.*, 1 (1964), pp. 233–247.
- [24] D. SCOTT AND P. SUPPES, *Foundational aspects of theories of measurement*, *J. Symbolic Logic*, 23 (1958), pp. 113–128.
- [25] S. SIEGEL, *A method for obtaining an ordered metric scale*, *Psychometrika*, 21 (1956), pp. 207–216.
- [26] P. SUPPES AND M. WINET, *An axiomatization of utility based on the notion of utility differences*, *Management Sci.*, 1 (1955), pp. 259–270.
- [27] P. SUPPES AND J. L. ZINNES, *Basic measurement theory*, in *Handbook of Mathematical Psychology*, R. D. Luce, R. R. Bush, and E. Galanter, eds., Vol. I, John Wiley, New York, 1963, pp. 1–76.
- [28] A. TVERSKY AND D. H. KRANTZ, *The dimensional representation and the metric structure of similarity data*, *J. Math. Psych.*, 7 (1970), pp. 572–596.

SEPARATING PARTITION SYSTEMS AND LOCALLY DIFFERENT SEQUENCES*

JÁNOS KÖRNER† AND GÁBOR SIMONYI‡

Abstract. The problem of perfect hashing is generalized and some initial results are obtained. As a corollary, an improvement on earlier results for (i, j) -separating systems of partitions is provided.

Key words. separating partition systems, code distance

AMS(MOS) subject classifications. primary 05B40; secondary 05A15, 94A10

1. Separating partition systems. Separating partition systems have been studied under different names by several authors. Our attention was drawn to this subject by a very stimulating paper by Fredman and Komlós [1] who used information theory to derive nonexistence bounds for separating partition systems in two special cases: systems of perfect hash functions and (i, j) -separating systems. For the latter model, stronger results exist in Djakov and Rykov [2], Erdős, Frankl, and Füredi [3], and Hwang and Sós [4] in the case $i = 1$. However, if $i > 1, j > 1$, no appropriate method seems to be available. By establishing an interesting connection between this problem and a new generalization of perfect hashing, we will improve upon earlier results for $(2, 2)$ -separating systems; the special case is singled out in the pioneering paper of Friedman, Graham, and Ullman [5].

DEFINITION [5]. An (i, j) -separating system for the set S is a family of bipartitions $P_1, \dots, P_t$ of S such that for every pair of disjoint subsets A, B of S such that $|A| = i, |B| = j$, there is at least one partition P_v in the given family for which A and B are contained in (the two) different classes of P_v .

Write $|S| = n$ and denote by $M(i, j, n)$ the minimum number t of partitions in any (i, j) -separating partition system (SPS) for S . In the papers [1]–[5] asymptotic bounds have been obtained for $M(i, j, n)$ if n tends to infinity, while i and j remain fixed. The case $i = 1$ seems to have raised the most interest since it has applications in conflict resolution in multi-access communication. The case $i > 1, j > 1$ is treated only in [1] and [5]. We will return to it at the end of this paper in order to improve on the asymptotic bounds for $M(2, 2, n)$.

Our starting point is, however, a different separating partition system, known under the name of perfect hash functions.

DEFINITION [1]. A (b, k) -system of perfect hash functions for the set S is a system of partitions $P_1, \dots, P_t$ of S into at most b classes such that for every k -element subset A of S there is at least one partition P_v in the system for which every element of A falls into a different class of P_v . (Clearly, such a system exists only if $k \leq b$.)

Following Fredman and Komlós [1], we denote by $Y(b, k, n)$ the smallest t for which a (b, k) -system of perfect hash functions exists for $|S| = n$. The best asymptotic bounds for $Y(b, k, n)$ in the case of $n \rightarrow \infty$, arbitrary finite b and k can be found in Körner and Marton [6] (cf. also [1] and Körner [7]). The problem seems to be hopelessly difficult and no exact asymptotic value of $Y(b, k, n)$ is known except for the trivial case $b = 2$.

* Received by the editors February 24, 1988; accepted for publication March 9, 1988.

† Department Informatique, Ecole Nationale Supérieure des Telecommunications, 75634 Paris CEDEX 13, France. On leave from the Mathematical Institute of HAS, POB 127, 1364 Budapest, Hungary.

Clearly, the nonbinary case has a different flavor, and in particular, has no natural reformulation using the familiar language of extremal set theory. In the hope that problems for binary sequences are appealing to a larger set of combinatorialists, we are now introducing a generalization of perfect hash functions. Our new problem is very nontrivial already in the binary case and will lead to a new treatment of $M(2, 2, n)$.

Log's and exp's are binary.

2. Locally different sequences. Let B be a set of b elements represented by $\{0, 1, \dots, b-1\}$, which we will consider as our alphabet. Let B^t denote the set of sequences of length t of elements from B . Given a set $A \subset B^t$ we say that the coordinates $i_1, i_2, \dots, i_l$ form a *separating domain* for A if the subsequences

$$x_{i1}, x_{i2}, \dots, x_{il} \quad \text{of } x = x_1 \dots x_t, \quad x \in A$$

are all different. Let $L(A)$ denote the smallest number l for which a separation domain with l coordinates exists for A . In this language, the existence of a (b, k) -system of perfect hash functions for an n -element set with $t = Y(b, k, n)$ partitions is equivalent to the existence of an n -element set $E \subset B^t$ such that for every $A \subset E$ with $|A| = k$ we have $L(A) = 1$. In other words, $Y(b, k, n)$ is the smallest length for which n sequences of length t can be constructed from a b -ary alphabet under the condition that for any k sequences (out of these n) their separation domain has just one element.

Intuitively, we shall imagine that a set of sequences is locally different if they have a small separation domain. Systems of perfect hash functions in the sense of Fredman and Komlós [1] are an example of this concept. To be more precise, we shall give the following definition.

DEFINITION. The set $A \subset B^t$ is l -different if its smallest separation domain has at most l coordinates. Let $Z(b, k, n, l)$ be the smallest t for which there exists an n -element set $E \subset B^t$ such that every k -element subset of E is l -different.

Then $Z(b, k, n, 1) = Y(b, k, n)$. Clearly, if l is large relative to k , the problem of determining $Z(b, k, n, l)$ becomes trivial. In fact, we have Proposition 1.

PROPOSITION 1. If $l \geq k - 1$, then $Z(b, k, l, n) = \log n / \log b$.

Proof. We shall prove that every k -element subset of B^t is $(k-1)$ -different, the rest being trivial. We will use induction on k . Notice that no other parameter is relevant.

The statement is obvious for $k = 2$. Therefore set $k > 2$ and suppose that the statement is true for $k' < k$. Let $A \subset B^t$ have k elements. As all its elements differ, consider an arbitrary coordinate in which at least two of them differ. Let A_c be the set of those sequences in A that have $c \in B$ in this coordinate. Then, by definition for some $c \in B$ both A_c and $A \setminus A_c$ are nonempty. The induction hypothesis implies

$$L(A_c) \leq |A_c| - 1 \quad \text{and} \quad L(A \setminus A_c) \leq |A \setminus A_c| - 1.$$

Therefore,

$$L(A) \leq 1 + L(A_c) + L(A \setminus A_c) \leq |A| - 1. \quad \square$$

It seems to us that the information-theoretic technique used in [6] is quite inefficient in dealing with the problem of locally different sequences if $l > 1$. Unfortunately, we cannot suggest any alternative in the general case. Rather, we have attempted to work on the first nontrivial examples. In fact, from now on, we restrict ourselves to the binary alphabet, i.e., $b = 2$. We know that for $k = 2$ the problem is trivial. The case $k = 3$ is settled by Proposition 1. We note that in order for $Z(b, k, n, l)$ to be finite, we must have

$$l \geq \left\lceil \frac{\log k}{\log b} \right\rceil.$$

The first k for which l can be smaller than $k - 1$ is $k = 4$. Unfortunately, we do not know the full answer here. We have no reason to believe that any of our bounds is tight. Their main interest is to provide nontrivial estimates in an area characterized by a lack of methods.

3. Two-different quadruples and (2, 2)-separation. Our key result is Theorem 1.

THEOREM 1. *For every sufficiently large n*

$$3.53 \leq \frac{Z(2, 4, 2, n)}{\log n} \leq \frac{3}{\log \frac{4}{3}} \sim 7.23.$$

Proof. Let us fix some n and let us write $t = Z(2, 4, 2, n)$. Thus we can construct n binary sequences of length t such that for any quadruple of these sequences there are two distinct coordinates where the corresponding four binary pairs are

00
01
10
11

in an arbitrary order.

We claim that this property of our n sequences is equivalent to the following: for any two disjoint pairs of sequences there exists a coordinate in which the two members of both differ simultaneously. In fact, suppose first that our original condition is satisfied. In other words, for every quadruple of our n elements there exist two different binary equipartitions (BEP), generated by the coordinates of the respective sequences. Let us now consider any two disjoint pairs of sequences and their prescribed configurations. Because the two coordinates involved represent BEPs, the two pairs are either both different or both equal, simultaneously. But they can both be equal in at most one of the two coordinates. In the other direction, look at an arbitrary quadruple and form two disjoint pairs in it arbitrarily. There is a coordinate in which both differ simultaneously. This gives one BEP in which the zeros are one class, and the ones are the other class. Now consider these two classes as the two disjoint pairs. Their simultaneous separation gives a new BEP that differs from the previous one.

Let us now proceed to prove

$$3.53 \log n \leq Z(2, 4, 2, n).$$

Consider the fixed optimal configuration of n sequences. For $x \in \{0, 1\}^t$, $y \in \{0, 1\}^t$ let the *Hamming distance* $d(x, y)$ be the number of coordinates in which they differ. Let d be the minimum Hamming distance between sequences in our optimal configuration and let (x^*, y^*) be a pair of sequences achieving it. Let the set of the remaining $(n - 2)$ sequences be denoted by C_t . Then the minimum Hamming distance between different elements of C_t is at least d . On the other hand, any pair of sequences in C_t must differ in at least one of those coordinates in which x^* and y^* disagree, and hence C_t cannot have more elements as there are binary sequences of length d . This gives

$$n - 2 \leq 2^d.$$

This can be viewed as a relation between the cardinality of C_t and its minimum Hamming distance, for every n . Denoting

$$R_t = \frac{\log(n - 2)}{t},$$

we obtain

$$(1) \quad R_t \leq \frac{d}{t}$$

for the sequence $t = t(n)$, where $n \rightarrow \infty$. Let us denote

$$R(D) = \frac{1}{t} \log |D| \quad \text{and} \quad \partial(D) = \frac{1}{t} \min \{d(x, y); x \in D, y \in D, x \neq y\}$$

for any $D \subset \{0, 1\}^t$. We are interested in how large a set with a given minimum Hamming distance can be. We write

$$R_t(\partial) = \max \{R(D); \partial(D) \leq \partial, D \subset \{0, 1\}^t\},$$

and $R(\partial) = \limsup_{t \rightarrow \infty} R_t(\partial)$. Our bound on $Z(2, 4, 2, n)$ will follow from (1) and an analysis of the function $R(\partial)$. In fact, let us suppose that we have a sequence of constructions satisfying (1) with

$$\limsup_{n \rightarrow \infty} R_{t(n)} = R_0.$$

Clearly, for this R_0 we must have

$$R_0 \leq \max \{\partial; R(\partial) \leq \partial\}.$$

Obviously, $R(\partial)$ is a monotonically decreasing function of ∂ , and hence it has a unique point ∂^* for which

$$R(\partial^*) = \partial^*.$$

Thus, by the previous inequality,

$$R_0 \leq R(\partial^*).$$

Although the value of $R(\partial^*)$ is unknown, coding theory provides us with interesting upper bounds on the function $R(\partial)$ that we can use to evaluate our last inequality. Using the celebrated linear programming bound on $R(\partial)$ by McEliece et al. [8], we get

$$R_0 \leq \max \{\partial, R''(\partial) \leq \partial\},$$

where $R''(\partial)$ is the rate-distance bound in [8]. As in the previous argument,

$$R_0 \leq R''(\partial^{**})$$

where ∂^{**} is the unique point in which $R''(\partial^{**}) = \partial^{**}$. Looking up the values of $R''(\partial)$ we complete the proof.

The upper bound can be obtained by randomly selecting the binary sequences, independently and equiprobably among all possible sequences of the stated length. We omit the calculations. $\square$

As an immediate consequence of the lower bound in the above theorem, we obtain a lower bound on $M(2, 2, n)$ that is stronger than the one in [1]. We have Theorem 2.

THEOREM 2. *For every sufficiently large n ,*

$$3.53 \leq \frac{M(2, 2, n)}{\log n} \leq \frac{3}{\log \frac{8}{7}} \sim 15.57.$$

Proof. The upper bound can be obtained by random selection, as in the previous proof. Similarly nonconstructive bounds for this problem can be found in [5]. For this

case those authors obtain $4/\log \frac{8}{7}$. A more careful evaluation gives the above. Once again, we omit the details.

The lower bound will follow upon noticing that

$$(2) \quad Z(2, 4, 2, n) \leq M(2, 2, n).$$

In order to prove (2), it is enough to observe that a $(2, 2)$ -separating partition system can be described equivalently as a set of binary sequences such that for every quadruple of these binary sequences there exist three coordinates in which they represent three different BEPs. By the first part of the proof of Theorem 1 we also know that in any set of binary sequences satisfying the conditions of that theorem, for every quadruple of sequences there are two coordinates where we have two different BEPs. Thus, our present conditions are stronger. This proves (2); whence the original statement follows by Theorem 1. $\square$

It seems unfortunate to reduce this problem to a weaker one and we would like to see a better argument.

Related problems are discussed in [9].

Acknowledgment. It is a pleasure to acknowledge our fruitful discussions with Vera T. Sós.

REFERENCES

- [1] M. FREDMAN AND J. KOMLÓS, *On the size of separating systems and perfect hash functions*, SIAM J. Algebraic Discrete Methods, 5 (1984), pp. 61–68.
- [2] A. G. DJACKOV AND V. V. RYKOV, *Bounds on the length of disjunctive codes*, Problems Inform. Transmission, 18 (1982), pp. 7–13.
- [3] P. ERDÖS, P. FRANKL, AND Z. FÜREDI, *Families of finite sets in which no set is covered by the union of two others*, J. Combin. Theory Ser. A, 33 (1982), pp. 158–166.
- [4] F. K. HWANG AND V. T. SÓS, *Non-adaptive hypergeometric group testing*, Combinatorica, to appear.
- [5] A. D. FRIEDMAN, R. L. GRAHAM, AND J. D. ULLMAN, *Universal single transition time asynchronous state assignments*, IEEE Trans. Comput., 18 (1969), pp. 541–547.
- [6] J. KÖRNER AND K. MARTON, *New bounds for perfect hashing via information theory*, European J. Combin., to appear.
- [7] J. KÖRNER, *Fredman–Komlós bounds and information theory*, SIAM J. Algebraic Discrete Methods, 7 (1986), pp. 560–570.
- [8] R. J. MCÉLIECE, E. R. RODEMICH, H. RUMSEY JR., AND L. R. WELCH, *New upper bounds on the rate of a code via the Delsarte–MacWilliams inequalities*, IEEE Trans. Inform. Theory, 23 (1977), pp. 157–166.
- [9] P. FRANKL AND Z. FÜREDI, *Union-free hypergraphs and probability theory*, European J. Combin., 5 (1984), pp. 127–131.

CERTAIN OCCUPANCY NUMBERS VIA AN ALGORITHM FOR COMPUTING THEIR RATIOS*

CARL E. LANGENHOP† AND WILLIAM E. WRIGHT‡

Abstract. Let $S(p, u)$ be the number of ways of distributing u indistinguishable balls among p distinguishable boxes with at most k balls per box. Let $t_\alpha(u) = S(\alpha - 1, u)/S(\alpha, u)$, which is positive and at most one on the range $0 \leq u \leq k(\alpha - 1)$. An efficient algorithm for computing these nonzero values for t_α for $2 \leq \alpha \leq p$ is given. All arithmetic in the algorithm is performed on numbers between zero and one. The values of $1/S(p, u)$ can be obtained by multiplying together appropriate computed values of t_α for α from two to p .

Key words. occupancy numbers, generating function

AMS(MOS) subject classifications. 05-04, 05A15, 68R05

1. Introduction. Let $S(p, u)$ denote the number of ways of distributing u indistinguishable balls among p distinguishable boxes with at most k balls per box. In this paper we present an efficient algorithm for computing $S(p, u)$ and certain ratios $S(p, u)/S(p', u')$, for very large values of p, u, p' , and u' . Since we keep k fixed throughout, we do not explicitly indicate the dependence of $S(p, u)$ on k .

The numbers $S(p, u)$ and certain ratios involving $S(p, u)$ are important in the calculation of transition probabilities in a Markov chain model for approximating some measures of the dynamic performance of B -trees (cf. [3]). A B -tree of order $2k + 1$ is a very popular type of file organization used for storing and retrieving information on random-access secondary memory (cf. [1] and [2, pp. 471–480]). For such a tree, all nodes except the root are maintained so as to contain at least k but no more than $2k$ keys.

If there are $p \geq 2$ nodes at some B -tree level with $kp + u$ keys distributed among these nodes, then $S(p - 1, u - k)$ is the number of such distributions for which a given one of these nodes contains $2k$ keys. Assuming all distributions are equally likely, we take $S(p - 1, u - k)/S(p, u)$ as the probability that if a new key is added to the given node, then it will contain $2k + 1$ keys and be forced to split into two nodes so as to maintain the B -tree restrictions. Similarly, we take the ratio $t_p(u) = S(p - 1, u)/S(p, u)$ as the probability that a given one of these nodes has only k keys in it so that if one of its keys is deleted the resulting node may be forced to coalesce with another (containing k keys) in order to maintain the B -tree restrictions.

Our Markov chain model has been used to estimate performance measures for very large B -trees. For example, B -trees of order 201 containing 130,000 keys, have been modeled using a modest amount of CPU time, primary memory, and secondary memory.

A critical factor in the practicality of the model is the efficient computation of the ratios involving $S(p, u)$, using the algorithm described herein. The algorithm affords an enormous leap in the size of problem that can be modeled. (For example, previous maximum problem sizes were on the order of 6,000 keys for B -trees of order 201.) Reasons for these improvements are discussed below. The algorithm can be applied to other situations requiring the computation of occupancy numbers for a large number p of boxes.

The algorithm does not compute exact integer values for $S(p, u)$, since such numbers can easily range up to 50 or 100 digits for realistic problems. Instead it computes values

* Received by the editors December 2, 1985; accepted for publication (in revised form) February 9, 1988.

† Department of Mathematics, Southern Illinois University, Carbondale, Illinois 62901.

‡ Department of Computer Science, Southern Illinois University, Carbondale, Illinois 62901.

of $t_p(u)$ using floating point arithmetic with finite precision. The implementation used carried 56 binary bits of precision, which is approximately 17 significant digits in decimal. The algorithm does not suffer from overflow or underflow even for very large p , and it requires only a modest amount of memory and reasonable amount of CPU time. We point out why for large p it is superior to a Pascal's triangle-type recursive calculation and to a direct calculation based on a generating function.

2. Problems of direct calculations. It is clearly possible to put u balls in p boxes with no more than k per box if and only if $0 \leq u \leq kp$. For notational convenience we take

$$(2.1) \quad S(p, u) = 0 \quad \text{if } u < 0 \quad \text{or} \quad u > kp.$$

Otherwise $S(p, u) > 0$. We assume $k \geq 2$ since the case $k = 1$ is trivial; then

$$S(p, u) = \binom{p}{u}.$$

It is known (see [4, Ex. 2-9, p. 32]) and not difficult to show that the numbers $S(p, u)$ are given by the generating function

$$(2.2) \quad g_p(x) = (1 + x + \cdots + x^k)^p = \sum_u S(p, u) x^u.$$

Using (2.1), we may take the sum here over all integer values of u . Direct application of (2.2) is not an efficient way of computing $S(p, u)$ for large values of p and u . Another approach exploits the fact that

$$(2.3) \quad (1 - x)g_1(x) = 1 - x^{k+1}.$$

Since $g_p(x) = g_1(x)^p$, we may write

$$(2.4) \quad g_p(x) = (1 - x^{k+1})^p (1 - x)^{-p}.$$

Expanding the binomial factors on the right, multiplying the two series, and equating coefficients of like powers of x , we may derive the relation

$$(2.5) \quad S(p, u) = \sum_{h=0}^{\lfloor u/(k+1) \rfloor} (-1)^h \binom{p}{h} \binom{p-1+u-(k+1)h}{p-1}$$

where $\lfloor s \rfloor$ denotes the greatest integer in s . Both binomial coefficients in (2.5) are positive so the terms in this sum alternate in sign. These terms can be quite large numerically with $S(p, u)$ of only moderate size in comparison so there can be a loss of precision in implementing (2.5) on the computer. For example, if $k = 3$, $p = 10$, and $u = 20$, then we have

$$\begin{aligned} S(10, 20) &= 10,015,005 - 20,429,750 + 13,226,850 - 2,917,200 + 150,150 - 252 \\ &= 44,803. \end{aligned}$$

From (3.5) below, the derivation of which is given in § 3, we obtain the following formula:

$$S(p+1, u) = S(p+1, u-1) + S(p, u) - S(p, u-k-1).$$

This formula permits a Pascal's triangle-type generation of $S(p+1, *)$ from $S(p, *)$. While this approach is generally suitable if values of $S(p, u)$ are needed, it has practical limitations for large p caused by floating point overflows.

The algorithm we present circumvents this overflow problem by computing the values of

$$(2.6) \quad t_p(u) = S(p-1, u)/S(p, u)$$

for $p \geq 2$ and $0 \leq u \leq kp$. The nonzero values of $t_p(u)$ lie between zero and one. The quantity $1/S(p, u)$ can be obtained if needed by multiplying together appropriate values of t_i for i ranging from two to p . The algorithm is inductive in that it computes the values of t_{p+1} from those of t_p , beginning with the easily obtained values of t_2 .

The principal benefit of our algorithm is realized if the actual values $S(p, u)$ are not needed, but instead the ratios $t_p(u)$, or quantities based on these ratios, are needed. That is the case with the Markov chain model discussed in § 1, and potentially with other applications. Again, we will show how $S(p, u)$ can itself be obtained from the t_i values, although this computation would of course also be susceptible to overflow problems for large p .

3. Some properties of $S(p, u)$. Before considering the algorithm for computing $t_p(u)$, it is necessary to establish some relations satisfied by the function S . These are used in the development of our algorithm.

PROPOSITION 3.1. *If $p \geq 1$, then S satisfies the following:*

$$(3.1) \quad S(p, 0) = 1,$$

$$(3.2) \quad S(p, u) > 0, 0 \leq u \leq kp \quad \text{and} \quad S(p, u) = 0 \quad \text{otherwise},$$

$$(3.3) \quad S(p+1, u) > S(p, u), \quad 1 \leq u \leq k(p+1),$$

$$(3.4) \quad S(p, u) = S(p, kp - u),$$

$$(3.5) \quad S(p+1, u) - S(p+1, u-1) = S(p, u) - S(p, u-k-1).$$

Proof. The proof can be established analytically using the definition of $g_p(x)$ in (2.2). In some cases an easy combinatorial argument can be given. Thus (3.1) and (3.2) are obvious from the combinatorial significance of $S(p, u)$. For (3.3) we note that if $1 \leq u \leq kp$ (when both sides are positive), the count represented by $S(p+1, u)$ includes all cases when the last box is empty, which are counted by $S(p, u)$, as well as those when it is not. When $k(p+1) \leq u \leq k(p+1)$, relation (3.3) follows from (3.2). Relation (3.4) follows by considering the situation of putting u red balls in p boxes with no more than k per box and then filling up the boxes with $kp - u$ green balls to a total of k balls per box. We may equally well put the $kp - u$ green balls in first and then fill each box to a level of k with a total of u red balls. For (3.5) it is easier to give an analytic proof. Using (2.3), we get

$$(1-x)g_{p+1}(x) = (1-x^{k+1})g_p(x).$$

Equating coefficients of like powers of x on the two sides of this relation, we readily get (3.5). $\square$

To give a combinatorial proof of (3.5) we note for any distribution of u balls among $p+1$ boxes with at most k per box that either the last box is empty or it is not. Clearly there are $S(p, u)$ arrangements for which the last box is empty. To count the ways in which the last box is not empty we can first put one ball in the last box and then distribute the remaining $u-1$ among all $p+1$ boxes. This is done in $S(p+1, u-1)$ ways if no more than k are put in any one box. But we must not include the possibility of putting k of these $u-1$ balls in the last box along with the original one. There would be $S(p, u-1-k)$ ways of doing this. Hence

$$S(p+1, u) = S(p, u) + [S(p+1, u-1) - S(p, u-k-1)],$$

which is (3.5).

From the definition (2.6) and properties (3.1)–(3.3) we immediately deduce for $p \geq 2$ that

$$(3.6a) \quad t_p(0) = 1,$$

$$(3.6b) \quad 0 < t_p(u) < 1 \quad \text{if } 1 \leq u \leq k(p-1),$$

$$(3.6c) \quad t_p(u) = 0 \quad \text{if } k(p-1) + 1 \leq u \leq kp.$$

From (2.2), or the combinatorial significance, it is clear that

$$(3.7) \quad S(1, u) = 1, \quad 0 \leq u \leq k.$$

Applying (3.5) with $p = 1$ and using (2.1), we then find that

$$(3.8) \quad S(2, u) = S(2, u-1) + 1, \quad 0 \leq u \leq k.$$

It then follows from (3.8) and (3.1) that $S(2, u) = u + 1$, $0 \leq u \leq k$. This with (3.7) produces the values of $t_2(u)$, which we use to start our algorithm:

$$(3.9) \quad t_2(u) = \frac{1}{1+u}, \quad 0 \leq u \leq k.$$

Some additional facts regarding $S(p, u)$ are formulated in the following proposition.

PROPOSITION 3.2. *If $p \geq 2$, then $S(p, u)$ is strictly increasing as u goes from zero to $\lfloor kp/2 \rfloor$ and is strictly decreasing as u goes from $\lceil kp/2 \rceil$ to kp . Moreover,*

$$(3.10) \quad S(p, \lfloor kp/2 \rfloor) = S(p, \lceil kp/2 \rceil).$$

Here $\lceil s \rceil$ denotes the least integer greater than or equal to s .

Proof. First, we consider relation (3.10), which holds trivially when kp is even. If $kp = 2m + 1$ for an integer $m \geq 1$, then $\lfloor kp/2 \rfloor = m$ while $\lceil kp/2 \rceil = m + 1$. But in this case $kp - m = m + 1$, so (3.10) holds by virtue of (3.4). To show that $S(p, u)$ is strictly increasing as u goes from zero to $\lfloor kp/2 \rfloor$, we note that this is true for $p = 2$ by (3.8). Suppose then that it is true for some $p \geq 2$. Then we have that

$$S(p, u) > S(p, u - k - 1), \quad 0 \leq u \leq \lfloor kp/2 \rfloor.$$

It now follows from (3.5) that $S(p + 1, u)$ is strictly increasing as u goes from zero to $\lfloor kp/2 \rfloor$. To extend this up to $\lfloor k(p + 1)/2 \rfloor$ we use (3.4) to rewrite (3.5) as

$$(3.11) \quad S(p + 1, u) - S(p + 1, u - 1) = S(p, kp - u) - S(p, u - k - 1).$$

But if $\lfloor kp/2 \rfloor + 1 \leq u \leq \lfloor k(p + 1)/2 \rfloor$, then $0 \leq u - k - 1 < kp - u \leq \lfloor kp/2 \rfloor$, so the right side in (3.11) is positive by the inductive hypothesis. Hence $S(p + 1, u)$ is strictly increasing as u continues up to $\lfloor k(p + 1)/2 \rfloor$ and the first part of the proposition is true by induction. The second part follows from that and the relation (3.4). $\square$

In our algorithm we also use the ratios

$$(3.12) \quad r_p(u) = \frac{S(p, u-1)}{S(p, u)}, \quad 0 \leq u \leq kp.$$

By (3.7) and (2.1) we have

$$(3.13) \quad r_1(u) = \begin{cases} 0, & u = 0, \\ 1, & 1 \leq u \leq k. \end{cases}$$

Using Proposition 3.2 we have immediately that if $p \geq 2$, then

$$(3.14a) \quad 0 < r_p(u) < 1 \quad \text{if } 1 \leq u \leq \lfloor kp/2 \rfloor,$$

$$(3.14b) \quad r_p(u) > 1 \quad \text{if } \lceil kp/2 \rceil + 1 \leq u \leq kp,$$

$$(3.14c) \quad r_p(\lceil kp/2 \rceil) = 1 \quad \text{if } kp \text{ is odd.}$$

4. Relations between t_p and r_p . If $p \geq 2$, then by (2.6) and (3.12) we may write

$$(4.1) \quad r_p(u) = t_p(u)r_{p-1}(u)/t_p(u-1), \quad 1 \leq u \leq k(p-1).$$

Knowing the values of t_p and r_{p-1} , we can use this to obtain the values of r_p in the range given in (4.1). Thus from (3.13) and (3.9) we can get the values of $r_2(u)$ for $1 \leq u \leq k$.

The relation (4.1) is not valid for $k(p-1) + 1 \leq u \leq kp$ since then $r_{p-1}(u)$ is not defined. But in this range the values of $r_p(u)$ can be obtained using (3.4). Thus

$$(4.2) \quad r_p(u) = \frac{S(p, kp-u+1)}{S(p, kp-u)} = \frac{1}{r_p(kp-u+1)}$$

and if $k(p-1) + 1 \leq u \leq kp$, and $p \geq 2$, then $1 \leq kp-u+1 \leq k(p-1)$.

We now divide (3.5) by $S(p+1, u)$ when $0 \leq u \leq k(p+1)$ so $S(p+1, u)$ is positive. In light of (2.6) and (3.12) the result is

$$(4.3) \quad 1 - r_{p+1}(u) = t_{p+1}(u) - S(p, u-k-1)/S(p+1, u)$$

and this holds for $0 \leq u \leq k(p+1)$. The last ratio in (4.3) can be expressed in terms of t_{p+1} and r_p . We find it convenient to introduce

$$(4.4) \quad R_p(u) = S(p, u-k)/S(p, u), \quad 0 \leq u \leq kp$$

and by (2.1) and (3.12) we have

$$(4.5) \quad R_p(u) = \begin{cases} 0, & 0 \leq u \leq k-1, \\ \prod_{\alpha=u-k+1}^u r_p(\alpha), & k \leq u \leq kp. \end{cases}$$

Then we may write

$$S(p, u-k-1)/S(p+1, u) = t_{p+1}(u)r_p(u-k)R_p(u)$$

in the range $k+1 \leq u \leq kp$. For $0 \leq u \leq k$ the last ratio in (4.3) is zero. Thus (4.3) can be written as

$$(4.6a) \quad 1 - r_{p+1}(u) = t_{p+1}(u), \quad 0 \leq u \leq k$$

$$(4.6b) \quad 1 - r_{p+1}(u) = t_{p+1}(u)[1 - r_p(u-k)R_p(u)], \quad k+1 \leq u \leq kp.$$

Now we use (4.1) with p replaced by $p+1$. If we substitute the result in (4.3) after using (4.6), we may solve for $t_{p+1}(u)$ in terms of $t_{p+1}(u-1)$. We get

$$(4.7a) \quad t_{p+1}(u) = \frac{t_{p+1}(u-1)}{r_p(u) + t_{p+1}(u-1)}, \quad 1 \leq u \leq k,$$

$$(4.7b) \quad t_{p+1}(u) = \frac{t_{p+1}(u-1)}{r_p(u) + t_{p+1}(u-1)[1 - r_p(u-k)R_p(u)]}, \quad k+1 \leq u \leq kp.$$

Our algorithm can now be described in general terms as follows. Use (3.9), (3.13), and (4.1) to generate the values of $r_2(u)$; then starting with $t_3(0) = 1$ use (4.7a), (4.7b)

to compute the nonzero values of t_3 ; then (4.1) is used to compute the values of r_3 and (4.7a), (4.7b) the values of t_4 , etc.

The algorithm as given later makes use of modified formulas in some ranges so that all values of $t_p(u)$ and $r_p(u)$, entering into and resulting from the computation, are between zero and one. Note from (3.14b) that $r_p(u) > 1$ in a certain range for u . In this range we replace $r_p(u)$ using (4.2) and rewrite our formulas by using $r_p(kp - u + 1)$, which is then between zero and one.

For $p \geq 2$ we have $\lfloor kp/2 \rfloor \geq k$, so by (4.5) and (3.14a) we have $0 < R_p(k) < 1$. The remaining values of $R_p(u)$ can be computed recursively from

$$(4.8) \quad R_p(u) = \frac{R_p(u-1)r_p(u)}{r_p(u-k)}, \quad k+1 \leq u \leq kp.$$

As before we can modify this by using (4.2) when $r_p(u) > 1$.

Finally, in (4.7b) we can avoid the problem of having $r_p(u-k)R_p(u) > 1$. Indeed, by (2.6), (3.4), and (4.4) we have

$$(4.9) \quad t_{p+1}(u) = R_p(k(p+1)-u)t_{p+1}(k(p+1)-u).$$

This can be used in the range $\lfloor k(p+1)/2 \rfloor + 1 \leq u \leq kp$ to get the values of $t_{p+1}(u)$ from ones computed earlier.

The restricted ranges for the values of u are very important because they preclude the possibility of overflow in carrying out the algorithm. The proof of this is presented in § 6. The algorithm itself is given formally in § 5.

5. The algorithm for t_p . Recall that we assume $k \geq 2$ and that some $t_p(u)$ values are zero by (3.6c). Our algorithm gives only the nonzero values, namely, when $0 \leq u \leq k(p-1)$. It consists of the following four steps.

Step 1. Initialize using

$$\begin{aligned} r_1(u) &= 1, & 1 \leq u \leq \lceil k/2 \rceil \\ t_2(u) &= \frac{1}{1+u}, & 0 \leq u \leq k. \\ p &= 2 \end{aligned}$$

For $p < p_{\text{final}}$ do Steps 2–4, and then terminate.

Step 2. Compute $r_p(u)$, $1 \leq u \leq \lceil kp/2 \rceil$ using

$$r_p(u) = \begin{cases} \frac{t_p(u)r_{p-1}(u)}{t_p(u-1)}, & 1 \leq u \leq \left\lfloor \frac{k(p-1)}{2} \right\rfloor, \\ \frac{t_p(u)}{t_p(u-1)r_{p-1}(k(p-1)-u+1)}, & \left\lfloor \frac{k(p-1)}{2} \right\rfloor + 1 \leq u \leq \left\lfloor \frac{kp}{2} \right\rfloor. \end{cases}$$

Step 3. Compute $R_p(u)$, $k \leq u \leq \lfloor k(p+1)/2 \rfloor$ using

$$\begin{aligned} R_p(k) &= \prod_{\alpha=1}^k r_p(\alpha), \quad \text{and} \\ R_p(u) &= \begin{cases} \frac{R_p(u-1)r_p(u)}{r_p(u-k)}, & k+1 \leq u \leq \left\lfloor \frac{kp}{2} \right\rfloor^*, \\ \frac{R_p(u-1)}{r_p(u-k)r_p(kp-u+1)}, & \left\lfloor \frac{kp}{2} \right\rfloor + 1 \leq u \leq \left\lfloor \frac{k(p+1)}{2} \right\rfloor. \end{cases} \end{aligned}$$

(*Note that this range is empty for $p = 2$ but not for $p \geq 3$.)

Step 4. Compute $t_{p+1}(u)$, $0 \leq u \leq kp$, using

$$t_{p+1}(0) = 1, \quad \text{and}$$

$$t_{p+1}(u) = \begin{cases} \frac{t_{p+1}(u-1)}{r_p(u) + t_{p+1}(u-1)}, & 1 \leq u \leq k, \\ \frac{t_{p+1}(u-1)}{r_p(u) + t_{p+1}(u-1)[1 - r_p(u-k)R_p(u)]}, & k+1 \leq u \leq \left\lceil \frac{kp}{2} \right\rceil^*, \\ \frac{r_p(kp-u+1)t_{p+1}(u-1)}{1 + r_p(kp-u+1)t_{p+1}(u-1)[1 - r_p(u-k)R_p(u)]}, & \left\lceil \frac{kp}{2} \right\rceil + 1 \leq u \leq \left\lfloor \frac{k(p+1)}{2} \right\rfloor, \\ R_p(k(p+1)-u)t_{p+1}(k(p+1)-u), & \left\lfloor \frac{k(p+1)}{2} \right\rfloor + 1 \leq u \leq kp \end{cases}$$

and then increment p by one.

(*Note that this range is empty for $p = 2$ but not for $p \geq 3$.)

6. Properties of the algorithm. The formulas in the algorithm are the relevant ones in § 4 modified as indicated at the end of that section. Here we show that all factors appearing in the formulas in § 5 are indeed between zero and one. This is easily verified for the factors r_p or r_{p-1} by using (3.14a), (3.14b), and (3.14c) and (4.2). It is also true for the factors t_p or t_{p+1} by (3.6a) and (3.6b). It remains to show that the values of R_p that appear are also in the range from zero to one.

PROPOSITION 6.1. *If $p \geq 2$ and $k \leq u \leq \lfloor k(p+1)/2 \rfloor$, then $0 < R_p(u) \leq 1$.*

Proof. For $k \leq u \leq \lfloor kp/2 \rfloor$ we have $0 < S(p, u-k) < S(p, u)$ by (3.2) and Proposition 3.2. Hence by (4.4) the result is valid on this range of u . If $\lfloor kp/2 \rfloor + 1 \leq u \leq \lfloor k(p+1)/2 \rfloor$, then we use (3.4) to write

$$(6.1) \quad R_p(u) = \frac{S(p, k(p+1)-u)}{S(p, u)}.$$

But on this range we have $\lceil kp/2 \rceil \leq u \leq k(p+1) - u \leq kp$ so

$$0 < S(p, k(p+1)-u) \leq S(p, u)$$

by Proposition 3.2 and again we have $0 < R_p(u) \leq 1$. $\square$

Using Proposition 6.1, we can verify that for all appearances of R_p in the formulas in § 5 the values are between zero and one. From the proof above and (6.1) it is clear that we can have $R_p(u) = 1$ only when $u = k(p+1) - u$, i.e., $u = k(p+1)/2$.

Since the computed values of $r_p(u)$, $t_p(u)$ and $R_p(u)$ are all between zero and one, we can show that neither overflow nor underflow presents a problem for our algorithm for any values of p and u , which conceivably may be used in its application. Overflow occurs when floating point values exceed the largest representable on the computer, say 16^{63} , which is approximately 10^{76} . An overflow forces termination of the program.

An underflow occurs when a floating point value is smaller numerically than the smallest positive number representable on the computer, say 16^{-64} . An underflow causes the computed value to be set to zero but does not terminate the program. Thus an underflow is equivalent to approximating a number very near zero by zero itself and is comparable to other truncation errors that are inherent in finite arithmetic. An underflow

will cause the program to terminate if it leads to an attempt to divide by zero. We will show that there is no such underflow problem in any of the denominators in the formulas of our algorithm because the relevant values of $r_p(w)$ and $t_p(w)$ are bounded away from zero. Since all numerators are no larger than one numerically, then clearly there will be no possibility of an overflow.

We now consider the denominators in the various formulas of the algorithm. The possibility of an underflow in one of the denominators can arise only from a too small value of $r_p(w)$ (or $r_{p-1}(w)$) or a too small value of $t_p(w)$ for some argument value w determined by the particular step in the algorithm. It is shown in the Appendix (§ 8) that $r_p(u)$ is an increasing function of u and since $w \geq 1$ in all cases then

$$(6.2) \quad r_p(w) \geq r_p(1) = \frac{S(p, 0)}{S(p, 1)} = \frac{1}{p}.$$

Thus none of the $r_p(w)$ (or $r_{p-1}(w)$) occurrences will cause an underflow problem for any value of p that conceivably might be used in applying the algorithm. This result eliminates Steps 3 and 4 of the algorithm from concern.

Step 2 of the algorithm involves the factor $t_p(w)$ in the denominator with $w = u - 1$ and $u \leq \lceil kp/2 \rceil$. But we show in the Appendix that $t_p(u)$ is a decreasing function of u so

$$(6.3) \quad t_p(w) \geq t_p(\lceil kp/2 \rceil)$$

for the relevant values of w . We also show in the Appendix that $t_p(\lceil kp/2 \rceil) \rightarrow 1/(k+1)$ as $p \rightarrow \infty$. (In our computations we have found even for small to moderate sized values of p that $t_p(\lceil kp/2 \rceil)$ is very near its limit value $1/(k+1)$.) Thus it would appear that none of the $t_p(w)$ occurrences will cause an underflow problem for values of k and p that conceivably may be used in applying the algorithm.

7. Recovery of $S(p, u)$. Using the algorithm in § 5, we may compute and store the nonzero values of t_α for $\alpha = 2$ up to some $p > 2$, which is of interest. If the values of $S(p, u)$ with $1 \leq u \leq kp$ are desired, they can be obtained from the formula

$$(7.1) \quad \frac{1}{S(p, u)} = \left[\prod_{\alpha=h+1}^p t_\alpha(u) \right] \left[\prod_{\alpha=2}^h t_\alpha(kh-u) \right]$$

where $h = \lceil u/k \rceil$ and the first product on the right is taken to be one if $h = p$ and the second product is taken to be one if $h = 1$. For h as defined we have $0 \leq kh - u \leq k(\alpha - 1)$ for $\alpha = 2$ up to h and $0 \leq u \leq k(\alpha - 1)$ for $\alpha = h + 1$ up to p . Thus the products in (7.1) are well defined and nonzero since $0 < t_\alpha(v) \leq 1$ for $0 \leq v \leq k(\alpha - 1)$.

To verify (7.1) we note that by (2.6) we have

$$(7.2) \quad \prod_{\alpha=h+1}^p t_\alpha(u) = \frac{S(h, u)}{S(p, u)} \quad \text{if } 1 \leq h \leq p-1$$

and by (3.4)

$$(7.3) \quad \prod_{\alpha=2}^h t_\alpha(kh-u) = \frac{S(1, kh-u)}{S(h, u)} \quad 2 \leq h \leq p.$$

Hence the product on the right in (7.1) is $S(1, u)/S(p, u)$ if $h = 1$ and $1 \leq u \leq k$; the product is $S(1, kh-u)/S(p, u)$ if $2 \leq h \leq p$ and $k(h-1)+1 \leq u \leq kh$. But by (3.7) we have $S(1, u) = 1$ for $1 \leq u \leq k$, while in the alternative case we have $0 \leq kh-u \leq k-1$ and $S(1, kh-u) = 1$. Thus (7.1) holds. $\square$

In the above application of the algorithm certain data structures are required for storing the necessary values of the functions t , r , and R . To compute the values of $1/S(p, u)$ we need the values of t_α for $2 \leq \alpha \leq p$. The only nonzero values of $t_\alpha(u)$ are those for $0 \leq u \leq k(\alpha - 1)$. But $t_\alpha(0) = 1$ so all the essential t values can be stored in a one-dimensional array of length

$$(7.4) \quad \sum_{\alpha=2}^p k(\alpha - 1) = \frac{kp(p-1)}{2}.$$

To obtain the values of t_α , $2 \leq \alpha \leq p$, we need the values of $r_{\alpha-1}$ and r_α for $\alpha \leq p-1$. The number of values needed is

$$(7.5) \quad \lceil k(\alpha - 1)/2 \rceil + \lceil k\alpha/2 \rceil \leq k(p-1).$$

Thus a one-dimensional array of length kp will suffice for storing the needed r values. Similarly the number of R_α values needed for $\alpha \leq p-1$ is

$$(7.6) \quad \lfloor k(\alpha + 1)/2 \rfloor - k \leq k(p-2)/2.$$

From (7.4)–(7.6) it is clear that the execution time for the algorithm is of order kp^2 . In our application to the study of B -trees it was not necessary to retain all values of t_α , $2 \leq \alpha \leq p$. The required values were moved to another data structure as they were generated. It was necessary only to save enough t values to keep the process going. That is, we needed only the values of $t_{\alpha-1}$ and t_α at any one stage, a total of $k(\alpha - 2) + k(\alpha - 1) \leq 2kp$ for the maximum p involved. For studying B -trees of order $2k + 1$ with n keys the maximum value of p we required is given by $\lfloor (n + 1)/(k + 1) \rfloor$ so the storage needs for the t algorithm were of order $2k(n/k) = 2n$. The execution time though (see (7.4)) was of order $k(n/k)^2/2 = n^2/2k$.

8. Appendix. Here we use the fact that

$$(8.1) \quad S(p+1, u) = \sum_{v=0}^k S(p, u-v).$$

This follows easily from the relation (see (2.2))

$$g_{p+1}(x) = (1 + x + \cdots + x^k)g_p(x).$$

LEMMA 8.1. *The following are equivalent:*

$$(8.2) \quad \forall u, \quad S(p, u)S(p, u) \geq S(p, u+1)S(p, u-1),$$

$$(8.3) \quad \forall u, \quad \text{if } v \leq u, \quad S(p, u)S(p, v) \geq S(p, u+1)S(p, v-1).$$

Proof. Clearly (8.3) implies (8.2). On the other hand, (8.2) implies that $r_p(u) = S(p, u-1)/S(p, u)$ is increasing as u goes from zero to kp since $S(p, u) > 0$ in that range. Hence, if $0 \leq v \leq u \leq kp-1$, then $r_p(u+1) \geq r_p(v)$, which gives (8.3) in that range. In all other cases, when $v \leq u$ then $S(p, u+1)S(p, v-1) = 0$ while

$$S(p, u)S(p, v) \geq 0. \quad \square$$

THEOREM 8.2. *Relation (8.2) is valid for all $p \geq 1$.*

Proof. Relation (8.2) certainly holds for $p = 1$ since $S(1, u) = 1$ for $0 \leq u \leq k$ and otherwise $S(1, u) = 0$. Suppose then that (8.2) holds for some $p \geq 1$. Then also (8.3) holds for this p by virtue of Lemma 8.1. Using (8.1), we have

$$(8.4) \quad S(p+1, u)S(p+1, u) = \sum_{v=0}^k \sum_{w=0}^k S(p, u-v)S(p, u-w).$$

On both sides of (8.4) we now replace the first u by $u + 1$ and the second u by $u - 1$. Then we replace the summation indices v and w by $v + 1$ and $w - 1$, respectively, to get

$$(8.5) \quad S(p+1, u+1)S(p+1, u-1) = \sum_{v=-1}^{k-1} \sum_{w=1}^{k+1} S(p, u-v)S(p, u-w).$$

Many terms cancel on the right when we subtract (8.5) from (8.4). The difference is

$$(8.6) \quad \begin{aligned} & S(p+1, u)S(p+1, u) - S(p+1, u+1)S(p+1, u-1) \\ &= \sum_{v=0}^k S(p, u-v)S(p, u) + \sum_{w=1}^k S(p, u-k)S(p, u-w) \\ &\quad - \sum_{w=1}^{k+1} S(p, u+1)S(p, u-w) - \sum_{v=0}^{k-1} S(p, u-v)S(p, u-k-1). \end{aligned}$$

In the first two sums on the right in (8.6) we replace the summation indices v and w by $w - 1$ and $v + 1$, respectively. The result can be written as

$$\begin{aligned} & S(p+1, u)S(p+1, u) - S(p+1, u+1)S(p+1, u-1) \\ &= \sum_{w=1}^{k+1} [S(p, u)S(p, u-w+1) - S(p, u+1)S(p, u-w)] \\ &\quad + \sum_{v=0}^{k-1} [S(p, u-v-1)S(p, u-k) - S(p, u-v)S(p, u-k-1)]. \end{aligned}$$

Here $u - w + 1 \leq u$ and $u - k \leq u - v - 1$, so each term in both sums is nonnegative by (8.3). Hence we get (8.2) with p replaced by $p + 1$, and the induction is complete. $\square$

THEOREM 8.3. *For all u*

$$(8.7) \quad S(p+1, u+1)S(p, u) \geq S(p+1, u)S(p, u+1).$$

Proof. Since (8.2) holds, then so does (8.3). Using the latter and (8.1), we can write

$$\begin{aligned} S(p+1, u+1)S(p, u) &= S(p, u+1)S(p, u) + \sum_{v=1}^k S(p, u)S(p, u+1-v) \\ &\geq S(p, u+1)S(p, u) + \sum_{v=1}^k S(p, u+1)S(p, u-v) \\ &= S(p, u+1)S(p+1, u). \end{aligned} \quad \square$$

COROLLARY 8.4. *The following hold for $0 \leq u \leq kp - 1$:*

$$(8.8) \quad r_p(u) \leq r_p(u+1),$$

$$(8.9) \quad t_p(u) \geq t_p(u+1).$$

Proof. Inequality (8.8) is (8.2), in effect, since $S(p, u) > 0$ when $0 \leq u \leq kp - 1$. Similarly, (8.9) is (8.7) with p replaced by $p - 1$, since $t_p(u) = S(p-1, u)/S(p, u)$.

Remark. We are indebted to our colleague Dr. John R. Greene, Department of Mathematics, Southern Illinois University, Carbondale, Illinois, for the proofs of Lemma 8.1 and Theorems 8.2 and 8.3. The content of (8.7) has an interesting combinatorial interpretation that seems clear intuitively but evidently is difficult to prove.

We now wish to estimate the size of $t_p(u-1)$, which appears in the denominator in Step 2 of our algorithm. By (8.9), the smallest value of this factor that concerns us is

at $u = \lceil kp/2 \rceil$, and this is at least as large as $t_p(\lceil kp/2 \rceil)$. For notational convenience we consider this in the form $t_{p+1}(u_0(p))$ where $u_0(p) = \lceil k(p+1)/2 \rceil$. As we now show, the size of $t_{p+1}(u_0(p))$ for large p can be estimated using the Central Limit Theorem.

We consider first the discrete random variable U with distribution

$$(8.10) \quad P(U = u) = \begin{cases} 1/(k+1), & u = 0, 1, \dots, k, \\ 0 & \text{otherwise.} \end{cases}$$

Then

$$E(x^U) = \sum_{u=0}^k \frac{x^u}{1+k} = \frac{g_1(x)}{1+k}.$$

From this we can compute $E(U) = k/2$ and $\text{Var}(U) = k(k+2)/12$.

Now let $U_1, U_2, \dots, U_p$ be independent random variables with the same distribution as U and let $W_p = U_1 + U_2 + \dots + U_p$. Then

$$(8.11) \quad E(x^{W_p}) = \prod_{\alpha=1}^p E(x^{U_\alpha}) = g_1(x)^p / (k+1)^p = g_p(x) / (k+1)^p.$$

On the other hand,

$$(8.12) \quad E(x^{W_p}) = \sum_w P(W_p = w) x^w.$$

Using (2.2) and comparing (8.11) and (8.12), we see that

$$(8.13) \quad S(p, u) = (k+1)^p P(W_p = u).$$

But for $0 \leq u \leq k(p+1)$

$$t_{p+1}(u) = \frac{S(p, u)}{S(p+1, u)} = \frac{S(p, u)}{\sum_{v=0}^k S(p, u-v)}.$$

Thus in light of (8.13) we can write

$$(8.14) \quad t_{p+1}(u_0(p)) = P(W_p = u_0(p)) / P(u_0(p) - k \leq W_p \leq u_0(p)).$$

Now $E(W_p) = pE(U) = pk/2$ and $\text{Var}(W_p) = p \text{Var}(U) = pk(k+2)/12$. If we define $s = (\text{Var}(W_p))^{-1/2}$, then the random variable

$$(8.15) \quad W_p^* = s(W_p - pk/2)$$

approaches the standard normal in probability as $p \rightarrow \infty$ or, equivalently, as $s \rightarrow 0$.

THEOREM 8.5. $\lim_{p \rightarrow \infty} t_{p+1}(u_0(p)) = 1/(k+1)$.

Proof. First we assume that k is even. Then $u_0(p) = k(p+1)/2$. Since W_p takes on integer values only, we have

$$\begin{aligned} P(W_p = u_0(p)) &= P(u_0(p) - 1/2 \leq W_p \leq u_0(p) + 1/2) \\ &= P(s(k-1)/2 \leq W_p^* \leq s(k+1)/2). \end{aligned}$$

By the Central Limit Theorem we then have for large p the approximation

$$(8.16) \quad P(W_p = u_0(p)) = \Phi(s(k+1)/2) - \Phi(s(k-1)/2),$$

where $\Phi(w) = \int_{-\infty}^w \varphi(x) dx$, $\varphi(x) = e^{-x^2/2} / \sqrt{2\pi}$. Similarly we get the approximation

$$(8.17) \quad \begin{aligned} P(u_0(p) - k \leq W_p \leq u_0(p)) &= P(|W_p^*| \leq s(k+1)/2) \\ &= 2\Phi(s(k+1)/2) - 1. \end{aligned}$$

Since $\Phi(0) = 1$ and $\varphi(0) \neq 0$, we can use (8.16) and (8.17) in (8.14) and apply l'Hôpital's rule to compute

$$\begin{aligned}\lim_{p \rightarrow \infty} t_{p+1}(u_0(p)) &= \lim_{s \rightarrow 0} \frac{\Phi(s(k+1)/2) - \Phi(s(k-1)/2)}{2\Phi(s(k+1)/2) - 1} \\ &= \frac{\varphi(0)(k+1)/2 - \varphi(0)(k-1)/2}{2\varphi(0)(k+1)/2} = \frac{1}{k+1}.\end{aligned}$$

When k is odd then the same computations apply when we allow $p \rightarrow \infty$ through the odd integers for then $u_0(p) = k(p+1)/2$ as before. When k is odd and p is even, then $u_0(p) = [k(p+1)+1]/2$ and (8.16) and (8.17) are changed accordingly. The end result is the same. $\square$

REFERENCES

- [1] R. BAYER AND E. MCCREIGHT, *Organization and maintenance of large ordered indexes*, Acta Inform., 1 (1972), pp. 173–189.
- [2] D. E. KNUTH, *The Art of Computer Programming*, Vol. 3, Addison-Wesley, Reading, MA, 1973.
- [3] C. E. LANGENHOP AND W. E. WRIGHT, *An efficient model for representing and analyzing B-trees*, Proc. Assoc. Comput. Mach., 85 (1985), pp. 34–40.
- [4] C. L. LIU, *Introduction to Combinatorial Mathematics*, McGraw-Hill, New York, 1968.

TOURNAMENT RANKING WITH EXPECTED PROFIT IN POLYNOMIAL TIME*

SVATOPLUK POLJAK†, VOJTĚCH RÖDL‡, AND JOEL SPENCER§

Abstract. An $O(n^3 \log n)$ algorithm is presented that, for a given tournament on n vertices, produces a ranking with fit at least $\frac{1}{2}\binom{n}{2} + c_1 n^{3/2}$, where $c_1 = \frac{1}{8}\pi^{-1/2}$.

Key words. tournament, probabilistic method

AMS(MOS) subject classification. 05C99

A *tournament* on n players is a set T of ordered pairs (i, j) such that either $(i, j) \in T$ or $(j, i) \in T$ but not both. A *ranking* of T is a linear ordering P of $[n] = \{1, 2, \dots, n\}$. The *fit* of a ranking P is the number of pairs i, j with $(i, j) \in T$ and $P(i) < P(j)$. The *tournament ranking problem* is to find a ranking with maximum fit for a given tournament. The weighted version of the problem is known to be NP-complete (see the Feedback-Arc-Set problem in [3]). The problem has several applications (see, e.g., [5]), and some approaches to solve it have been developed (e.g., [2], [3]).

It is also well known that there are tournaments that are almost unrankable. More precisely, let

$$h(n) = \min_{T_n} \max_P |T_n \cap P|$$

where T_n is a tournament on n players and P is a ranking. Spencer [8] and de la Vega [9] proved

$$h(n) \leq \frac{1}{2} \binom{n}{2} + c_2 n^{3/2}$$

where c_2 is a positive constant. On the other hand, there is also an analogous lower bound

$$(1) \quad h(n) \geq \frac{1}{2} \binom{n}{2} + c_1 n^{3/2}$$

with $c_1 < c_2$, by Spencer using the probabilistic methods [7].

Let us mention an important fact. For arbitrary ranking P , either P itself or its inverse P^{-1} has the fit at least $\frac{1}{2}\binom{n}{2}$. Hence $|T \cap P| - \frac{1}{2}\binom{n}{2}$ may be considered as a profit of a ranking P . It is not difficult to find a ranking with the profit cn (see [6]). In this note we present a polynomial time procedure that constructs a ranking with profit at least $c_1 n^{3/2}$, and thus ensures the lower bound given by (1). The same approach is applied to a related problem. Given a graph, we construct a subset of vertices with "large" discrepancy.

Let X_i , $1 \leq i \leq n$, be mutually independent random variables with $\Pr[X_i = +1] = \Pr[X_i = -1] = \frac{1}{2}$. The following lemma, Lemma 1, is well known.

LEMMA 1. $E[|X_1 + \dots + X_n|] \geq c_3 n^{1/2}$, $c_3 = (2/\pi)^{1/2}$.

* Received by the editors May 15, 1987; accepted for publication (in revised form) December 28, 1987.

† Faculty of Mathematics and Physics, Department of Applied Mathematics, Charles University, Malostranské 25, 118 00 Praha 1, Czechoslovakia.

‡ Faculty of Nuclear Engineering, Department of Mathematics, Technical University, Husova 5, 110 00 Praha 1, Czechoslovakia.

§ Department of Mathematics, State University of New York, Stony Brook, New York 11794.

Proof. Let $\mathbf{1} = (1, 1, \dots, 1)$ be the vector of the length n , and let $t = \lfloor n/2 \rfloor$. We have

$$\begin{aligned} 2^n \cdot E[|X_1 + \dots + X_n|] &= \sum_{\mathbf{x} \in \{-1, 1\}^n} |\mathbf{1} \cdot \mathbf{x}| = 2 \sum_{k=0}^t (n-2k) \binom{n}{k} \\ &= 2 \left(n + \sum_{k=1}^t (n-2k) \binom{n}{k} \right) = 2n + 2n \sum_{k=1}^t \left(\binom{n-1}{k} - \binom{n-1}{k-1} \right) \\ &= 2n \binom{n-1}{t}. \end{aligned}$$

Using the Stirling formula, we obtain

$$2n \binom{n-1}{t} > \left(\frac{2}{\pi} \right)^{1/2} n^{1/2} \cdot 2^n. \quad \square$$

Let $\|\mathbf{v}\| = \sum_{i=1}^k |v^i|$ denote the one-norm of a vector $\mathbf{v} = (v^1, \dots, v^k)$.

LEMMA 2. Let $\mathbf{v}_1, \dots, \mathbf{v}_n \in \{-1, 1\}^k$ be arbitrary vectors. Then there are signs $\varepsilon_1, \dots, \varepsilon_n \in \{-1, 1\}$ such that

$$\|\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_n \mathbf{v}_n\| \geq c_3 k n^{1/2}.$$

Proof. Let $\varepsilon_1, \dots, \varepsilon_n$ be independent random variables with

$$\Pr[\varepsilon_i = +1] = \Pr[\varepsilon_i = -1] = \frac{1}{2}.$$

Set $\mathbf{v}_i = (a_{i1}, \dots, a_{ik})$ and set $L_j = \varepsilon_1 a_{1j} + \dots + \varepsilon_n a_{nj}$, $1 \leq j \leq k$, so that $\varepsilon_1 v_1 + \dots + \varepsilon_n v_n = (L_1, \dots, L_k)$. Regardless of the value a_{ij}

$$\Pr[\varepsilon_i a_{ij} = +1] = \Pr[\varepsilon_i a_{ij} = -1] = \frac{1}{2},$$

and for j fixed the $\varepsilon_i a_{ij}$ are mutually independent. By Lemma 1, therefore, $E[|L_j|] \geq c_3 n^{1/2}$. Linearity of expectation gives

$$\begin{aligned} E[\|\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_n \mathbf{v}_n\|] &= E[|L_1| + \dots + |L_k|] \\ &= E[|L_1|] + \dots + E[|L_k|] \\ &\geq k c_3 n^{1/2}. \end{aligned}$$

There must be a specific set of values $\varepsilon_1, \dots, \varepsilon_n$ so that the random variable $\varepsilon_1 v_1 + \dots + \varepsilon_n v_n$ exceeds or equals its expectation; these values satisfy Lemma 2. $\square$

The existential nature of the above proof is typical of the probabilistic method. There are no logical difficulties as the probability space is finite (the 2^n possible sequences $(\varepsilon_1, \dots, \varepsilon_n)$) and probabilistic methods could be rephrased as counting arguments. From the modern viewpoint of computational complexity such proofs are not fully satisfactory as the probability space is exponential in size. We now give a polynomial time algorithm for finding the $\varepsilon_1, \dots, \varepsilon_n$, the existence of which is guaranteed by Lemma 2.

THEOREM 3. There is an $O(n^3 \log n)$ algorithm that for given vectors $\mathbf{v}_1, \dots, \mathbf{v}_n \in \{-1, 1\}^k$, $k = O(n)$, computes $\varepsilon_1, \dots, \varepsilon_n \in \{-1, 1\}$ so that

$$\|\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_n \mathbf{v}_n\| \geq c_3 k n^{1/2}.$$

Proof. Set

$$(2) \quad W(\mathbf{s}, i) = \sum_{j=0}^i \binom{i}{j} \sum_{t=1}^k |s^t + i - 2j|$$

for a vector $\mathbf{s} = (s^1, \dots, s^k) \in Z^k$.

The signs $\varepsilon_n, \dots, \varepsilon_1$ (for notational convenience in the reversed order) are computed by the following procedure.

Procedure SIGNS

Begin

$\mathbf{s} := (0, \dots, 0);$

For $i := n, n-1, \dots, 1$ *Do*

Begin

If $W(\mathbf{s} + \mathbf{v}_i, i-1) \geq W(\mathbf{s} - \mathbf{v}_i, i-1)$ *Then* $\varepsilon_i := 1$ *Else* $\varepsilon_i := -1$;

$\mathbf{s} := \mathbf{s} + \varepsilon_i \mathbf{v}_i$

End

End

We first prove the correctness of the procedure. To motivate the definition of the function W let $\varepsilon_1, \dots, \varepsilon_i$ be independent random variables, each $+1$ or -1 with probability $\frac{1}{2}$, and set $\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_i \mathbf{v}_i + \mathbf{s} = (L_1, \dots, L_k)$. Then $L_t = \varepsilon_1 a_{1t} + \dots + \varepsilon_i a_{it} + s'_t$ and, regardless of the specific values $a_{1t}, \dots, a_{it}$

$$\Pr[L_t = s'_t + i - 2j] = \binom{i}{j} 2^{-i}$$

as this event occurs precisely when j of the $\varepsilon_k a_{kt}$ are -1 . Thus

$$\begin{aligned} E[\|\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_i \mathbf{v}_i + \mathbf{s}\|] &= E[|L_1|] + \dots + E[|L_k|] \\ &= 2^{-i} W(\mathbf{s}, i). \end{aligned}$$

Now let $\mathbf{s}_i, \mathbf{s}_{i-1}$ be the value of $\mathbf{s}$ in the beginning and the end of the i th iteration, respectively. We have

$$W(\mathbf{s}_i, i) = 2^i E[\|\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_i \mathbf{v}_i + \mathbf{s}_i\|]$$

so that, splitting the expectation according to the value of ε_i ,

$$W(\mathbf{s}_i, i) = W(\mathbf{s}_i + \mathbf{v}_i, i-1) + W(\mathbf{s}_i - \mathbf{v}_i, i-1).$$

Thus $W(\mathbf{s}_{i-1}, i-1) \geq W(\mathbf{s}_i, i)/2$ by the choice of ε_i in the procedure. By Lemma 2 we have

$$\begin{aligned} \|\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_n \mathbf{v}_n\| &= W(\mathbf{s}_0, 0) \geq \frac{1}{2} W(\mathbf{s}_1, 1) \geq \dots \geq 2^{-n} W(\mathbf{s}_n, n) \\ &= E[\|\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_n \mathbf{v}_n\|] \geq c_3 k n^{1/2}. \end{aligned}$$

We estimate the complexity of the algorithm. We assume $k = O(n)$ as only such values of k will appear in our applications. All of the combinatorial coefficients $\binom{j}{i}$, $n \geq i \geq j \geq 0$, can be computed in advance using the identity

$$\binom{i}{j} = \binom{i-1}{j} + \binom{i-1}{j-1}$$

and then stored for later usage. This requires $O(n^2)$ arithmetic operations on numbers of $O(n)$ bits. One evaluation of $W(\mathbf{s}, i)$ can be performed with $O(n)$ arithmetic operations on numbers of $O(n)$ and $O(\log n)$ bits. It is not difficult to conclude that the total complexity of the procedure can be estimated by $O(n^3 \log n)$ time. $\square$

Let $G = (V, E)$ be a graph. Consider a function $f: [V]^2 \rightarrow \{1, -1\}$ defined by

$$f(e) = \begin{cases} 1 & \text{if } e \in E, \\ -1 & \text{if } e \notin E. \end{cases}$$

For a subset $S \subset V$, set $f(S) = \sum \{f(e) \mid e \in [S]^2\}$. We call $|f(S)|$ the discrepancy of S .

THEOREM 4. For every graph G there is a subset $S \subset V(G)$ with discrepancy at least $c_4 n^{3/2}$, where $c_4 = (1/24)\pi^{-1/2}$ and $n = |V(G)|$. Moreover, there is an $O(n^3 \log n)$ algorithm that finds such a set S .

The existence of a set S with the property was proved in [1]. The merit of the present approach is that an explicit S is found. Note also that the constant c_4 is somewhat better than the corresponding one from [1].

Proof. Let $X \cup Y$ be a partition of G satisfying $|X| = \lfloor n/2 \rfloor$ and $|Y| = \lceil n/2 \rceil$ ($\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ denote the lower and upper integer approximations, respectively). Set $n_0 = \lfloor n/2 \rfloor$, $k = \lceil n/2 \rceil$, $X = \{x_1, \dots, x_{n_0}\}$, and $Y = \{y_1, \dots, y_k\}$. Define vectors $\mathbf{v}_1, \dots, \mathbf{v}_{n_0}$ by

$$v_i^j = \begin{cases} +1 & \text{if } (x_i, y_j) \in E, \\ -1 & \text{if } (x_i, y_j) \notin E. \end{cases}$$

Now we apply Theorem 3 with n_0 and k . We obtain signs $\varepsilon_1, \dots, \varepsilon_{n_0}$ such that

$$(3) \quad \|\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_{n_0} \mathbf{v}_{n_0}\| \geq c_3 k n_0^{1/2} \geq c_3 \left(\frac{n}{2}\right)^{3/2}.$$

We partition the sets X and Y into $X = X^+ \cup X^-$ and $Y = Y^+ \cup Y^-$ where $X^+ = \{x_i \mid \varepsilon_i = 1\}$ and $Y^+ = \{y_j \mid \sum_{i=1}^{n_0} \varepsilon_i f(x_i, y_j) \geq 0\}$. We have

$$(4) \quad \begin{aligned} \|\varepsilon_1 \mathbf{v}_1 + \dots + \varepsilon_{n_0} \mathbf{v}_{n_0}\| &= \sum_{j=1}^k |\sum_{i=1}^{n_0} \varepsilon_i f(x_i, y_j)| \\ &= \sum_{\substack{y \in Y^+ \\ x \in X^+}} f(x, y) + \sum_{\substack{y \in Y^+ \\ x \in X^-}} -f(x, y) + \sum_{\substack{y \in Y^- \\ x \in X^+}} -f(x, y) + \sum_{\substack{y \in Y^- \\ x \in X^-}} f(x, y). \end{aligned}$$

It follows that there are $X^* \in \{X^+, X^-\}$ and $Y^* \in \{Y^+, Y^-\}$ such that

$$|f(X^*, Y^*)| \geq \frac{1}{4} c_3 \left(\frac{n}{2}\right)^{3/2} = c_1 n^{3/2} \quad \text{where } f(X^*, Y^*) = \sum_{\substack{x \in X^* \\ y \in Y^*}} f(x, y).$$

We finish our proof with the same argument as in [1]. We have

$$f(X^*, Y^*) = f(X^* \cup Y^*) - f(X^*) - f(Y^*).$$

Let S be one of the sets X^* , Y^* , $X^* \cup Y^*$ for which $|f(S)| \geq \frac{1}{3} f(X^*, Y^*)$. Thus we have constructed a subset S with

$$|f(S)| \geq \frac{1}{3} \cdot \frac{1}{4} c_3 \left(\frac{n}{2}\right)^{3/2} = c_4 n^{3/2}. \quad \square$$

THEOREM 5. Let T be a tournament on n vertices. Then there is a ranking P with fit at least $\frac{1}{2} \binom{n}{2} + c_1 n^{3/2}$ where $c_1 = \frac{1}{8} \pi^{-1/2}$. The ranking P can be constructed in $O(n^3 \log n)$ time.

Proof. For $1 \leq i < j \leq n$ define

$$f(i, j) = \begin{cases} 1 & \text{if } (i, j) \in T, \\ -1 & \text{if } (j, i) \in T. \end{cases}$$

Let X^*, Y^* be the sets constructed in the proof of Theorem 4 so that, reversing the roles of X^*, Y^* if necessary,

$$f(X^*, Y^*) > c_1 n^{3/2}.$$

Set $R = \{1, \dots, n\} - X^* - Y^*$. Trying an arbitrary ranking and its inverse, as discussed in the Introduction, we may find rankings P_{X^*}, P_{Y^*}, P_R on X^*, Y^*, R , respectively, with nonnegative profit.

Suppose $f(X^* Y^*, R) \geq 0$. We define a ranking P on $1, \dots, n$ by

$$xPy \quad \text{for all } x \in X^*, \quad y \in Y^*,$$

$$zPr \quad \text{for all } z \in X^* \cup Y^*, \quad r \in R,$$

$$xPx' \quad \text{iff } xP_{X^*}x', \quad x, x' \in X^*,$$

$$yPy' \quad \text{iff } yP_{Y^*}y', \quad y, y' \in Y^*,$$

$$rPr' \quad \text{iff } rP_Rr', \quad r, r' \in R.$$

P has nonnegative profit on X^*, Y^*, R and $(X^* \cup Y^*) \times R$ and profit at least $c_1 n^{3/2}$ on $X^* \times Y^*$ so $|T \cap P| - \frac{1}{2} \binom{n}{2} \geq c_1 n^{3/2}$.

When $f(X^* Y^*, R) \leq 0$ we change P by defining

$$rPz \quad \text{for all } z \in X^* Y^*, \quad r \in R,$$

and the same conclusion holds. $\square$

REFERENCES

- [1] P. ERDŐS AND J. SPENCER, *Probabilistic Methods in Combinatorics*, Academic Press (Akadémiai Kiadó), New York, 1974.
- [2] A. IQBAL, W. D. COOK, AND M. KRESS, *On the minimum violations ranking of a tournament*, Management Sci., 32 (1986), pp. 660–672.
- [3] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability. A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, 1979.
- [4] M. GRÖTSCHEL, M. JÜNGER, AND G. REINELT, *A cutting plane algorithm for the linear ordering problem*, Oper. Res., 32 (1984), pp. 1195–1220.
- [5] J. K. LENSTRA, *Sequencing by Enumerative Methods*, Mathematical Center Tracts 69, Mathematisch Centrum, Amsterdam, 1977.
- [6] S. POLJAK AND D. TURŽIK, *A polynomial time heuristic for certain subgraph optimization problems with guaranteed lower bound*, Discrete Math., 58 (1986), pp. 99–104.
- [7] J. SPENCER, *Optimal ranking of tournaments*, Networks, 1 (1971), pp. 135–138.
- [8] ———, *Nonconstructive methods in discrete mathematics*, in Studies in Combinatorics, G. C. Rota, ed., Mathematical Association of America, Washington, DC, 1978, pp. 142–178.
- [9] W. F. DE LA VEGA, *On the maximal cardinality of a consistent set of arcs in a random tournament*, J. Combin. Theory Ser. B, 35 (1983), pp. 328–332.

IMPROVED COMPLEXITY BOUNDS FOR CENTER LOCATION PROBLEMS ON NETWORKS BY USING DYNAMIC DATA STRUCTURES*

ARIE TAMIR†

Abstract. Several center location problems on general networks and even on tree networks are nonconvex. Known solution procedures are based on decomposing the problem into subproblems, finding each one of the local solutions independently, and then selecting among them the best solution. The purpose of this paper is to demonstrate that for some location models, it is possible to take advantage and use information that is common to certain subproblems in order to obtain better complexity bounds. The main tool implemented is a dynamic data structure that is used to maintain the objective (dynamically) as we move from one subproblem to the next.

This study focuses on (nonconvex) obnoxious center location problems on trees, and on the classical p -center problem on general networks. In both cases better complexity bounds are derived.

Key words. location theory, network center problems, dynamic data structures

AMS(MOS) subject classifications. 05C35, 90C27, 68R10, 68U05

1. Introduction. Let $G = (V, E)$ be an undirected graph with node set V and edge set E . Let $|V| = n$ and $|E| = m$. Each edge has a positive length and is assumed to be rectifiable. Thus, we refer also to interior points on an edge by their distances (along the edge) from the two nodes of the edge. We let $A(G)$ denote the continuum set of points on the edges of G .

The edge lengths induce a distance function on $A(G)$. For any x, y in $A(G)$, $d(x, y)$ will denote the length of a shortest path connecting x and y . Also, if Y is a subset of $A(G)$ we define the distance of x from Y by

$$d(x, Y) = \inf_{y \in Y} d(x, y).$$

Assume that each node v in V is associated with a positive weight w_v .

For each x in $A(G)$ we define

$$(1.1) \quad F(x) = \max_{v \in V} \{w_v d(x, v)\},$$

$$(1.2) \quad G(x) = \min_{v \in V} \{w_v d(x, v)\}.$$

A (weighted) *one-center* of the graph G is any point x^* in $A(G)$ at which the function F is minimized. A (weighted) *obnoxious one-center* is any point x^* in $A(G)$ at which the function $G(x)$ is maximized. Both problems have extensions to the multifacility case that will be considered later.

Focusing on (1.1) first we note that this problem is not convex on a general graph, and thus may possess local solutions in $A(G)$ that are not global. In fact, the most efficient known procedure to solve this problem [KH1] is based on calculating all local solutions. This algorithm considers each edge independently and finds the best local solutions on that edge. Even when restricted to an edge, the function $F(x)$ may not be convex. However, $F(x)$ is continuous and piecewise linear, and it has at most $3n - 2$ breakpoints on each

* Received by the editors January 14, 1987; accepted for publication (in revised form) March 9, 1988.

† Department of Statistics and Operations Research, New York University, New York, New York 10003 and Department of Statistics, Tel Aviv University, Tel Aviv, Israel.

edge [KH1]. By using a standard divide-and-conquer approach, all these break-points (and in particular the local minima) can be found in $O(n \log n)$ time [At], [KH1], [HS].

Since the algorithm in [KH1] considers each edge separately, the total effort of finding the one-centers of G is $O(mn \log n)$. (This bound also includes the effort of calculating all the internodal distances.)

A key question is whether we can improve the above bound by exploiting possible relations between local solutions corresponding to different edges.

When the graph is a tree the function $F(x)$ is convex on each path of $A(G)$ [DFL]. Therefore, each local solution is also global. There is a unique optimal solution that can be found in linear time by utilizing the convexity of $F(x)$ [Me].

Turning next to the obnoxious center problem (1.2), we note that it may have local solutions that are not global even when the graph is a tree.

The function $G(x)$ is not, in general, concave on each path. (Notice that concavity is a desirable property since we maximize $G(x)$.) However, when we restrict ourselves to any edge of a general graph, $G(x)$ is concave. Each one of its components, $w_v d(x, v)$, $v \in V$, is a minimum of two linear functions [Ha]. Therefore, $G(x)$ is a minimum of $2n$ linear functions. It has a unique maximum point on each edge, which can be found in $O(n)$ time by using the linear programming methods of [Dy] and [Me].

Again, if we ignore possible relations between subproblems defined on different edges and solve all of them independently, we find all the global obnoxious centers in $O(mn)$ time. (We must also add the effort to compute all internodal distances, which is $O(mn + n^2 \log n)$ [FT] for general graphs and $O(n^2)$ for trees and planar graphs [Fr].)

Our main goal is to demonstrate that for several graph location models, it is possible to take advantage and use information that is common to certain subproblems in order to obtain better complexity bounds.

The main tool we use is a dynamic data structure that we use to maintain the objective (dynamically) as we move from one subproblem to the next.

In the first part of this paper, we discuss the obnoxious one-center problem on tree graphs and improve significantly its current complexity bound. We then use our result for the one-center problem to solve an obnoxious multifacility problem defined in [DW]. Again, we improve upon the best known bounds.

In the second part of the paper, we demonstrate the usefulness of dynamic data structures in solving p -center problems on general graphs. We first solve the two-center model, improving the bounds obtained previously in [KH1] and [Mo]. This result is then used to reduce the bound for a general p .

2. Obnoxious facilities on tree graphs. In this section we discuss the obnoxious one-center problem on a tree graph. The extension to the multifacility case is considered in the next section.

As mentioned above the objective function $G(x)$, defined in (1.2), may have local maxima that are not global. However, when we restrict our search to an edge the subproblem is concave and has a unique maximum.

To solve the problem globally our idea is to do some preprocessing and then find an order on the edges, according to which the respective subproblems will be solved. With this ordering it will be possible to solve a given subproblem by extensively utilizing information about subproblems that have already been solved. On the average the marginal effort to solve a subproblem will be significantly less than the effort needed to solve it independently.

To motivate our approach for a general tree we first consider two cases: the path tree and the star tree. Both will be utilized in the general case.

The path tree. In this model the tree consists of a single path. We may assume without loss of generality that the nodes in V are points on the real line. Thus, suppose that $V = \{v_1, \dots, v_n\}$ and $v_1 < v_2 < \dots < v_n$. Also, let $N = \{1, 2, \dots, n\}$. The function $G(x)$, defined on the interval $[v_1, v_n]$, is given by

$$(2.1) \quad G(x) = \min_{1 \leq i \leq n} \{w_{v_i} |v_i - x|\}.$$

We will now solve the subproblems on the edges following the natural ordering; i.e., first we find the maximum of $G(x)$ on $[v_1, v_2]$, then we find its maximum on $[v_2, v_3]$, and so on. Let $G^i(x)$ denote the restriction of $G(x)$ to the interval (edge) $[v_i, v_{i+1}]$. In particular, $G^1(x)$, $x \in [v_1, v_2]$ is given by

$$(2.2) \quad G^1(x) = \min [w_{v_1}(x - v_1), \min_{2 \leq j \leq n} \{w_{v_j}(v_j - x)\}].$$

Suppose we use (2.2) to define $G^1(x)$ not only on the interval $[v_1, v_2]$ but on the entire real line. $G^1(x)$ is the minimum of n linear functions. The key observation is that $G^2(x)$ is obtained from $G^1(x)$ by deleting the linear function $w_{v_2}(v_2 - x)$, and inserting, instead, its negation, the function $w_{v_2}(x - v_2)$. Such a step is called a *pivot*. Of course, for our purposes, only the domain $[v_2, v_3]$ is relevant to the optimization of $G^2(x)$. This observation can clearly be extended. For each $j = 2, \dots, n$, $G^j(x)$ is obtained from $G^{j-1}(x)$ by a deletion of $w_{v_j}(v_j - x)$ and an insertion of the function $w_{v_j}(x - v_j)$. Thus, the function we maximize in the j th subproblem is obtained from the preceding one by one pivot. What we need is a data structure that will enable us to execute this pivot efficiently. Indeed, the dynamic data structure in [OV] is quite suitable.

The data structure in [OV] was designed to dynamically maintain the intersection of a set of halfplanes in the xy -plane. Clearly the graph of each function $G^j(x)$ can be viewed as the boundary of the intersection of n halfplanes corresponding to the n linear functions that define $G^j(x)$.

To help the reader gain some general understanding of the idea behind the above data structure we provide a short description of it (modified for our purposes).

Assume that the linear functions that are initially in the set are stored at the leaves of a balanced binary tree T , using their slopes as the sorting key. Each internal node α of T will store the pointwise minimum function of the set of linear functions corresponding to leaves in its subtree. This pointwise minimum function is represented by a concatenable queue, Q_α , which can be computed efficiently from the queues Q_γ and Q_δ "stored" at the two sons γ and δ of α . Generally, it is easy to check that Q_α is obtained from Q_γ and Q_δ by taking a "left" piece of the first and a "right" piece of the second. If we want to build Q_α from Q_γ and Q_δ and retain Q_γ and Q_δ as they are, the updating process will turn out to be too expensive. The main idea in [OV] is to cut off the portions of Q_γ and Q_δ required for the construction of Q_α and pass them on to α , leaving γ and δ with only a fragment of their original associated structure.

Thus, the binary tree T is augmented to a tree T^* , in which an internal node α has associated with it the left or right portion of Q_α , say Q_α^* , that was not used to construct the pointwise minimum function (as a concatenable queue) at its father node. The pointwise minimum function corresponding to the entire collection of linear functions will be available at the root of T^* .

It is shown in [OV] that with this data structure we can go down in the tree and reassemble the Q -structure at the nodes bordering a path from the pieces reclaimed by the continued splitting of the Q_γ or Q_δ on the way down, and later climb back along the same path meanwhile reconstructing the Q_α -structure for each node α visited, and passing on the part needed as the climb continues to its father. It is proved in [OV] that the effort to perform this task of traversing a path is $O(\log n)$ per node visited.

The above results are summarized in the following theorem taken from [OV].

THEOREM 2.1. *In the xy -plane the intersection of a set of n halfplanes can be computed in $O(n \log n)$ time and $O(n)$ space. This intersection can then be maintained with $O(\log^2 n)$ time per insertion or deletion of a halfplane such that the adjacent edges of a given edge along the intersection boundary are available in constant time (n denotes the current number of halfplanes stored).*

Each function $G^j(x)$ can be viewed as the boundary of the intersection of the n halfplanes corresponding to the n linear functions that define $G^j(x)$. When we use the above data structure to store the boundary, it will allow for binary searches over the boundary in only $O(\log n)$ time [OV].

With the above theorem we can now state the following. After a preprocessing of $O(n \log n)$ time, ($j = 1$), it takes $O(\log^2 n)$ time to obtain $G^j(x)$ from $G^{j-1}(x)$. In particular, the incremental effort to compute the maximum of $G(x)$ over the edge $[v_{j-1}, v_j]$ is $O(\log^2 n)$. Therefore, all obnoxious one-centers are computed in $O(n \log^2 n)$ time and $O(n)$ space. (The bound quoted in [DW] for this problem on a path is $O(n^3)$). Also note that we have already shown in the Introduction how to solve this problem on a general planar graph in $O(n^2)$ time, by solving $O(n)$ subproblems in linear time.)

There is a global approach for this problem defined on a path tree that yields an $O(n \log n)$ bound. However, this latter approach does not seem to extend to a general tree. For the sake of the record we now briefly describe the $O(n \log n)$ time algorithm.

From (2.1) we note that the function $G(x)$, assumed now to be defined on the entire real line, is the pointwise minimum function of n (convex) functions. The i th function has a minimum at $x = v_i$. It is easy to verify that each pair in this collection of n functions intersects at two points at the most. Thus, from [At], [HS], $G(x)$ has at most $2n - 2$ breakpoints corresponding to intersection points of pairs of functions in the collection. (This result is related to Davenport-Schinzel sequences of order two [HS].) Since each function in the collection has one breakpoint, the total number of breakpoints of the piecewise linear function $G(x)$ is at most $3n - 2$. Using a standard divide-and-conquer approach [At], [HS] we can (recursively) compute the breakpoints of the minimum function of the subcollection $\{w_{v_i}|v_i - x|\}$, $i = 1, \dots, [n/2]$, and the breakpoints of the minimum function of the subcollection $\{w_{v_i}|v_i - x|\}$, $i = [n/2] + 1, \dots, n$. A merging procedure will then yield all the breakpoints of $G(x)$. Since $G(x)$ has $O(n)$ breakpoints the divide-and-conquer approach will take a total of $O(n \log n)$ time to find them. The maximum point of $G(x)$ in $[v_1, v_n]$ is one of these breakpoints.

The star tree. A star tree is a tree that has at most one node, its center, which has a degree greater than one. Let $V = \{v_1, \dots, v_n\}$ and suppose that v_1 is the center. For each $j = 2, \dots, n$, let a_j be the length of the edge $[v_1, v_j]$. For convention $a_1 = 0$.

Let x be a point on edge $[v_1, v_j]$. (x will indicate the distance of this point from the center v_1 .) Let $G^j(x)$ denote the restriction of $G(x)$ in (1.2) to the edge $[v_1, v_j]$:

$$(2.3) \quad G^j(x) = \min [w_{v_j}(a_j - x), \min_{\substack{1 \leq i \leq n \\ i \neq j}} \{w_{v_i}(x + a_i)\}].$$

Using (2.3) suppose that $G^j(x)$ is defined for all x on the real line. (Only the range $0 \leq x \leq a_j$ is of interest for maximizing $G^j(x)$.)

For each $j = 2, \dots, n$, $G^j(x)$ is a minimum of n linear functions. Therefore, it is piecewise linear and concave. Moreover, $G^{j+1}(x)$ can be obtained from $G^j(x)$ by two pivots. First, delete the function $w_{v_j}(a_j - x)$ and insert, instead, the function $w_{v_j}(a_j + x)$. Second, delete the function $w_{v_{j+1}}(a_{j+1} + x)$ and insert, instead, the function $w_{v_{j+1}}(a_{j+1} - x)$. From Theorem 2.1, these two pivots can be performed in

$O(\log^2 n)$ time. The incremental effort to compute the maximum of $G(x)$ over $[v_1, v_j]$ is $O(\log^2 n)$. Therefore, the obnoxious one-center of a star tree is computed in $O(n \log^2 n)$ time and $O(n)$ space.

Again, as in the case of a path tree, we have an ad hoc $O(n \log n)$ procedure for the obnoxious one-center of a star, which does not seem to be extendible to a general tree. We now briefly sketch this ad hoc approach.

Consider the following collection of n linear functions: $\{w_{v_i}(x + a_i)\}_{i=1}^n$. Let $H^1(x)$ be the first hull of the collection, i.e.,

$$(2.4) \quad H_1(x) = \min_{1 \leq i \leq n} \{w_{v_i}(x + a_i)\}.$$

Similarly, let $H^2(x)$ be the second hull of the above collection, i.e., at each x , $H^2(x)$ is defined as the value of the second smallest function at x . (See [CSY] for results concerning k th hulls.)

We first compute both hulls using, for example, the $O(n \log n)$ scheme given in [CSY]. To complete the preprocessing we also merge the lists of breakpoints of $H^1(x)$ with that of $H^2(x)$.

To show that an obnoxious one-center can be found in $O(n \log n)$ time, it will suffice to demonstrate that the maximum of each function G^j , defined above, can now be obtained in $O(\log n)$ time.

Consider some function G^j . To calculate $G^j(x)$ for some x we simply note that the function attaining the inner minimum in (2.3) is always either on the first or second hull. Thus, to find the maximum of G^j we run a binary search through the merged list of the breakpoints of $H^1(x)$ and $H^2(x)$. Clearly, each step of this search can be executed in constant time. This completes the description of an $O(n \log n)$ time algorithm for the star tree.

An interesting observation, followed from the above algorithm, is that the total number of distinct breakpoints of all the concave functions $G^j(x)$, $1 \leq j \leq n$, is $O(n)$.

We now turn to the case of a general tree.

The general tree. The goal is to find the order according to which the subproblems corresponding to the edges of the tree will be solved. As mentioned above with respect to the path and star trees, the objective is to keep the total number of pivots as small as possible.

We will use the dynamic data structure of [OV], mentioned in Theorem 2.1, to construct and maintain dynamically the function $G(x)$ of (1.2), as we move along the tree from one edge to another. Recall from above that the restriction of $G(x)$ to an edge is a piecewise linear concave function; i.e., $G(x)$ is the pointwise minimum of n linear functions.

Suppose that the tree T is rooted at some arbitrary node r . We now define the following sets. Let v be a node of T . $D(v)$, the set of descendants of v , is defined as the set consisting of v and all nodes having v on the unique path connecting them to the root r . $S(v)$, the set of sons of v , is defined as the set consisting of all nodes u , $u \neq v$, having v as their closest node on the path connecting them to the root. v is also called the father of its sons. If u and v are two distinct nodes and u is in $D(v)$ then u is called a *proper descendant* of v . This definition introduces a partial ordering on the nodes of T , with the root r as a maximum element of this ordering.

Let $V^1 \subseteq V$ be the set of nodes consisting of the root r and all nodes v with $|S(v)| \geq 2$. A node in V^1 will be called a *blessed father* or a *blessed node*. We show that if we start at the root and visit the edges according to a depth first search [AHU],

the total number of pivots needed to generate the reductions of $G(x)$ to all edges is

$$O\left(\sum_{v \in V^1} |D(v)|\right).$$

Note that the latter bound is linear for path trees and star trees, regardless of the root choice.

To formalize our claim we need some definitions. Throughout the process we use the data structure of [OV] (Theorem 2.1) where we always store exactly n linear functions defined on the entire real line. Each node v of V will contribute one such function, the slope of which is either w_v or $-w_v$. The data structure will dynamically maintain the pointwise minimum of these n functions. A pivot is the operation where a given linear function (currently in storage) is replaced by another linear function corresponding to the same node. A pivot amounts to one deletion and one insertion, and therefore can be implemented in $O(\log^2 n)$ time (Theorem 2.1).

Let v be a blessed node. We say that the data structure is *conformal* with v if there exists a real number a_v such that the n functions currently in storage are of the form $\{w_u(d(v, u) + a_v + x)\}$, $u \in V$. (x is the argument of the linear functions and a_v can be viewed as a translation coefficient.) To motivate this definition imagine that there is a new rectifiable (infinite) edge emanating from v . Consider a point on this edge the distance of which from v is y . Then the closest (over all nodes) weighted distance to that point is given by

$$\min_{u \in V} \{w_u(d(v, u) + a_v + (y - a_v))\}.$$

Thus, the data structure can be used efficiently to compute the function $G(x)$ along such an edge.

By using an appropriate sequence of pivots we will adjust the collection of functions in storage to correspond to a given edge. We will say that the data structure is currently *conformal* with a given edge, say $(v, u) \in E$, if there exists some interval of the real line of length $d(v, u)$, such that the pointwise minimum of the collection of functions currently in storage, restricted to this interval, represents the restriction of the function $G(x)$ in (1.2) to the edge (v, u) .

THEOREM 2.2. *Suppose that initially the data structure is conformal with r , the root of the tree. Then there exists a sequence of $O(\sum_{v \in V^1} |D(v)|)$ pivots such that for each edge of the tree there is a point in the sequence where the data structure is conformal with this edge.*

Proof. Suppose that we apply a depth first search to the tree starting at its root. Each edge will be traversed exactly twice. The data structure will be conformal with an edge when it is traversed for the second time.

For each blessed node v in V^1 , let T_v denote the minimal subtree containing the nodes in $D(v)$.

It is sufficient to prove the following claim.

CLAIM. *Let v be a blessed node of the tree. Suppose that the data structure (for the entire tree) is conformal with v . There exists a sequence of*

$$O\left(\sum_{u \in D(v) \cap V^1} |D(u)|\right)$$

pivots, such that for each edge of T_v there is a point in the sequence where the data structure is conformal with this edge. Moreover at the end of the sequence the data structure is again conformal with v .

Proof. We use an inductive proof based on the partial ordering of the tree. Using the above discussion on path trees and star trees, we observe that the claim is true for the lowest level of the induction. (See Fig. 2.1.)

Let v be a blessed node. Let $v_1, \dots, v_k$ be the set of maximal proper descendants in $(D(v) \cap V^1) - \{v\}$. ($k \geq 2$ if v is not the root.) In particular, for each $i = 1, \dots, k$, all the intermediate nodes on $P(v, v_i)$, the path connecting v with v_i , are not blessed, i.e., they are of degree two.

Without loss of generality suppose that v_i is visited before v_{i+1} , $i < k$, in the depth first search. The data structure is now conformal with v . Consider some node v_i . To make the data structure conformal with v_i , we choose $a_{v_i} = a_v - d(v, v_i)$; then it suffices to perform pivots on the functions, currently in storage, that correspond to the nodes in $D(v_i)$ as well as the nodes on $P(v, v_i)$, such that each such pivot amounts to a translation of some current function. Let u be one of these nodes. If u is on the path $P(v, v_i)$ its respective function will have to be translated (along the x axis) by $2d(v, u)$. If u is in $D(v_i)$ the translation should be by $2d(v, v_i)$.

In any case after $|P(v, v_i)| + |D(v_i)|$ pivots the data structure is conformal with v_i . By the induction hypothesis it takes

$$O\left(\sum_{u \in D(v_i) \cap V^1} |D(u)|\right)$$

additional pivots to ensure that each edge in T_{v_i} is conformal with the data structure.

Following the pivots performed so far, we next show the pivots required for the data structure to be conformal with the edges on the path $P(v, v_i)$. At this point the data structure is conformal with v_i . We first pivot all the linear functions corresponding to the nodes in $D(v_i)$, so that the data structure becomes conformal with the edge of $P(v, v_i)$, which is incident to v_i ; i.e., the data structure yields $G(x)$ for this edge. Let u be a node in $D(v_i)$ and suppose that the linear function, currently in storage, corresponding to u is given by $f_u(x) = w_u(x - b_u)$. Then $f_u(x)$ is replaced by $\tilde{f}_u(x) = -w_u(x - (b_u + 2d(u, v_i)))$; i.e., the function representing u will change its slope from w_u to $-w_u$ and then be translated (along the x axis) by $2d(u, v_i)$.

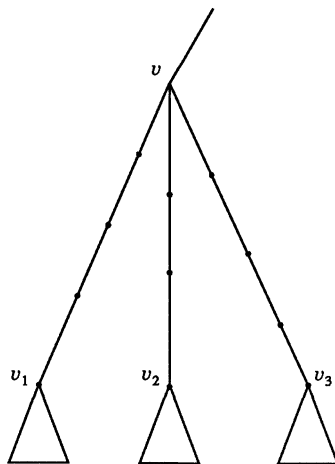


FIG. 2.1

After these $|D(v_i)|$ pivots are accomplished, we proceed from v_i to v along the path $p(v, v_i)$ exactly as we have done above for path trees. Thus $|P(v, v_i)|$ additional pivots are required for this phase. After the data structure is finally conformal with the closest edge to v on $P(v, v_i)$, we need $|P(v, v_i)| + |D(v_i)|$ additional pivots to make the data structure conformal with v again. These pivots are performed as follows. Let $u \neq v$ be a node in $D(v_i) \cup P(v, v_i)$, and suppose that the linear function, currently in storage, corresponding to u is given by $f_u(x) = -w_u(x - b_u)$. (Note that at this point the slopes of the functions corresponding to all nodes $u \neq v$ in $D(v_i) \cup P(v, v_i)$ are indeed negative.) Then $f_u(x)$ is replaced by

$$f_u^+(x) = w_u(x - (b_u + 2d(u, v))).$$

To summarize, a total of $O(\sum_{u \in D(v_i) \cap V^1} |D(u)|)$ pivots are needed to make the data structure conformal with all edges in the branch of T_v containing v_i . Summing the latter bound over all $i = 1, \dots, k$, we obtain the bound stated in the claim, i.e.,

$$O\left(\sum_{u \in D(v) \cap V^1} |D(u)|\right).$$

This completes the proof of the theorem.

To implement the scheme suggested by the above theorem we also must initialize the data structure so that it is conformal with r , the root of the tree. Applying Theorem 2.1 to the collection of linear functions $\{w_u(d(u, r) + x)\}$, $u \in V$, we note that this initialization step can be performed in $O(n \log n)$ time and $O(n)$ space.

The depth first search, and hence the bound on the number of pivots, depend on the root chosen. For example, for the tree in Fig. 2.2, rooting the tree at v yields a linear bound as opposed to an $\theta(n^{3/2})$ bound for rooting at u .

Suppose that the tree is rooted at some vertex v . Define

$$H(T, v) = \sum_{u \in V^1} |D(u)|.$$

From Theorem 2.2 it is desirable to root the tree at a node v for which $H(T, v)$ is minimized.

Given the tree T we first define T^1 , the homeomorphic reduction of T , as the tree obtained from T by deleting successively each node of T of degree two (and its two

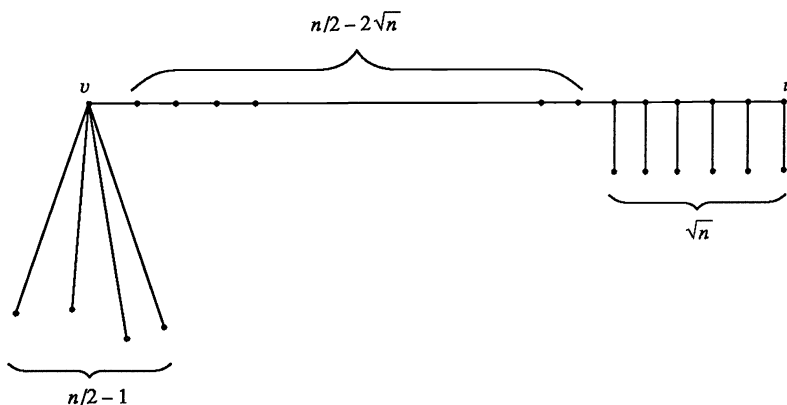


FIG. 2.2

edges), and connecting its two neighbors by a new edge. It is easy to see that the homeomorphic tree is independent of the sequence of node deletions.

Next we associate a weight f_u with each node u of T^1 . Let $N(u)$ denote the set of nodes in T^1 that are adjacent to u . For each x in $N(u)$ let $n(x, u)$ denote the number of intermediate nodes on the unique simple path connecting x and u in T , e.g., $n(x, u) = 0$ if and only if x and u are also adjacent in T . Define

$$f_u = 1 + \sum_{x \in N(u)} \frac{n(x, u)}{2}.$$

A node v in T^1 is a (weighted) centroid if the deletion of v disconnects T^1 into components none of which has a total node weight greater than one half of the total node weights of T^1 . It is well known that for positive weights every tree has either one or two centroids. Moreover, if the latter occurs then the two centroids are adjacent. The weighted centroid can be found in linear time [Go], [KH2].

We prove that a minimizer of $H(T, v)$ is a weighted centroid of T^1 , with respect to the weights $\{f_u\}$. Notice that by our definition the sum of the weights of all nodes in T^1 is n , the number of nodes in T .

THEOREM 2.3. *Suppose that $H(T, v) = \min_{u \in V} H(T, u)$. Then v is a weighted centroid of T^1 .*

Proof. Let v be a vertex of T with $H(T, v) = \min_{u \in V} H(T, u)$. Suppose first that v is of degree two. Let u and w be the two adjacent nodes of T^1 such that v is an intermediate node of the path $P(u, w)$ in T . Consider the two components of T obtained by deleting all the $n(u, w)$ intermediate nodes on the path $P(u, w)$. (See Fig. 2.3(a).) Let $n(u)(n(w))$ denote the number of nodes in the component containing $u(w)$. It is now easily seen that

$$H(T, u) - H(T, v) = [n(u, w) + n(w)] - [n(u) + n(u, w) + n(w)] = -n(u) < 0,$$

which contradicts the minimality of v . Thus, v is a node of T^1 .

Suppose that v is not a centroid, and let T_1^1 be the connected subtree of T^1 , obtained by deleting v , such that its total node weight is greater than $n/2$. Let v_1 be the closest node to v in T_1^1 . To complete the proof we show that $H(T, v) > H(T, v_1)$.

Consider the two components of T obtained by deleting all the $n(v, v_1)$ intermediate nodes on the path connecting v and v_1 . (See Fig. 2.3(b).) Let $n(v_1)(n(v))$ denote the number of nodes in the component containing $v_1(v)$. Then from the above $n(v_1) +$

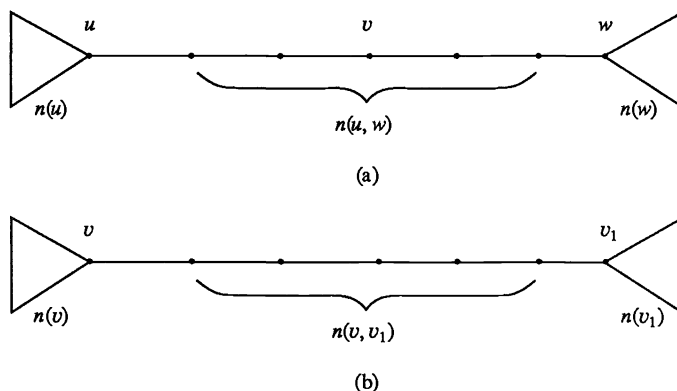


FIG. 2.3

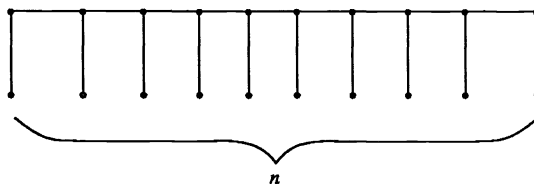


FIG. 2.4

$n(v, v_1)/2 > n/2$. Also, $n(v) + n(v, v_1) + n(v_1) = n$. Therefore, $H(T, v_1) - H(T, v) = n(v) - n(v_1) = n - (2n(v_1) + n(v, v_1)) < 0$, and the proof is complete.

A simple bound on $H(T) = \min_{u \in V} H(T, u)$ is given by the following result.

THEOREM 2.4. *Let T^1 be the homeomorphic reduction of T , and let k denote the diameter of T^1 . (k is the maximum number of nodes on a simple path in T^1 .) Then $H(T) = O(kn)$.*

Proof. Let v be some node of T^1 , and consider $H(T, v)$. Then each node of T is counted at most k times in the definition of $H(T, v)$.

The bound of Theorem 2.4 implies, for example, that $H(T)$ is linear for path trees and star trees. Also, for balanced trees [AHU], $H(T) = O(n \log n)$. Figure 2.4 provides an example of a tree for which $H(T) = \theta(n^2)$.

Using Theorem 2.1 we can now conclude that it takes $O(H(T) \log^2 n)$ time and $O(n)$ space to compute the $(n - 1)$ local maxima of the function $G(x)$, in (1.2), corresponding to the $(n - 1)$ edges of T . Therefore, all the obnoxious one-centers of the tree can be obtained in $O(H(T) \log^2 n)$ time and $O(n)$ space. We have already noted in the Introduction that there is another method to solve the problem on a general tree in $O(n^2)$ time, i.e., solving $(n - 1)$ subproblems each in linear time. Thus, when $H(T)$ is large, this might be the preferred method.

3. The multifacility model on general tree graphs. The data structure used above to solve the obnoxious one-center problem can also be applied to improve the complexity bound of a multifacility obnoxious location problem on general trees.

Given radii $\{r_v\}$, $v \in V$, and an integer p , the *multifacility obnoxious center problem* is

$$(3.1) \quad \begin{aligned} &\max_{\substack{X \subseteq A(T) \\ |X| = p}} G(X) \quad \text{where } G(X) = \min_{v \in V} \{w_v d(v, X)\}, \\ &\text{subject to } w_v d(v, X) \leq r_v \quad \text{for all } v \in V. \end{aligned}$$

To motivate the constraints $w_v d(v, X) \leq r_v$, $v \in V$, we can think of nuclear reactors, garbage depots, or ammunition storage areas as examples of obnoxious centers. Due to safety or ecological reasons, the desire is to locate the centers as far as possible from the customers, with respect to some objective function. On the other hand, transportation or other logistical considerations usually impose some upper bounds on the distance to the service centers. The latter aspect is captured by the constraints in (3.1).

This problem is defined and solved in [DW] for path trees only. We consider the case of a general tree.

A set $X \subseteq A(T)$, $|X| = p$, will be called a set of p (obnoxious)-centers. First we note that since no lower bounds are imposed on the distances among the p obnoxious centers themselves, we can assume without loss of generality that $p \leq n$.

Second, the optimal objective value of (3.1) is either equal to r_v for some $v \in V$, or it is equal to one of the $(n - 1)$ edge local maximum values of the one-dimensional

version of $G(X)$, as given by (1.2). Therefore, when we use the above data structure, it takes $O(H(T) \log^2 n)$ time to identify a set R of at most $n + (n - 1) = 2n - 1$ values containing the optimal objective value of (3.1).

Let r be an element of R , and consider the following obnoxious r -covering problem.

Find $p(r)$, the minimum value of p , and points $x_1, \dots, x_p$, such that

$$(3.2) \quad w_v d(v, x_i) \geq r \quad \text{for all } v \in V \quad \text{and} \quad i = 1, \dots, p$$

and

$$\min_{1 \leq i \leq p} \{w_v d(v, x_i)\} \leq r_v \quad \text{for all } v \in V.$$

The optimal objective value of (3.1) is the largest element r in R for which $p(r) \leq p$. Since $p(r)$ is monotone in r , we can use a binary search on R and obtain the optimal solution to (3.1) by solving $O(\log |R|) = O(\log n)$ obnoxious r -covering problems.

Solving the obnoxious r -covering problem. Suppose that a value of r is specified, and consider the constraints $w_v d(v, x_i) \geq r$ for $i = 1, \dots, p$. Using (1.2), these constraints can be stated as $G(x_i) \geq r$ for $i = 1, \dots, p$. Focusing on any edge of T , and using the concavity of the function G in (1.2) on this edge, we note that each obnoxious center is restricted to a subinterval of the (rectifiable) edge.

Suppose that the dynamic data structure used above is conformal with some edge. Then it is easy to see (due to the concavity of G), that it takes $O(\log n)$ time to find the two endpoints of the subinterval, which is consistent with the constraint $G(x) \geq r$. Therefore, given r , it takes $O(H(T) \log^2 n)$ time to find the endpoints of all such $(n - 1)$ subintervals.

Being restricted only to the above subintervals, the problem reduces to finding $p(r)$, the minimum value of p , and points $x_1, \dots, x_p$, in the union of the above subintervals, such that

$$\min_{1 \leq i \leq p} \{w_v d(v, x_i)\} \leq r_v \quad \text{for all } v \in V.$$

This latter problem can be solved in $O(n)$ time by the algorithm presented in [TZ]. We now give a brief description of the method in [TZ]. The method is based on first rooting the given tree at an arbitrary root. It then performs a one-pass of the tree in a bottom-up fashion, starting with the leaves of the original tree, and successively eliminating leaves of the current tree. The elimination step is done as follows. A node of the current tree, say x , with the property that all its sons are leaves of the current tree, is selected. The node and its sons are regarded as one cluster. A simple scan of the cluster will suffice to determine the minimum number of points needed to “cover” the leaves v of the cluster within their respective radii r_v / w_v . That immediately enables us to eliminate the entire cluster and replace it by a single node, say y , that will be directly connected by an edge to the father of x . y represents either a demand node with some updated radius, or an existing center that might be utilized in the next steps to cover other nodes of the tree. The computation effort is proportional to the size of the cluster, thus yielding an $O(n)$ algorithm for processing the entire tree.

We can now state that the multifacility obnoxious center problem is solvable in $O(H(T) \log^3 n)$ time on a general tree graph.

Since $H(T)$ is bounded between n and n^2 , the above complexity order can vary between $n \log^3 n$ and $n^2 \log^3 n$. In fact, if $H(T)$ is “large” we may prefer the following, more conventional, solution procedure that has an $O(n^2 \log n)$ complexity bound.

First we find the $(n - 1)$ edge local maximum points of $G(x)$. As mentioned in § 1, it takes $O(n)$ time to find each local solution using the linear programming algorithm in [Dy] and [Me]. Therefore, in $O(n^2)$ time we identify the set R containing the optimal objective value of (3.1). To locate the optimal value in R , we again solve $O(\log n)$ obnoxious r -covering problems. Each one of these covering problems can be solved in $O(n^2)$ time, since for each edge (independently) it takes $O(n)$ time to find the two endpoints of the subinterval consistent with the constraint $G(x) \geq r$. As mentioned above, it then takes $O(n)$ time to solve the r -covering problem when all the $O(n)$ endpoints are known (by the algorithms in [TZ]).

The above bound can be further improved when the tree is either a path or a star. Consider, for example, a path tree. In this case (see [DW]) we can assume without loss of generality that each one of the p obnoxious centers can be confined to a set of $2n + (n - 1) = 3n - 1$ possible locations. These are the $(n - 1)$ edge local maximum points of G in (1.2) and the $2n$ points x that solve some equation of the form $w_v d(v, x) = r_v$, $v \in V$.

Following the above discussion on path trees we can find these $(3n - 1)$ locations and their respective G values in a total of $O(n \log n)$ time.

Given a value of r , we can now solve the obnoxious r covering problem in $O(n)$ time. First we eliminate from our list of $(3n - 1)$ potential locations those points the G value of which is smaller than r , and then we use the linear time algorithm in [TZ].

This approach yields an $O(n \log n)$ time bound for the multifacility obnoxious center problem on a path tree. It significantly improves the $O(n^3)$ bound in [DW]. For star trees the $O(H(T) \log^3 n) = O(n \log^3 n)$ bound can be reduced to $O(n \log^2 n)$ using the algorithm (discussed in § 2) that solves the obnoxious one-center problem on star trees in $O(n \log n)$ time.

4. The p -center and r -covering problems on general graphs. In this section we demonstrate the usefulness of implementing dynamic data structures to improve the complexity bound of the classical p -center and r -covering location problems.

We extend the definition of F in (1.1) to the p -dimensional case as follows. Let $X \subseteq A(G)$ with $|X| = p$. Define

$$(4.1) \quad F(X) = \max_{v \in V} \{w_v d(v, X)\}.$$

Given an integer p the p -center problem is

$$\begin{aligned} \min \quad & F(X), \\ \text{subject to} \quad & |X| = p, \quad X \subseteq A(G). \end{aligned}$$

Given $r > 0$ the r -covering problem is to find a subset $X \subseteq A(G)$ of smallest possible cardinality, say $p = p(r)$, such that $F(X) \leq r$.

The r -covering problem is NP-hard even for the unweighted case ($w_v = 1$, $v \in V$) when $r = 1$ and the graph is a bipartite planar graph of maximum node degree four [KH1]. When p is arbitrary the p -center problem is NP-hard [KH1]. However, for a fixed value of p the problem can be solved polynomially.

Suppose that the (shortest) distances between all pairs of nodes are known. The algorithms in [KH1] can then solve the p -center problem in $O(m^p n^{2p-1} \log n)$ time. For the unweighted case the complexity bound is reduced in [KH1] to $O(m^p n^{2p-1})$. The algorithm in [Mo] has an $O(m^p n^{p+1} \log n)$ bound. By combining the main sub-routines of [KH1] and [Mo] we can obtain the improved bounds of $O(m^p n^p \log^2 n)$ and $O(m^p n^p \log n)$ for the weighted and unweighted cases, respectively. We will show

how a further improvement can be achieved by using dynamic data structures. We will first discuss the two-dimensional case, $p = 2$, and then use our results to improve the general bounds.

To solve the two-center problem the basic approach in [KH1] and [Mo] is to find all the $O(m^2)$ local solutions. A local two-center is a solution to the problem where the two centers must be located on a particular pair of distinct edges. Notice that we may assume that if an edge contains two centers they must appear at the two nodes of the edge. Thus, without loss of generality, we can always associate each one of the pair of points, constituting a local two-center, with a different edge. The $O(m^2)$ subproblems are solved independently. Focusing on the unweighted case each subproblem is solved in $O(n^3)$ and $O(n^3 \log n)$ times in [KH1] and [Mo], respectively. By implementing a dynamic data structure we will solve a subproblem in subquadratic time, i.e., $O(n \log^3 n)$.

We present the above subproblem in an extended form.

The two-dimensional model. Let $\{f_i(x_1)\}$, $1 \leq i \leq n$, and $\{g_i(x_2)\}$, $1 \leq i \leq n$, be two collections of one-dimensional piecewise linear concave functions, where $0 \leq x_1 \leq a$, $0 \leq x_2 \leq b$. Each of these functions is assumed to be of constant complexity. For each i , $1 \leq i \leq n$, define $h_i(x_1, x_2) = \min [f_i(x_1), g_i(x_2)]$. Let $h(x_1, x_2)$ be the upper envelope function of the collection $\{h_i(x_1, x_2)\}$, $1 \leq i \leq n$, i.e.,

$$(4.2) \quad h(x_1, x_2) = \max_{1 \leq i \leq n} \{h_i(x_1, x_2)\}.$$

The objective is to find a (global) minimum point of $h(x_1, x_2)$ in the rectangle $0 \leq x_1 \leq a$, $0 \leq x_2 \leq b$. One solution approach is to calculate the upper envelope, which is a piecewise linear two-dimensional surface, and then select the vertex ("breakpoint" of h) with the smallest h value. It has been shown recently in [PS] that the combinatorial complexity of the upper envelope of a collection of polyhedral functions, having k faces altogether, is $O(k^2 \alpha(k))$, where $\alpha(k)$ is the inverse Ackermann function. In particular, if each one of the $2n$ functions $\{f_i(x_1)\}$, $1 \leq i \leq n$, $\{g_i(x_2)\}$, $1 \leq i \leq n$, has at most s , (a constant number), one-dimensional faces, we obtain an $O(n^2 \alpha(n))$ complexity bound. The construction given in [PS] shows that the $O(n^2 \alpha(n))$ bound is tight in the worst case even for our model when each function f_i , $1 \leq i \leq n$, has three faces (two breakpoints) and each function g_i , $1 \leq i \leq n$, has two faces (one breakpoint). The reader is referred to [PS], [Sh], and [SS] for additional results on the complexity of upper envelopes.

An algorithm for calculating the upper envelope in $O(n^2 \alpha(n))$ time is presented in [EPSS]. Thus, we can find the minimum of $h(x_1, x_2)$ with this effort. In the two-center problem the functions are of a slightly simpler nature. It is easily checked [Ha], [KH1] that each of the functions $\{f_i\}$, $\{g_i\}$, $1 \leq i \leq n$, has at most one breakpoint; i.e., each is a pointwise minimum of two linear functions. It is not known whether the $O(n^2 \alpha(n))$ complexity bound is tight even for the upper envelope corresponding to this two-center model.

For the unweighted two-center problem these linear functions have ± 1 slopes only. In this case the complexity bound of the envelope reduces to $O(n^2)$. (Simple examples demonstrate that this bound is tight in the worst case.) To obtain this quadratic bound we invoke the following result, which is implicit in [PS].

LEMMA 4.1. *Suppose that for every i , $1 \leq i \leq n$, the functions f_i and g_i have at most s (a constant) one-dimensional faces. If for every line L , in the (x_1, x_2) plane, the restriction of $h(x_1, x_2)$ over L has $O(n)$ breakpoints, then the complexity bound of the envelope $h(x_1, x_2)$ is $O(n^2)$.*

To see that the conditions of Lemma 4.1 are satisfied for the unweighted two-center problem we prove the next lemma, which we will need later.

LEMMA 4.2. *Let $S = \{s_1, \dots, s_k\}$, $s_1 < \dots < s_k$, be a finite set of reals. Let $\{W_i(x)\}$, $i = 1, \dots, n$, be a collection of piecewise linear concave functions defined on the real line. For each i , $i = 1, \dots, n$, there exist a_i and c_i satisfying $a_i \geq s_j \geq c_i$ for all $j = 1, \dots, k$, and $W_i(x)$ is the pointwise minimum function of $k + 2$ linear functions whose slopes are elements of $S \cup \{a_i, c_i\}$. Define $W(x) = \max_{1 \leq i \leq n} \{W_i(x)\}$. Then $W(x)$ has $O(k^2n)$ breakpoints.*

Proof. Let M be some real such that $M \gg \max_{1 \leq i \leq n} \{\max \{|a_i|, |c_i|\}\}$. For each i, j , $1 \leq i \leq n$, $0 \leq j \leq k + 1$, define

$$s_{ij} = \begin{cases} a_i & \text{if } j = 0, \\ c_i & \text{if } j = k + 1, \\ s_j & \text{otherwise.} \end{cases}$$

Suppose that $W_i(x) = \min_{0 \leq j \leq k+1} \{W_{ij}(x)\}$, where $W_{ij}(x)$ is a linear function with slope s_{ij} . Let $x_{i,0} \leq x_{i,1} \leq \dots \leq x_{i,k+2}$, ($x_{i,0} = -\infty$, $x_{i,k+2} = \infty$), denote the breakpoints of $W_i(x)$ (s_{ij} is the slope of W_i on $[x_{i,j}, x_{i,j+1}]$). Define, for $1 \leq i \leq n$, $0 \leq j \leq k + 1$,

$$\bar{W}_{ij}(x) = \min [W_{ij}(x), M(x - x_{i,j}) + W_{ij}(x_{i,j}), -M(x - x_{i,j+1}) + W_{ij}(x_{i,j+1})].$$

Then $W_i(x) = \max_{0 \leq j \leq k+1} \{\bar{W}_{ij}(x)\}$ for $1 \leq i \leq n$. Next, for $0 \leq j \leq k + 1$, define

$$\bar{W}_j(x) = \max_{1 \leq i \leq n} \{\bar{W}_{ij}(x)\}.$$

It is now easily checked that $W(x) = \max_{0 \leq j \leq k+1} \{\bar{W}_j(x)\}$. For any j , $0 \leq j \leq k + 1$, the collection $\{\bar{W}_{ij}(x)\}$, $i = 1, \dots, n$ has the property that no pair of functions intersect at more than two points. It follows from [At], [HS] that $\bar{W}_j(x)$ has $O(n)$ breakpoints. Therefore, the collection $\{\bar{W}_j(x)\}$, $j = 0, \dots, k + 1$, has a total of $O(kn)$ breakpoints. Each function $\bar{W}_j(x)$ is linear on each interval connecting two such consecutive breakpoints. Thus, $W(x)$ has at most k additional breakpoints in such an interval. This completes the proof that $W(x)$ has $O(k^2n)$ breakpoints.

We comment that the above result is significant only when k is constant or very small ($< \alpha(kn)$), because in general $W(x)$ is the maximum of $O(kn)$ straight segments, so its complexity is always $O(kn\alpha(kn))$.

As mentioned above, for the unweighted two-center problem the slopes of all linear functions are ± 1 only. The restriction of $h(x_1, x_2)$ over any line yields a one-dimensional envelope of n piecewise linear and concave functions. The slopes of all these functions are either ± 1 or $\pm \beta$, where β is the slope of the line. Therefore, from Lemmas 4.1 and 4.2, the complexity bound of the envelope in this case is $O(n^2)$.

To achieve a subquadratic time for finding the minimum of $h(x_1, x_2)$ we should not calculate the entire envelope. We will now show a different approach to minimize $h(x_1, x_2)$ that yields the $O(n \log^3 n)$ bound for several models, including the unweighted two-center model.

First we need the following observation. Let (x_1^*, x_2^*) be a minimum point of h , and consider x_1^* (x_2^*). Then without loss of generality if x_1^* (x_2^*) is not a boundary point of $[0, a]$ ($[0, b]$), there exist $i \neq j$ such that $f_i(x_1^*) = f_j(x_1^*)$ ($g_i(x_2^*) = g_j(x_2^*)$). Moreover, the left derivative of f_i (g_i) and the right derivative of f_j (g_j) have opposite signs at x_1^* (x_2^*). Therefore each pair of functions f_i, f_j (g_i, g_j) contributes at most one element to the set of possible values that the minimum of h can take on. We denote this

set of $O(n^2)$ elements by R . Again, to achieve a subquadratic bound for finding the minimum we should not explicitly calculate all the elements in R .

Motivated by the center problems we now assume that

$$(4.3) \quad \begin{aligned} f_i(x_1) &= \min [a_i x_1 + b_i, a_i^1 x_1 + b_i^1], \\ g_i(x_2) &= \min [c_i x_2 + d_i, c_i^1 x_2 + d_i^1], \quad i = 1, \dots, n. \end{aligned}$$

In this case the set R can be given a compact $O(n)$ representation, where we can then compute any element of R in constant time. Suppose that $h(x_1^*, x_2^*) = f_i(x_1^*)$. Then from the above there exists f_j with $f_i(x_1^*) = f_j(x_1^*)$. The pair of functions contributes at most four intersection points:

$$(4.4) \quad \{(b_j - b_i)/(a_i - a_j), (b_j - b_i^1)/(a_i^1 - a_j), \\ (b_j^1 - b_i)/(a_i - a_j^1), (b_j^1 - b_i^1)/(a_i^1 - a_j^1)\}.$$

(We note in passing that at any given instance only three of these four intersection points can actually arise on h .)

Evaluating f_i at each one of these points contributes four values for the minimum of h .

If, for example, $x_1^* = (b_j - b_i)/(a_i - a_j)$ the respective value that $h(x_1^*, x_2^*)$ takes on is $(\alpha_j - \alpha_i)/(\beta_j - \beta_i)$, where

$$\beta_i = 1/a_i, \quad \beta_j = 1/a_j, \quad \alpha_j = b_j/a_j, \quad \alpha_i = b_i/a_i.$$

By considering all pairs $i \neq j$ we have the sequence $\{(\alpha_j - \alpha_i)/(\beta_j - \beta_i)\}$ $i \neq j$ of cardinality $O(n^2)$. Three more sequences of the same structure are induced by the other terms in (4.4). Similarly, when we assume that $h(x_1^*, x_2^*) = g_i(x_2^*)$ for some i , we will get four more sequences of cardinality $O(n^2)$ from the solutions to pairs of equations $g_i(x_2^*) = g_j(x_2^*)$, $i \neq j$.

The set R consists of the above eight sequences. These sequences are not explicitly generated. To store each of them we need only $O(n)$ space, and we can then compute any element of a sequence in constant time. The particular representation of the sequences enables us to search efficiently over the set R and locate the minimum value of $h(x_1, x_2)$. We will implement the search procedures in [Co] and [MT], which specifically deal with sets having the same representation as R .

Let r^* be a specific element in R , which is defined by some property. In our case r^* is the minimum value of the function $h(x_1, x_2)$. Both models focus on the search for r^* within R . The main ingredient required by these two procedures is a subroutine that can test, for a given value of r , whether $r^* \leq r$. In our context, $r^* \leq r$ if and only if there exist (x_1, x_2) such that $h(x_1, x_2) \leq r$ for the given value of r . Let $T(n)$ indicate the complexity bound of such a feasibility test. Then the procedures in [Co] and [MT] will consume a total of $O[(n + T(n)) \log n]$ time and $O[(n + T(n)) \log^2 n]$ time, respectively, to find r^* .

The feasibility test. We will present a feasibility test for the above model with $T(n) = O(n^2 \log n)$. We will also show that if at least one of the two collections $\{f_i(x_1)\}$ $\{g_i(x_2)\}$ consists only of linear functions, then the test can be performed in $O(n \log^2 n)$ time.

Finally, motivated by the (unweighted) two-center problem, we demonstrate that if the slopes of all the $4n$ linear functions in (4.3) are $(+1, -1)$, the feasibility can also be executed in $O(n \log^2 n)$ time.

Let $r > 0$ be specified. Our task is to determine whether there exists $0 \leq x_1 \leq a$, $0 \leq x_2 \leq b$ such that

$$(4.5) \quad \min [f_i(x_1), g_i(x_2)] \leq r \quad \text{for } i = 1, \dots, n.$$

Consider some i and the set of solutions $\{x_1 | f_i(x_1) \leq r, 0 \leq x_1 \leq a\}$. Due to the concavity of f_i this set consists of at most two disjoint intervals. If there are exactly two intervals each of them will contain a different endpoint of $[0, a]$. The endpoints of the (at most) two intervals can be found in constant time. Therefore, it takes linear time to find the (at most) $2n + 2$ interval endpoints contributed by all the $\{f_i\}$ functions.

Clearly if (4.5) is solvable then it has a solution with x_1 being one of those endpoints.

Let us first sort the $O(n)$ interval endpoints. Define the following subsets:

$$(4.6) \quad \begin{aligned} I_1 &= \{i | f_i(0) > r \text{ and } f_i(a) > r\}, \\ I_2 &= \{i | f_i(x_1) \leq r \text{ for all } 0 \leq x_1 \leq a\}, \\ I_3 &= \{i | f_i(0) \leq r \text{ and } f_i(a) > r\}, \\ I_4 &= \{i | f_i(0) > r \text{ and } f_i(a) \leq r\}, \text{ and} \\ I_5 &= \{f_i(0) \leq r, f_i(a) \leq r \text{ and } i \notin I_2\}. \end{aligned}$$

Let $I = I_3 \cup I_4 \cup I_5$. For each $i \in I$ let $s_i(t_i)$ be the largest (smallest) value of x_1 such that $f_i(x) \leq r$ for all $0 \leq x \leq x_1$ ($x_1 \leq x \leq a$). (For convention $s_i = -\infty$ for $i \in I_4$ and $t_i = \infty$ for $i \in I_3$.)

Let t be one of these $O(n)$ endpoints. Define $I(t) = I_1 \cup \{i | s_i < t < t_i\}$. Inequality (4.5) has a solution with $x_1 = t$ if and only if (4.7) has a solution:

$$(4.7) \quad g_i(x_2) \leq r \quad \text{for all } i \in I(t).$$

Define $G_t(x_2) = \max_{i \in I(t)} \{g_i(x_2)\}$. When we (dynamically) scan the endpoints from left to right the total number of changes in the set $I(t)$ is at most $2n$. An index i will enter the collection at $t = s_i$ and then leave at $t = t_i$. In view of this observation it would be desirable to have a data structure that can maintain $G_t(x_2)$, the pointwise maximum function, dynamically. In particular, such a data structure should support an insertion and a deletion of a function from the collection in sublinear time. For example, if all the functions $\{g_i\}$ in (4.3) are linear we can use the dynamic data structure of Theorem 2.1 to perform such an operation in $O(\log^2 n)$ time, and search over it in $O(\log n)$ time to find the new minimum of G_t . This, in turn, will yield an $O(n \log^2 n)$ feasibility test.

Another special case that can be handled efficiently is when the slopes of the $4n$ linear functions in (4.3) are $+1$ or -1 . In this case we can use the data structure presented in §§ 8 and 9 of [OV] to dynamically maintain the maximal elements of a plane set. Again, the cost of insertion and deletion is $O(\log^2 n)$, which yields an $O(n \log^2 n)$ total time for the feasibility test. (It is implicit in [OV] that the total number of distinct breakpoints of the $O(n)$ functions $\{G_t(x_2)\}_t$ is $O(n \log n)$. It takes $O(n \log^2 n)$ time to generate all of them.)

To help the reader understand the general idea behind the above data structure and its connection to our problem, we now provide a brief description. Let P be a finite set of points in the x, y -plane. A point u in P is maximal (with respect to the natural ordering) if there is no v in P such that $u \leq v$ and $u \neq v$. Consider the subset of P consisting of all maximal points in P . It is customary to connect the maximal elements of a set P by drawing horizontal and vertical lines from each one of them until they cross. This generates

a contour of horizontal and vertical line segments creating a staircase going up toward the left. Such a contour is called the m -contour of P .

To see the connection to our problem, consider the collection of functions $\{g_i\}$, $1 \leq i \leq n$, in the unweighted case. For each i , $1 \leq i \leq n$, let x^i denote the maximum point of the function g_i in the respective interval. Define $P = \{(x^i, g_i(x^i)), 1 \leq i \leq n\}$. It is easy to see that if we rotate the plane by 45° the upper envelope of the functions $\{g_i\}$, $1 \leq i \leq n$, coincides with the m -contour of the above set P . Thus, the upper envelope will be maintained as this m -contour.

We next describe a sketch of the data structure used in [OV] for m -contours. Assume that initially the set P consists of n points in the x, y -plane. These points are stored at the leaves of a balanced binary search tree T , using their y -coordinate as the sorting key. Each internal node α of T will store the m -contour; i.e., the maximal elements of the subset of points corresponding to the leaves in the subtree "covered" by α . The m -contour at α is represented by a concatenable queue Q_α , which can be computed efficiently from the queues Q_γ and Q_δ stored at the two sons γ and δ of α . Generally, Q_α is obtained from Q_γ and Q_δ in a very regular fashion by concatenating one of these queues, say Q_γ (and not just a portion of it), with a tail part of Q_δ . Very much as in the data structure described in § 2, the idea here is to cut off the portions of Q_γ and Q_δ required for the construction of Q_α , and pass them on to α , leaving γ and δ with only a fragment of their original associated structure. From the above it follows that at least at one of the sons an empty structure remains.

Thus, the binary tree T is augmented to a tree T^* in which every internal node α has associated with it that portion of Q_α , say Q_α^* , that was not used in building the m -contour associated with its father. The m -contour of the complete set will be available in one piece at the root of T^* .

We do not know how to obtain a subquadratic test for the general weighted case. We now present an $O(n^2 \log n)$ time test.

Since there are at most $2n + 2$ values that t can take on, it will suffice to show that all the breakpoints of the piecewise linear function $G_t(x_2)$ can be obtained in $O(n \log n)$ time.

Each function $g_i(x_2)$, $1 \leq i \leq n$, has at most one breakpoint ((4.3)). Thus, from Lemma 4.2 we conclude that for each t the continuous piecewise linear function $G_t(x_2)$ has $O(n)$ breakpoints. Applying the standard divide-and-conquer-approach (see § 2), we can obtain all these breakpoints in $O(n \log n)$ time. Inequality (4.7) is solvable if and only if at least one of the breakpoints has an objective value smaller or equal to r .

To summarize we have shown that the two-dimensional model (4.2)–(4.3) can be solved in $O(n^2 \log^2 n)$ time. For the case where all slopes in (4.3) are $+1$ or -1 the model is solved in $O(n \log^3 n)$ time. The latter bound can also be achieved for the case when at least one of the two collections $\{f_i\}$, $\{g_i\}$ consists only of linear functions. We note in passing that in the context of the location model the latter case may correspond to an edge that is a bridge of a graph, i.e., its removal disconnects the graph. The node distance functions to points on a bridge edge are linear.

We now use the above results to claim the complexity bounds for the p -center problem.

First we have shown that the local two-center problem on a general graph can be solved in $O(n^2 \log^2 n)$ and $O(n \log^3 n)$ times for the weighted and unweighted cases, respectively. (The bounds in [KH1], [Mo] for the unweighted case are at least cubic.) Our respective bounds for the global two-center problem are $O(m^2 n^2 \log^2 n)$ and $O(m^2 n \log^3 n)$.

We will use the bounds for the two-center problem to obtain improvements also for $p \geq 3$.

It is well known (see, e.g., [Ha], [KH1]) that the optimal value of the p -center objective (4.1) is an element of a set R of cardinality $O(mn^2)$ that can be constructed from the $O(n^2)$ internodal distances in $O(mn^2)$ time. Therefore the optimal value can be found by determining whether $p(r) \leq p$ for $O(\log(mn^2)) = O(\log n)$ r -covering problems.

Consider some $r > 0$. If there is a solution to the r -covering problem with $p(r) \leq p$, then there is clearly a solution where each of the $p(r)$ centers is located at a (weighted) distance of exactly r from some node. Therefore, each node contributes at most two potential sites on each edge, for a total of $O(mn)$ points. To determine whether the objective value r can be ensured by p -centers we start by selecting $(p - 2)$ points from the set of $O(mn)$ potential sites. (There are $O((mn)^{p-2})$ different selections.) Let X_{p-2} denote the set of $(p - 2)$ points selected. Given a node $v \in V$, it now takes $O(p)$ time to test whether v is "covered" by X_{p-2} , i.e., whether $w_v d(v, X_{p-2}) \leq r$. Therefore, in $O(np)$ time we next determine V^1 , the subset of nodes defined by

$$V^1 = \{v \mid w_v d(v, X_{p-2}) > r\}.$$

Finally we apply the two-dimensional feasibility test to the distance functions corresponding only to the nodes in V^1 . Applying this test for all pairs of edges, we obtain the bounds $O(m^2 n^2 \log n)$ and $O(m^2 n \log^2 n)$ for the weighted and unweighted cases, respectively. The respective bounds for determining whether $p(r) \leq p$ are $O(m^p n^p \log n)$ and $O(m^p n^{p-1} \log^2 n)$.

To conclude we achieve the $O(m^p n^p \log^2 n)$ and $O(m^p n^{p-1} \log^3 n)$ bounds for the weighted and unweighted p -center problems, respectively.

5. Concluding remarks. (1) The above approach and techniques can also be used to improve the complexity bounds of other location problems. For example, the continuous p -center problem [Ta] can be solved by methods similar to those used in § 4. In particular, a continuous local one-center is found in $O(m \log m)$ time, and a continuous local two-center is computed in $O(m^2)$ time.

(2) Our study raises several interesting complexity questions. For example, in several cases above, we have considered a collection $\{G_t(x)\}$, $t = 1, \dots, k$, of one-dimensional functions, where $G_t(x)$ is a pointwise maximum (or minimum) of a set of n linear functions. Moreover, $G_{t+1}(x)$ is obtained from $G_t(x)$ by a single pivot. An interesting problem is to determine the total number of (distinct) breakpoints of all the functions $G_t(x)$, $t = 1, \dots, k$.

Suppose that $k = O(n)$. We have noted in § 2 that for the collection $\{G_t(x)\}$ corresponding to star trees this number is $O(n)$. Using the data structure in [OV] and the sequences of segments considered in [WS], we can show that for a general collection $O(n \log n)$ is an upper bound and $\Omega(n \alpha(n))$ is a lower bound in the worst case.

To explain the upper bound the reader is referred to the description of the data structure in § 2. Each time we perform an insertion or a deletion of a linear function we affect only the concatenable queues $\{Q_\alpha\}$ of $O(\log n)$ internal nodes α of the balanced binary search tree T . (Recall that Q_α "stores" the pointwise minimum function of the set of linear functions corresponding to leaves of the subtree having α as its root.) Updating the above, $O(\log n)$ concatenable queues consumes a total of $O(\log^2 n)$ time. However, it is clear from the process in [OV] that the total number of new breakpoints, "bridges," introduced is only $O(\log n)$; i.e., on the average a constant number of bridges per visited internal node are introduced. This leads to the $O(n \log n)$ upper bound.

To explain the lower bound we consider the sequence of n segments described in [WS]. Let $G(x)$ denote the lower envelope, i.e., the pointwise minimum function, of the above segments when viewed as real functions. (The value of such a real function outside the interval on the real line obtained from the projection of the respective segment is $+\infty$.) It is shown in [WS] that $G(x)$ has $\theta(n\alpha(n))$ distinct breakpoints. We now show how to obtain all the breakpoints of $G(x)$ according to the above model. We first sort the $k = 2n$ endpoints of the n segments. Without loss of generality let the sorted set be given by $-\infty < x_1 < x_2 \cdots < x_k < \infty$. Next we define the functions $\{G_t(x)\}$, $1 \leq t \leq k$, recursively. $G_t(x)$ will be the lower envelope of a collection I_t of linear functions. Suppose each segment is now associated with a linear function that is defined by the unique line containing the given segment. The first collection I_1 , defining $G_1(x)$ will consist only of the linear function corresponding to the segment having x_1 as its left endpoint. I_{t+1} is defined from I_t as follows. If the point x_{t+1} is a right endpoint of some interval, then I_{t+1} is obtained from I_t by deleting the linear function corresponding to that interval. Otherwise, I_{t+1} is derived from I_t by inserting that linear function to the collection. (To ensure that each collection contains exactly n linear functions, as required above in the general model, we can always augment or delete an appropriate number of constant functions, where the constant is $+\infty$.) It is easy to see that $G(x)$ coincides with $G_t(x)$ over the interval $[x_t, x_{t+1}]$. Hence, all the breakpoints of $G(x)$ will appear as breakpoints of the collection of piecewise linear and concave functions $\{G_t(x)\}$, $1 \leq t \leq k$, where $k = 2n$.

In view of the above lower and upper bounds the interesting open problem is to determine what is the exact bound on the total number of distinct breakpoints obtained in this general construction.

Acknowledgment. I wish to thank Micha Sharir for the constructive discussions I had with him as this work developed.

REFERENCES

- [AHU] A. V. AHO, J. E. HOPCROFT, AND J. D. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.
- [At] M. ATALLAH, *Dynamic computational geometry*, Comput. Math. Appl., 11 (1985), pp. 1171–1181.
- [Co] R. COLE, *Slowing down sorting networks to obtain faster sorting algorithms*, J. Assoc. Comput. Mach., 34 (1987), pp. 200–208.
- [CSY] R. COLE, M. SHARIR, AND C. K. YAP, *On K-hulls and related problems*, SIAM J. Comput., 16 (1987), pp. 61–77.
- [DFL] P. M. DEARING, R. L. FRANCIS, AND T. J. LOWE, *Convex location problems on tree networks*, Oper. Res., 24 (1976), pp. 628–642.
- [DW] Z. DREZNER AND G. O. WESOŁOWSKY, *Location of multiple obnoxious facilities*, Transportation Sci., 19 (1985), pp. 193–202.
- [Dy] M. E. DYER, *Linear time algorithms for two- and three-variable linear programs*, SIAM J. Comput., 13 (1984), pp. 31–45.
- [EW] H. EDELSBRUNNER AND E. WELZL, *Constructing belts in two-dimensional arrangements with applications*, SIAM J. Comput., 15 (1986), pp. 271–284.
- [EPSS] H. EDELSBRUNNER, J. PACH, J. SCHWARTZ, AND M. SHARIR, *The lower envelope of bivariate functions and its applications*, in Proc. 28th Annual IEEE Symposium on Foundations of Computer Science, 1987, pp. 27–37.
- [Fr] G. N. FREDERICKSON, *Shortest path problems in planar graphs*, SIAM J. Comput., 16 (1987), pp. 1004–1022.
- [FT] M. FREDMAN AND R. TARJAN, *Fibonacci heaps and their uses in improved network optimization algorithms*, J. Assoc. Comput. Mach., 34 (1987), pp. 596–615.
- [Go] A. J. GOLDMAN, *Optimal center location in simple networks*, Transportation Sci., 5 (1971), pp. 212–221.

- [Ha] S. L. HAKIMI, *Optimal locations of switching centers and the absolute centers and medians of a graph*, Oper. Res., 12 (1964), pp. 450–459.
- [HS] S. HART AND M. SHARIR, *Nonlinearity of Davenport–Schinzel sequences and of generalized path compression schemes*, Combinatorica, 6 (1986), pp. 151–177.
- [KH1] O. KARIV AND S. L. HAKIMI, *An algorithmic approach to network location problems. I: The p -centers*, SIAM J. Appl. Math., 37 (1979), pp. 513–537.
- [KH2] ———, *An algorithmic approach to network location problems. II: The p -medians*, SIAM J. Appl. Math., 37 (1979), pp. 538–563.
- [Me] N. MEGIDDO, *Linear-time algorithms for linear programming in R^3 and related problems*, SIAM J. Comput., 12 (1983), pp. 759–776.
- [MT] N. MEGIDDO AND A. TAMIR, *New results on the complexity of p -center problems*, SIAM J. Comput., 12 (1983), pp. 751–758.
- [Mo] J. A. MORENO, *A new result on the complexity of the p -center problem*, Technical Report, Universidad Complutense, Madrid, Spain, 1986.
- [OV] M. H. OVERMARS AND J. VAN LEEUWEN, *Maintenance of configurations in the plane*, J. Comput. System Sci., 23 (1981), pp. 166–204.
- [PS] J. PACH AND M. SHARIR, *The upper envelope of piecewise-linear functions and the boundary of a region enclosed by convex plates: combinatorial analysis*, Technical Report 279, Computer Science Department, Courant Institute of Mathematical Sciences, New York University, New York, March 1987.
- [SS] J. T. SCHWARTZ AND M. SHARIR, *On the two-dimensional Davenport–Schinzel problem*, Technical Report, Courant Institute of Mathematical Sciences, New York University, New York, 1986.
- [Sh] M. SHARIR, *Almost linear upper bounds on the length of general Davenport–Schinzel sequences*, Combinatorica, 7 (1987), pp. 131–143.
- [Ta] A. TAMIR, *On the solution value of the continuous p -center location problem on a graph*, Math. Oper. Res., 12 (1987), pp. 340–349.
- [TZ] A. TAMIR AND E. ZEMEL, *Locating centers on a tree with discontinuous supply and demand regions*, Math. Oper. Res., 7 (1982), pp. 183–192.
- [WS] A. WIERNIK AND M. SHARIR, *Planar realization of nonlinear Davenport–Schinzel sequences by segments*, Discrete Comput. Geometry, 3 (1988), pp. 15–47.

POLYNOMIAL BOUND FOR A CHIP FIRING GAME ON GRAPHS*

GÁBOR TARDOS†

Abstract. Björner, Lovász, and Shor have introduced a chip firing game on graphs. This paper proves a polynomial bound on the length of the game in terms of the number of vertices of the graph provided the length is finite. The obtained bound is best possible within a constant factor.

Key words. graph, game, algorithm, polynomial time

AMS(MOS) subject classifications. 68R10, 68Q25, 05C35

Introduction. Analyzing a certain “balancing game,” Spencer [3] introduced a chip-distribution process on a line. The running time and other properties of this process were also studied by Anderson et al. [1]. Björner, Lovász, and Shor [2] extended the process to general graphs. This generalization can be formulated as a one-person game as follows.

A finite simple graph G is given with a finite stack of chips at each vertex. A legal move consists of selecting a vertex with stack size at least its degree and passing a chip from its stack simultaneously to each neighboring vertex. We call this *firing* a vertex. The game continues as long as legal moves exist.

By definition, the total number of chips remains constant during the game.

The length of the game is the total number of moves (this may be infinite).

Björner, Lovász, and Shor proved the following theorem.

THEOREM 1. *The initial configuration determines whether or not the game will continue indefinitely. If the game eventually terminates, then both the final configuration and the length of the game are determined by the initial configuration.*

Their results yield an exponential upper bound (in terms of the number of vertices) on the number of moves provided the game is finite. They asked if there exists a polynomial upper bound. In this paper such a bound is given in Theorem 2.

THEOREM 2. *Let G be a graph with n vertices, m edges, and diameter d . Then any finite game on G terminates within $2nmd$ moves.*

In terms of the parameter n , this yields an upper bound of $O(n^4)$. This bound is best possible (apart from the constant), as seen from the following observation.

PROPOSITION 3. *There exists a graph on n vertices and an initial configuration of chips such that the length of the resulting game is $\Theta(n^4)$.*

The proofs. Our proofs do not refer to Theorem 1 and are self-contained. First we prove two lemmas.

LEMMA 4. *If each vertex has already been fired at least once, then the game will never terminate.*

Proof. Assume each vertex has already been fired at least once, and yet the game terminates. Let v be the vertex that has been idle (not fired) for the longest time at termination. After v had last been fired, all its neighbors were fired, so at least $\deg(v)$ chips arrived at v and no chips left. So at least $\deg(v)$ chips are now in the stack at v and therefore firing v is legal. This contradiction proves the lemma. $\square$

Let u denote the total number of chips.

LEMMA 5. *The numbers indicating how often each of two adjacent vertices were fired cannot differ by more than u at any time in the game.*

* Received by the editors January 13, 1988; accepted for publication January 19, 1988.

† Department of Algebra, Eötvös University, Múzeum krt 6-8, Budapest, Hungary 1088.

Proof. In the course of the game we move the chips along edges. If (v, w) is an edge of G , v has been fired i times, and w has been fired j times, then we moved i chips from v to w and j chips back. Now let us fix an edge (v_0, w_0) of G where v_0 was fired i_0 times and w_0 was fired j_0 times. Suppose $i_0 < j_0$. Let H be the set of those vertices of G which were fired at most i_0 times. So $v_0 \in H$, but $w_0 \notin H$. By the definition of H along any edge connecting H with its complement, more chips moved to H than from H . On the edge (v_0, w_0) this difference is $j_0 - i_0$. So the total number of chips at vertices of H has gone up by at least $j_0 - i_0$. Consequently, $j_0 - i_0 \leq u$. $\square$

Proof of Theorem 2. Let d be the diameter of G (i.e., the largest distance between pairs of vertices). By Lemma 4 there exists a vertex v of G that was idle during the entire game. By repeated application of Lemma 5 we obtain that any vertex w at distance d' from v has been fired at most $d'u$ times. So each vertex was fired at most du times. This means there were at most ndu moves. If u is greater than the sum of degrees of the vertices of G , obviously the game will go on forever. So in a halting game, $u \leq 2m$. Consequently the number of moves is $\leq 2nmd$. $\square$

Proof of Proposition 3. Consider the following graph G . Take the complete graph on $\lfloor n/2 \rfloor$ vertices and attach a path of length $n - \lfloor n/2 \rfloor$ to it starting at one of the vertices of the complete graph. Put $\lfloor n/2 \rfloor^2 - 2n$ chips at the other end of the path. It is easy to prove that with this starting configuration, the game will continue for $n^4/32 + O(n^3)$ moves and then halt. $\square$

REFERENCES

- [1] R. ANDERSON, L. LOVÁSZ, P. SHOR, J. SPENCER, É. TARDÓS, AND S. WINOGRAD, *Disks, balls and walls: the analysis of a combinatorial game*, Amer. Math. Monthly, to appear.
- [2] A. BJÖRNER, L. LOVÁSZ, AND P. SHOR, *Chip firing games on graphs*, manuscript in preparation.
- [3] J. SPENCER, *Balancing vectors in the max norm*, Combinatorica, 6 (1986), pp. 55–65.

THE PARALLEL COMPLEXITY OF ELEMENT DISTINCTNESS IS $\Omega(\sqrt{\log n})^*$

PRABHAKAR RAGDE†, WILLIAM STEIGER‡, ENDRE SZEMERÉDI§,
AND AVI WIGDERSON¶

Abstract. We consider the problem of element distinctness. Here n synchronized processors, each given an integer input, must decide whether these integers are pairwise distinct, while communicating via an infinitely large shared memory.

If simultaneous write access to a memory cell is forbidden, then a lower bound of $\Omega(\log n)$ on the number of steps easily follows (from S. Cook, C. Dwork, and R. Reischuk, *SIAM J. Comput.*, 15 (1986), pp. 87-97.) When several (different) values can be written simultaneously to any cell, then there is a simple algorithm requiring $O(1)$ steps.

We consider the intermediate model, in which simultaneous writes to a single cell are allowed only if all values written are equal. We prove a lower bound of $\Omega((\log n)^{1/2})$ steps, improving the previous lower bound of $\Omega(\log \log \log n)$ steps (F.E. Fich, F. Meyer auf der Heide, and A. Wigderson, *Adv. in Comput.*, 4 (1987), pp. 1-15).

The proof uses Ramsey-theoretic and combinatorial arguments. The result implies a separation between the powers of some variants of the PRAM model of parallel computation.

Key words. parallel computation, lower bounds, parallel random access machines

AMS(MOS) subject classification. 68Q10

1. Introduction. The parallel random access machine (PRAM) is the most widely used theoretical model of parallel computation. In this machine, n synchronized processors $P_1, P_2, \dots, P_n$ have read and write access to a shared memory $\{M_i | i \in \mathbb{N}\}$. Each cell M_i of shared memory is initialized to zero. Each processor is a (possibly infinite) state machine. One step of computation consists of two phases. In the write phase, each processor may write an integer value to some shared memory cell. All writes take place simultaneously. In the read phase, each processor may read some shared memory cell. Based on what it has seen, each processor then assumes a new state.

A PRAM may be restricted to disallow simultaneous read or write access by several processors to the same cell. We allow concurrent reads and writes. If several processors attempt to simultaneously write different values into the same cell, a write conflict arises. Here, we discuss two methods of write-conflict resolution that have appeared in the literature. In the COMMON model [K], the algorithm itself must ensure that a write conflict never occurs; all processors simultaneously writing into the same cell must be writing the same value. In the ARBITRARY model, the machine will resolve the write conflict arbitrarily; one of the values being written will appear in the cell, but it is impossible in advance to know which value. Algorithms on the ARBITRARY model must work regardless of who wins each competition to write.

* Received by the editors January 8, 1987; accepted for publication (in revised form) February 12, 1988.

†Department of Computer Science, University of Toronto, Toronto, Ontario, Canada M5S 1A4. This work was performed while the author was at the University of California, Berkeley, California 94720, and was supported by an IBM Graduate Fellowship and a Natural Sciences and Engineering Research Council of Canada Postgraduate Scholarship.

‡Rutgers University, New Brunswick, New Jersey 08903.

§Hungarian Academy of Sciences, Budapest, Hungary.

¶The Hebrew University, Jerusalem, Israel. This work was done while the author was at the Mathematical Sciences Research Institute, Berkeley, California 94720, and was supported by National Science Foundation grant MCS-8120790

The input to the PRAM is an n -tuple $(x_1, x_2, \dots, x_n)$, where the input variable x_i affects the initial state of P_i . Thus the input variables are initially distributed one to a processor. (This state of affairs is equivalent to storing the values in the first n cells of memory at the start of the computation.) We will choose input variables from $\mathbb{N}$ (the positive integers). We denote the set $\{1, 2, \dots, k\}$ by $[k]$. All logarithms are to the base 2.

The element distinctness problem demonstrates the weakness of the COMMON model, relative to the ARBITRARY model, when the size of shared memory is infinite. In this problem, processor P_1 must halt in state “not-distinct” if there exist indices i, j (where $i \neq j$) such that $x_i = x_j$; otherwise, it must halt in state “distinct”. In §2, we show how to solve this problem in $O(1)$ steps on the ARBITRARY model, and discuss two algorithms to solve it on the COMMON model in $O(\log n)$ steps. In §3, we state and prove a generalization of a result by Hensel concerning the covering of a clique by bipartite subgraphs, which is needed for the next section. Finally, in §4 we present a lower bound of $\Omega(\sqrt{\log n})$ steps for solving element distinctness on the COMMON model. The lower bound proceeds by using Ramsey theory to restrict the set of inputs under consideration and thus simplify the actions of processors to the point where they transfer information only by a combination of the two methods discussed in §2. Then an adversary argument is given, in which the result from §3 and other combinatorial theorems are utilized.

2. Upper bounds. We first show how to solve element distinctness in constant time on the ARBITRARY model, and then give two different $O(\log n)$ algorithms for the COMMON model. All of these algorithms appear in [FMW].

THEOREM 1. *Element distinctness can be solved in $O(1)$ steps on the ARBITRARY model.*

Proof. Consider the following algorithm, which uses all of the infinite memory as a hash table. In the first step, each processor P_i writes the value i into cell M_{x_i} . Each processor then reads the cell into which it just wrote. If all variables are distinct, then each processor will see the value it wrote. But if two variables are equal, then at least two processors will attempt to write into the same cell, and at least one will see a value different from the one it wrote. If the latter occurs, P_1 can be informed in one more step, by having any such processor write the value “not-distinct” into M_1 . $\square$

Notice that this algorithm will work on any model that allows different values to be simultaneously written to the same cell, and that the infinite memory is essential to the algorithm.

Obviously, this algorithm will not work on the COMMON model. On this model, however, shared memory can be used as a hash table in a more indirect way. Given two nonintersecting sets of indices S_1 and S_2 (both subsets of $[n]$), each P_i ($i \in S_1$) writes 1 into cell M_{x_i} . Then each P_j ($j \in S_2$) reads cell M_{x_j} . The sets of values $\{x_i | i \in S_1\}$ and $\{x_j | j \in S_2\}$ are distinct if and only if each processor P_j ($j \in S_2$) reads the value zero.

This method can be used to solve element distinctness in $\lceil \log n \rceil + 1$ steps on the COMMON model. At the t th step ($t \leq \lceil \log n \rceil$), the set S_1 consists of those indices i for which the t th binary digit of i is 1, and S_2 consists of those indices j for which the t th binary digit of j is zero. There is a slight technical problem because memory is not re-initialized to zero, but this can be easily overcome by having each processor write the value t (instead of 1) at the t th step. In the last step, any processor who has detected nondistinctness informs P_1 by writing the value ‘not-distinct’ into M_1 . Since any distinct indices i, j differ in at least one binary digit, the distinctness of every pair

(x_i, x_j) will be verified.

Throughout the above algorithm, processors are only aware of the value of the variable that they are given at the beginning of the algorithm. The algorithm does not use the local computational power of processors, nor does it require memory cells to hold large values. But again, the fact that memory is infinite is essential, as it is not the values written that are important, but the location into which they are written.

Another method of answering the question is to allow processors to learn the values of other variables. If a processor knows two or more variables (as reflected in the state of that processor), then it knows if those variables are distinct.

This idea can be used to give a different algorithm for solving element distinctness on the COMMON model, in $\lceil \log n \rceil$ steps. One cell of shared memory is dedicated to each processor, to serve as its mailbox. Initially, each processor knows just one variable. In the t th step, each processor P_i computes an encoding of all the variables it knows and places this encoding in the mailbox of processor P_j , where j differs from i only in the t th binary digit. Each processor then reads its own mailbox and learns the values stored there. A straightforward analysis shows that at each step, the number of distinct variables that any processor knows will double. Thus after $\lceil \log n \rceil$ steps, each processor will know the value of all n variables, and processor P_1 can halt in the desired state.

This second method illustrates the power of the PRAM model; it shows that *any* function of n variables can be computed in $O(\log n)$ steps. It uses only n cells, but the capacity of those cells must be large, and the local computational power of each machine is used in computing encodings. Indirect access to memory is not used, nor are concurrent reads and writes; it is the values written that are important, as the location into which each processor writes at each step is independent of the input.

In §4 we shall see, in the course of proving an $\Omega(\sqrt{\log n})$ lower bound for element distinctness on the COMMON model, that these two methods are essentially the only two ways in which processors can verify distinctness. The temptation to conjecture that the complexity of element distinctness on the COMMON model is $\Theta(\log n)$ is misleading; in [FRW] a general simulation result is given, which has as a corollary an algorithm for element distinctness on this model requiring $O(\log n / \log \log n)$ steps.

3. A combinatorial theorem. In this section we prove a combinatorial result that, in addition to aiding in the proof of the subsequent lower bound, is of independent interest.

A *semi-bipartition* of a set X is a pair of sets $\{A, B\}$, where $A, B \subseteq X$, and $A \cap B = \emptyset$. For $y, z \in X$, we say the pair $\{y, z\}$ is covered by $\{A, B\}$, if either $y \in A$ and $z \in B$, or $y \in B$ and $z \in A$. (We always assume that the elements of a pair are distinct.) The size of a semi-bipartition $\{A, B\}$ is $|A| + |B|$. For a collection $\mathcal{C}$ of semi-bipartitions, we define $\text{size}(\mathcal{C}) = \sum_{\{A, B\} \in \mathcal{C}} (|A| + |B|)$.

Hansel [H] provides a very short and elegant proof of the following theorem. Pippenger [P] gives an interesting proof using information-theoretic arguments.

THEOREM 2. *Let X be a set of size n , and let $\mathcal{C}$ be a collection of semi-bipartitions over X such that every pair of elements of X is covered by some semi-bipartition in $\mathcal{C}$. Then $\text{size}(\mathcal{C}) \geq n \log n$.*

We generalize this result in two ways: first, we relax the requirement that every pair of elements be covered, and second, we generalize the notion of covering by semi-bipartitions.

The following theorem is implied by a theorem of Fredman and Komlós [FK]; it admits short proofs in the styles of [H] and [P]. We give a proof that is a generalization of Hansel's.

THEOREM 3. *Let X be a set of size n , and let E be a set of pairs of elements from X . Let α be the largest number of elements from X that can be chosen so that no two of them are in a pair of E . If $\mathcal{C}$ is a collection of semi-bipartitions over X such that every pair in E is covered by some semi-bipartition in $\mathcal{C}$, then $\text{size}(\mathcal{C}) \geq n \log(n/\alpha)$.*

Proof. Let $\mathcal{C} = \{\{A_i, B_i\}, i = 1, 2, \dots, g\}$. For $x \in X$, let c_x be the number of semi-bipartitions $\{A_i, B_i\}$ such that $x \in A_i \cup B_i$. Then $\text{size}(\mathcal{C}) = \sum_{x \in X} c_x$.

Let U be the set of all tuples u of length g , where $u_i \in \{A_i, B_i\}$. We say that a tuple $u \in U$ is *consistent* with x if, for $1 \leq i \leq g$, either $x \in u_i$ or $x \notin A_i \cup B_i$. For each $x \in X$, there are 2^{g-c_x} tuples consistent with x . A tuple may be consistent with more than one element of X . If a tuple u were consistent with $\alpha + 1$ elements of X , then two of those elements (say x and y) would appear in a pair of E . But since $\mathcal{C}$ covers all pairs in E , there must exist some semi-bipartition $\{A_i, B_i\}$ such that, say, $x \in A_i, y \in B_i$. This means that $u_i = A_i$ by the consistency of u with x , and $u_i = B_i$ by the consistency of u with y , a contradiction. Hence any tuple is consistent with at most α elements of X .

Thus $\alpha 2^g = \alpha |U| \geq \sum_{x \in X} (\# \text{ tuples consistent with } x) = \sum_{x \in X} 2^{g-c_x}$, and so, $\alpha \geq \sum_{x \in X} 2^{-c_x}$.

$(1/n) \sum_{x \in X} 2^{-c_x} \geq (\prod_{x \in X} 2^{-c_x})^{1/n}$, since the geometric mean does not exceed the arithmetic mean. Thus $(\prod_{x \in X} 2^{-c_x})^{1/n} \leq \alpha/n$. Taking logarithms, we have $\sum_{x \in X} c_x \geq n \log(n/\alpha)$, as required. $\square$

Next, we introduce a more general notion of covering. Let A be a set of tuples of length ℓ with components chosen from the set X . Let B be another such set. If $A \cap B = \emptyset$, we call $\{A, B\}$ a *tuple system* of length ℓ . For $y, z \in X$, we say that the tuple system $\{A, B\}$ covers the pair (y, z) if there exists a tuple $\alpha \in A$, a tuple $\beta \in B$, and a position i such that $\alpha_i = y, \beta_i = z$, and $\alpha_j = \beta_j$ for all $j \neq i$. That is, α, β differ only in position i , and y, z are the elements from the sets A, B in that position. If such a pair α, β of tuples exists, we say they are *similar* and cover (y, z) .

If $\ell = 1$, this reduces to covering by semi-bipartitions. A collection of tuple systems $\mathcal{C}$ is a set $\{\{A_1, B_1\}, \{A_2, B_2\}, \dots, \{A_k, B_k\}\}$, where A_i, B_i are sets of tuples of length ℓ_i , and $A_i \cap B_i = \emptyset$. We define $\text{size}(\mathcal{C}) = \sum_{i=1}^k (|A_i| + |B_i|)$.

The following is the main result of this section.

THEOREM 4. *Given a set X of size n and a set E of pairs of elements from X , let α be the largest number of elements of X , no two of which are in a pair of E . Suppose $\mathcal{C}$ is a collection of tuple systems of length at most ℓ , and every pair of E is covered by a tuple system in $\mathcal{C}$. Then, for n sufficiently large, $\text{size}(\mathcal{C}) \geq n(\frac{1}{4} \log n - \frac{1}{2} \log(\alpha\ell) - \log \log n)$.*

As ℓ becomes larger, this bound becomes weaker. We conjecture that the lower bound on size is in fact $n \log(n/\alpha)$ regardless of the length of tuples. For our application, it will be the case that $\alpha\ell = O(n^{1/2-\epsilon})$ for some positive constant ϵ , and thus we obtain a lower bound of $\Omega(n \log n)$ on $\text{size}(\mathcal{C})$.

Proof of Theorem 4. Suppose we are given a tuple system $\mathcal{C}$ that covers E in the fashion described, and in addition, $\text{size}(\mathcal{C}) < n(\frac{1}{4} \log n - \frac{1}{2} \log(\alpha\ell) - \log \log n)$. For each tuple system $\{A_i, B_i\} \in \mathcal{C}$, we are going to construct semi-bipartitions that cover most of the edges that the tuple system covered, and then apply Theorem 3 to obtain a contradiction.

Let $\mathcal{C}'$ be an initially empty collection of semi-bipartitions, and E' an initially empty collection of pairs of distinct elements of X . E' will contain the pairs from E that $\mathcal{C}'$ does not cover.

Let a_i, b_i be the initial sizes of A_i, B_i respectively. We say the tuple system $\{A_i, B_i\}$ is *sparse* if there are at most $(|A_i||B_i|)/\sqrt{n}$ similar pairs in $\{A_i, B_i\}$. If

$\{A_i, B_i\}$ is not sparse, then there exists some $\alpha \in A_i$ which is similar to at least $|B_i|/\sqrt{n}$ tuples in B_i . There is some position p such that at least $|B_i|/\ell\sqrt{n}$ tuples in B_i differ from α only in position p , since α is a tuple of length at most ℓ . Let A'_i be the set that includes α and all tuples in A_i that differ from α only in position p , and B'_i be all tuples in B_i that differ from α only in position p . Then $|B'_i| \geq |B_i|/\ell\sqrt{n}$. Also, $A'_i \cap B'_i = \phi$.

Let A be the set of elements occurring at position p in tuples of A'_i , and let B be the set of elements occurring at position p in tuples of B'_i . We know that $|A| = |A'_i|$, and $|B| = |B'_i|$. Furthermore, since $A_i \cap B_i = \phi$ and all tuples in A'_i, B'_i differ only in position p , $A \cap B = \phi$. Thus $\{A, B\}$ is a semi-bipartition of X . We add $\{A, B\}$ to C' . $\{A, B\}$ covers all pairs covered by $\{A'_i, B'_i\}$.

We remove A'_i from A_i and B'_i from B_i . We have not accounted for pairs covered by similarities between A'_i and $B_i - B'_i$, or between B'_i and $A_i - A'_i$. But any tuple in $B_i - B'_i$ is similar to at most one tuple in A'_i , for it can differ from a tuple in A'_i only in a position $p' \neq p$, and all tuples in A'_i have the same entry in position p' but different entries in position p . Similarly, any tuple in $A_i - A'_i$ is similar to at most one tuple in B'_i . Thus we have neglected at most $|A_i| + |B_i| \leq a_i + b_i$ similar pairs of tuples. We add the pairs of elements that these similar pairs of tuples cover to E' .

If $\{A_i, B_i\}$ is not yet sparse, we repeat this process. Each time, we shrink B_i by at least a factor of $\gamma = 1 - (1/\ell\sqrt{n})$. In at most $\log_{(1/\gamma)} b_i$ iterations, $\{A_i, B_i\}$ becomes sparse (note that an empty tuple system is sparse). The total number of pairs we have added to E' is at most $\log_{(1/\gamma)} b_i (a_i + b_i) \leq \ell\sqrt{n} \log b_i (a_i + b_i)$.

When $\{A_i, B_i\}$ becomes sparse, we simply add to E' all pairs of elements that are covered by similarities in $\{A_i, B_i\}$. This adds at most $a_i b_i / \sqrt{n}$ pairs to E' .

When this process is completed, clearly $\text{size}(C') \leq \text{size}(C)$. Furthermore, $|E'| \leq \sum_i ((a_i b_i / \sqrt{n}) + (a_i + b_i) \ell \sqrt{n} \log b_i)$. Since a_i, b_i , and $\sum_i (a_i + b_i)$ are all less than $\frac{1}{4} n \log n$ by the assumption on $\text{size}(C)$, it follows that for n sufficiently large, $|E'| \leq \frac{1}{3} \ell n^{3/2} \log^2 n$.

As a result, we have a collection C' of semi-bipartitions and a set E' of at most $\frac{1}{3} \ell n^{3/2} \log^2 n$ pairs such that every pair in $E - E'$ is covered by C' . Let D be the largest set of elements of X such that no pair in $E - E'$ contains two elements of D . By Theorem 3, we know that $\text{size}(C') \geq n \log(n/|D|)$. All pairs in E that contain two elements of D must also be pairs in E' . Hence the number of pairs in E that have two elements in D is at most $|E'|$. Let D' be the largest subset of D such that no pair in E contains two elements of D . We know that $|D'| \leq \alpha$ by the definition of α .

Turán's theorem [B, p. 282] states that given any set S and any collection P of pairs from S , there is a set S' such that no pair in P contains two elements from S' , and $|S'| \geq |S|^2 / (|S| + 2|P|)$. Applying this to D and the pairs of E with both elements in D , we know that $|D'| \geq |D|^2 / (|D| + 2|E'|)$.

If $|D| > |E'|$, then $\alpha \geq |D'| \geq |D|^2 / 3|D|$, and so $|D| \leq 3\alpha$. If $|D| \leq |E'|$, then $\alpha \geq |D|^2 / 3|E'|$, and $|D| \leq n^{3/4} \log n \sqrt{\alpha \ell}$. In either case we have our contradiction, since we can conclude that $\text{size}(C') \geq n(\frac{1}{4} \log n - \frac{1}{2} \log(\alpha \ell) - \log \log n)$, but $\text{size}(C') \geq \text{size}(C)$. $\square$

4. The lower bound. In this section, we prove a lower bound for the element distinctness problem on the COMMON model, thus demonstrating a separation between the COMMON and ARBITRARY models when both are allowed infinite memory. The proof combines techniques from [FMW] and [MW] with the result from §3.

The following rather odd definition is motivated by technical problems that arise in the main result if overwriting a shared memory cell is allowed. A similar technique was used in [FMW].

DEFINITION. A *write-once* COMMON PRAM is a COMMON PRAM with the following modifications: if a cell in shared memory is written into at a particular step, then it may not be written into at any subsequent step. Also, at step t , each processor is allowed to simultaneously read t cells $M_{i_1}, M_{i_2}, \dots, M_{i_t}$ of shared memory, instead of just one. The final restriction is that, if cell M_{i_j} defined above was written into at all, it must have been written into at step j .

LEMMA 5. *T steps of a COMMON PRAM with m cells of shared memory can be simulated by T steps of a write-once COMMON PRAM with m^2 cells of shared memory.*

Proof. Let $\langle \cdot, \cdot \rangle : [m] \times [m] \rightarrow [m^2]$ be any one-to-one function. At step t of the simulation, if P_i in the simulated machine writes into M_j and reads from M_k , then P_i in the write-once machine writes into $M_{\langle t, j \rangle}$, and reads $M_{\langle 1, k \rangle}, M_{\langle 2, k \rangle}, \dots, M_{\langle t, k \rangle}$. From the contents of these t cells, P_i can determine what M_k in the simulated machine would have contained at step t . $\square$

By Lemma 5, any lower bound for element distinctness on write-once COMMON with infinite memory is a lower bound on regular COMMON with infinite memory. We can now state and prove the main result of this paper.

THEOREM 6. *On a write-once COMMON PRAM with infinite memory, element distinctness requires $\Omega(\sqrt{\log n})$ steps.*

We introduce some definitions to be used in the proof. A (k, s, a) -bundle on X will be a partial order on the set X , where $ks + a = |X|$. Of the $ks + a$ elements, ks are in k disjoint antichains of size $s > 1$; the remaining a of the elements are in disjoint antichains of size 1, and there is a total ordering among all antichains. An antichain of size s will be called *nontrivial*. Figure 1 gives a simple example of such a partial order.

$$x_{10} < \frac{x_7}{x_3} < \frac{x_1}{x_9} < x_2 < x_4 < \frac{x_6}{x_5} < x_8$$

FIG. 1. A $(3, 2, 4)$ bundle on $\{x_1, x_2, \dots, x_{10}\}$.

Let X denote the set of variables $\{x_1, x_2, \dots, x_n\}$. Suppose we are given an algorithm that solves element distinctness. We will prove a lower bound of $T + 1$ steps on the running time of this algorithm, where $T = \Omega(\sqrt{\log n})$.

The state of a processor P_i after step t (and consequently, the cells it chooses to write into and read from at step $t + 1$) can be a complicated function of the variables in X . The idea is to concentrate on a restricted set of inputs I_t which are “indistinguishable” to the algorithm through step t . For inputs in I_t , the state of a processor during the first t steps will be a function of a “small” number of variables. We can then apply Ramsey theory to simplify the functions describing access to shared memory. The input set I_t will be described in terms of objects Π_t , C_t , and S_t . We will display, for t ranging from zero to T , the following:

1. An infinite set S_t , where $S_t \subseteq \mathbb{N}$, and $S_t \subseteq S_{t-1}$. S_t is the set of possible values of a variable for inputs in I_t . We will define S_t so as to simplify all functions describing access to shared memory.

2. A (k_t, s_t, a_t) -bundle Π_t over X . Π_t will be a refinement of Π_{t-1} ; that is, if x_i, x_j are comparable in Π_{t-1} , then they are comparable in Π_t . The bundle will satisfy the inequalities:

- $k_t \leq 3 \cdot 2^{t^2} + 5t + 3$,

$$\bullet a_t \leq \frac{n}{3} - \frac{n}{3^{t+2}} + 3^{2t^2} + 5t + 4.$$

P_{i_t} will ‘describe’ inputs in I_t ; the values of the variables in any input of I_t will satisfy the constraints of Π_t . We shall choose T so that it will always be the case that $a_t \leq 2n/3$ for $t \leq T$, that is, at least a third of the variables are in nontrivial antichains of Π_t .

3. A collection C_t of tuple systems over X , where $C_t \supseteq C_{t-1}$, $\text{size}(C_t) \leq n(t+1)^2$, and the size of tuples in C_t is at most 3^t .

4. For each processor P_i , a set $V_t^i \subseteq X$ with the properties:

- $V_{t-1}^i \subseteq V_t^i$ and $|V_t^i| \leq 3^t$;
- All the variables in V_t^i are comparable in Π_t , that is, V_t^i is a chain of Π_t ;
- The state of P_i after step t will be a function of *only* the variables in V_t^i , for all inputs in I_t .

We may now describe the set of inputs I_t . $\hat{x} \in I_t$ if and only if:

1. $\hat{x} \in S_t^n$;
2. $\hat{x}$ is consistent with Π_t ;
3. There is at most one pair (j, k) such that $\hat{x}_j = \hat{x}_k$;
4. If $\hat{x}_j = \hat{x}_k$, (x_j, x_k) is not covered by any tuple system in C_t . Note that the containment properties of C_t, Π_t , and S_t ensure that $I_t \subseteq I_{t-1}$.

Suppose there is at least one pair (x_j, x_k) such that (x_j, x_k) is not covered by any tuple system in C_t , and x_j, x_k are in the same antichain of Π_t . We know that at most one of x_j, x_k can affect the state of P_1 after step t , since V_t^1 is a chain in Π_t . Suppose x_k does not affect the state of P_1 after step t . Let $\hat{x}$ be an input in I_t such that all $\hat{x}_i$ are distinct. Let $\hat{x}'$ be the input such that $\hat{x}'_i = \hat{x}_i$ for $i \neq k$, but $\hat{x}'_k = \hat{x}'_j$. By choice of (x_j, x_k) , $\hat{x}'$ is also an input in I_t , and by construction, not all the components of $\hat{x}'$ are distinct. Furthermore, since the state of P_1 is not a function of x_k , P_1 cannot distinguish these inputs after step t . Thus we obtain a lower bound of $t+1$ steps. We now show how to construct the objects described above, up to index T , where $T = \Omega(\sqrt{\log n})$.

Before step 1 we let Π_0 be the empty order, that is, $k_0 = 1$, $s_0 = n$, $a_0 = 0$. We also set $C_0 = \phi$, $S_0 = \mathbb{N}$, $I_0 = (\mathbb{N})^n$, $V_0^i = \{x_i\}$. Now suppose we have constructed all the required objects up to index t . We show how to extend the construction to index $t+1$, by consideration of what happens at step $t+1$ for inputs in I_t .

We say a function $f(y_1, y_2, \dots, y_\ell)$ is *defined on increasing tuples over S* if the domain of f consists of the increasing tuples over S . A tuple $\hat{y}$ is increasing if $\hat{y}_i < \hat{y}_j$ for $i < j$.

For a step $t' \leq t+1$, let $x_{i_1}, x_{i_2}, \dots, x_{i_k}$ be the variables in $V_{t'-1}^i$, where the indices are chosen so that $x_{i_1} < x_{i_2} < \dots < x_{i_k}$ in Π_t . We know this is possible because $V_{t'-1}^i$ is totally ordered in Π_t . The state of P_i just before step t' , for inputs in I_t , is a function of $x_{i_1}, x_{i_2}, \dots, x_{i_k}$. Thus the indices of the cells that P_i writes into and reads from at step t' are also a function of these variables.

Let $w_i^{t'}(x_{i_1}, x_{i_2}, \dots, x_{i_k})$ be the *write index function* of P_i at step $t' \leq t+1$, for inputs in I_t . The value of $w_i^{t'}$ is the index of the cell into which P_i writes, or zero if the processor does not write at step t' , for any setting of the input variables. We know that $w_i^{t'}$ is defined on increasing tuples over S_t , since for every input $\hat{x}$ in I_t , $\hat{x}_{i_1} < \hat{x}_{i_2} < \dots < \hat{x}_{i_k}$.

Similarly, let $r_{i,j}^{t+1}$ be the *jth read index function* of P_i at step $t+1$ for inputs in I_t . The value of $r_{i,j}^{t+1}$ is the index of the j th cell that P_i reads from at step $t+1$, for inputs in I_t . $r_{i,j}^{t+1}$ is also defined on increasing tuples over S_t . We use the following lemma by Meyer auf der Heide and Wigderson [MW], which is based on a theorem by Erdős and Rado [ER]:

LEMMA 7. *Let $\mathcal{F}$ be a finite collection of functions defined on increasing tuples over S , for S infinite. The functions in $\mathcal{F}$ may be functions of differing numbers of variables. Then there exists an infinite $\tilde{S} \subseteq S$ such that, restricted to increasing tuples over $\tilde{S}$, each $f \in \mathcal{F}$ is one-to-one on the variables it depends on. Furthermore, two distinct $f, f' \in \mathcal{F}$ have the property that, restricted to increasing tuples over $\tilde{S}$, they either have disjoint ranges, or are identical.*

We apply Lemma 7 to the collection $\mathcal{F}$, where

$$\mathcal{F} = \{w_i^{t'} \mid 1 \leq t' \leq t+1, 1 \leq i \leq n\} \cup \{r_{i,j}^{t+1} \mid 1 \leq i \leq n, 1 \leq j \leq t+1\}.$$

Let $\tilde{S}$ be the infinite subset of S_t with the properties mentioned in Lemma 7. For $f \in \mathcal{F}$, when f takes on the value zero for some input, it means that processors using f at some step choose not to write (or read) at that step for that input. We would like to simplify the situation by ensuring that, at a particular timestep, processors either write for all inputs or do not write for any input. We can do this by reducing $\tilde{S}$. For each $f \in \mathcal{F}$, if f is not constant when restricted to increasing tuples over $\tilde{S}$, let $\{v_1, v_2, \dots, v_k\}$ be the variables on which f depends. Since f is 1-1 on those variables, if zero is in the range of f , then there are unique values $\hat{v}_1, \hat{v}_2, \dots, \hat{v}_k$ such that setting $v_i = \hat{v}_i$ for $i = 1, 2, \dots, k$ makes $f = 0$. In this case, we remove the values $\hat{v}_1, \hat{v}_2, \dots, \hat{v}_k$ from $\tilde{S}$. Then the range of f when restricted to $\tilde{S}$ does not include zero. If this is done for all $f \in \mathcal{F}$, then for inputs in I_{t+1} , processors will either never write (read) or always write (read) at any given time step among the first $t+1$ steps.

Set $S_{t+1} = \tilde{S}$. If we also set $\Pi_{t+1} = \Pi_t$, and $\mathcal{C}_{t+1} = \mathcal{C}_t$, this defines a set of inputs I_{t+1} such that for inputs in this set, the read index and write index functions of all processors through step $t+1$ will have the properties mentioned in the lemma. The fact that $f, f' \in \mathcal{F}$ either have disjoint ranges or are identical means that, when a processor uses f as a write index function, it is communicating only with those processors that use f as a read index function at subsequent steps. Thus we have considerably simplified the pattern of communication in the first $t+1$ steps, by restricting consideration to inputs in I_{t+1} . In what follows, our goal is to define V_{t+1}^i for all processors P_i . To do this, we will add tuple systems to $\mathcal{C}_{t+1}$; this only reduces I_{t+1} .

We also set $Q_i = \{\{V_t^i\}\}$ for $1 \leq i \leq n$. The set Q_i will represent the variables that P_i knows about at the end of step $t+1$. We will add to Q_i sets of variables that P_i could learn at step $t+1$, by considering the access functions of P_i in $\mathcal{F}$. At the conclusion of this process, Q_i will be a collection of at most $t+2$ sets of variables. Each set in Q_i will be totally ordered in Π_t , and the state of P_i at time $t+1$ will, by construction, depend only on variables in the union of those sets. We will then refine Π_{t+1} in such a way that this union is totally ordered in Π_{t+1} , thus ensuring our conditions after step $t+1$ and completing the induction step.

We consider each function $f \in \mathcal{F}$ in turn, and deal with processors that use this f as an access function.

Case 1: f constant. If f is constant, then it represents the index of a single cell that processors using f read from or write to. Let t' be the unique step at which processors write into that cell. If no such step exists, let $\mathcal{W} = \emptyset$; otherwise, let $\mathcal{W} = \{P_i \mid w_i^{t'} = f\}$. Let $\mathcal{R}_j = \{P_i \mid r_{i,j}^{t+1} = f\}$. $\mathcal{R}_j$ is the set of processors that access cell f as the j th of the $t+1$ cells that they read at step $t+1$.

If $\mathcal{W} = \emptyset$, then all processors in $\mathcal{R}_j$, $1 \leq j \leq t+1$, read zero from cell f at step $t+1$, and so no information is transmitted. If $\mathcal{W} \neq \emptyset$, the value written into cell f at step t' must be a function of variables that the processors in $\mathcal{W}$ “have in common”; that is, it must depend only on variables in $V = \bigcap_{P_i \in \mathcal{W}} V_{t'-1}^i$. Recall that processors

using f always write on all inputs in I_{t+1} ; if the value depended on some variable not in V , then there would be an input for which two processors in $\mathcal{W}$ attempt to write different values into the cell indexed by f , a violation of the COMMON model.

Thus, the state of any processor $P_i \in \mathcal{R}_j$ at time $t+1$ may be affected by variables in V , for any input in I_{t+1} . The variables in V are totally ordered in Π_t , and $|V| \leq 3^{t'-1}$. For each reading processor $P_i \in \mathcal{R}_j$, $1 \leq j \leq n$, we add V to Q_i .

Case 2: f not constant. Let $y_1, y_2, \dots, y_\ell$ be the variables that f depends on (considered as free variables, not members of X). We note that if f is used as a write index function by some processor at time t' , then $\ell \leq 3^{t'-1}$. Let $\mathcal{R}_j = \{P_i | r_{i,j}^{t+1} = f\}$, and $\mathcal{W}_{t'} = \{P_i | w_i^{t'} = f\}$, for $t' \leq t+1$.

Consider $P_i \in \mathcal{W}_{t'}$ and $P_j \in \mathcal{R}_h$. To say that P_i uses f as a write index function at time t' means that $w_i^{t'}$ is equal to f with $y_1, y_2, \dots, y_\ell$ replaced by some variables in $V_{t'-1}^i$. If the replacement variables for P_i are exactly the same as the replacement variables for P_j using f as its h th read function, then P_j at step $t+1$ will read as its h th cell the same cell that P_i wrote into at time t' . Let $\mathcal{W}'_j$ be the set of processors in $\mathcal{W}_{t'}$ that have the same set of replacement variables for f as P_j . As in case 1, we may set $V = \bigcap_{P_k \in \mathcal{W}'_j} V_{t'-1}^k$, and note that the value written is a function of V . As before, it is important that processors in $\mathcal{W}'_j$ always write on all inputs in I_{t+1} . We also have $|V| \leq 3^{t'-1}$. We add V to Q_j and, having dealt with what P_j could learn using function f , remove P_j from $\mathcal{R}_h$. If we do this for all such P_i, P_j , then no processor in any $\mathcal{W}_{t'}$ will have exactly the same replacement variables as any processor in any $\mathcal{R}_h$.

For the remaining processors, our goal is to ensure that no processor using f as a read index function at step $t+1$ accesses the same cell as any processor using f as a write index function at step $t+1$ or earlier. In order to ensure this, we add tuple systems to $\mathcal{C}_{t+1}$. Again, consider $P_i \in \mathcal{W}_{t'}$ and $P_j \in \mathcal{R}_h$. Let $x_{i_1}, x_{i_2}, \dots, x_{i_\ell}$ be the replacement variables of P_i for $y_1, y_2, \dots, y_\ell$, and $x_{j_1}, x_{j_2}, \dots, x_{j_\ell}$ be the replacement variables of P_j . At most two variables are equal for any input in I_{t+1} . If there are positions $1 \leq a < b \leq \ell$ such that $i_a \neq j_a$ and $i_b \neq j_b$, then for every input in I_t , either $x_{i_a} \neq x_{j_a}$ or $x_{i_b} \neq x_{j_b}$. It follows that, for every input in I_t , the fact that f is one-to-one ensures that $f(x_{i_1}, \dots, x_{i_\ell}) \neq f(x_{j_1}, \dots, x_{j_\ell})$, and so P_j will read a different cell from the one into which P_i wrote.

On the other hand, suppose there is only one position a such that $i_a \neq j_a$. If, for some input $\hat{x}$, P_j reads the same cell as P_i , it “knows” that $\hat{x}_{i_a} = \hat{x}_{j_a}$. We would like the PRAM to be unable to answer after $t+1$ steps, for inputs in I_{t+1} . To ensure that P_i and P_j access different cells, we must ensure that $x_{i_a} \neq x_{j_a}$ for all inputs in I_{t+1} . If this is done, then $P_i \in \mathcal{R}_h$ will read the value zero for all inputs in I_{t+1} , and it will gain no additional information as a result of that read.

Let A_f be the set of all tuples $(x_{i_1}, \dots, x_{i_\ell})$ that are the replacement variables for f of some processor P_i in some $\mathcal{W}_{t'}$ ($t' \leq t+1$). Similarly, B_f is the set of all tuples $(x_{j_1}, \dots, x_{j_\ell})$ that are the replacement variables for f of some processor P_j in some $\mathcal{R}_h$ ($h \leq t+1$). $\{A_f, B_f\}$ forms a tuple system, since our actions above ensured that no processor in $\mathcal{W}'_t$ has the same set of replacement variables as any processor in $\mathcal{R}_h$. Furthermore, the pairs of variables that we require to be distinct are precisely those covered by this tuple system. Adding $\{A_f, B_f\}$ to $\mathcal{C}_{t+1}$ will ensure (by the definition of I_{t+1}) that no reader using f as a read index function at time $t+1$ will read a cell written into by any writer using a different write index function or a different set of replacement variables than the reader. This concludes the description of what is done in the case of one function $f \in \mathcal{F}$.

After this has been done for all f , $\mathcal{C}_{t+1}$ has been defined. Each processor has $t+1$ read index functions at time $t+1$, and hence can contribute at most $t+1$ tuples

to some set B_f as described above. Each processor also has at most $t + 1$ write index functions, at steps $1, 2, \dots, t + 1$, and so can contribute at most $t + 1$ tuples to sets A_f above.

It follows that $\sum_{f \in \mathcal{F}} |B_f| \leq n(t + 1)$ and $\sum_{f \in \mathcal{F}} |A_f| \leq n(t + 1)$. Thus

$$\begin{aligned} \text{size}(\mathcal{C}_{t+1}) &\leq \text{size}(\mathcal{C}_t) + \sum_{f \in \mathcal{F}} (|A_f| + |B_f|) \\ &\leq (t + 1)^2 n + 2(t + 1)n \leq (t + 2)^2 n \text{ as required.} \end{aligned}$$

At most one set V is added to Q_i for each read index function that P_i uses at time $t + 1$, so $|Q_i| \leq t + 2$. Furthermore, a set $V \in Q_i$ added as a result of a read index function that reads a cell written into at step t' is of size at most $3^{t'-1}$. It follows that Q_i contains at most one set of size $3^{t'-1}$ for $t' < t + 1$, and at most two sets of size 3^t (one is V_t^i , and the other may be added as a result of reading a cell written at step $t + 1$). Thus, if we set $V_{t+1}^i = \bigcup_{V \in Q_i} V$, then $|V_{t+1}^i| \leq 2 \cdot 3^t + \sum_{t' < t+1} 3^{t'-1} \leq 3^{t+1}$ as required.

We must still ensure that V_{t+1}^i is ordered in Π_{t+1} . Each $V \in Q_i$ is ordered in Π_t , so each antichain in Π_t contains at most $t + 2$ elements that appear in sets $V \in Q_i$. Let R be the set of pairs (x_j, x_k) such that x_i, x_j are in the same antichain of Π_t but in different sets of some Q_i . By counting, we can show that $|R| \leq n3^{2t+1}$. If we refine Π_{t+1} so that every pair $(x_j, x_k) \in R$ is comparable in Π_{t+1} , then $V_{t+1}^i = \bigcup_{V \in Q_i} V$ will be totally ordered in Π_{t+1} , as required.

LEMMA 8. *There exists Π_{t+1} a refinement of Π_t such that all pairs in R are comparable in Π_{t+1} , and further, Π_{t+1} satisfies the required conditions on k_{t+1}, a_{t+1} .*

Proof. Each pair in R consists of two variables in the same nontrivial antichain of Π_t . At most $k_t/3^{t+2}$ antichains contain more than $3^{3t+3}n/k_t$ pairs in R , because $|R| \leq n3^{2t+1}$. For each such antichain, choose an arbitrary total order for its elements, in effect converting the nontrivial antichain to s_t trivial antichains. This increases a_{t+1} by at most $k_t s_t/3^{t+2} \leq n/3^{t+2}$.

No remaining nontrivial antichain contains more than $3^{3t+3}n/k_t$ pairs in R . From each nontrivial antichain, take the $s_t/3^{t+2}$ elements that are in the most pairs in R , choose an arbitrary total order for them, and make them all larger than every element of the antichain. As a result, any element in any nontrivial antichain is in less than $3^{4t+6}n/k_t s_t$ pairs of R . If this were not true for some antichain, then counting occurrences of deleted elements in pairs would lead to a total of $3^{3t+4}n/k_t$ occurrences, a contradiction since each pair contains two elements. The deletions increase a_{t+1} by at most $k_t s_t/3^{t+2} \leq n/3^{t+2}$.

The following theorem by Hajnal and Szemerédi [HS] is useful at this point.

THEOREM 9. *Let $R \subseteq X^2$ be a set of pairs such that no element of X occurs in Δ or more pairs in R . Then there is a number s such that X can be partitioned into Δ classes of size s or $s + 1$, such that no pair in R contains two elements in the same class.*

Then, by Theorem 9, we can partition each nontrivial antichain into $3^{4t+6}n/k_t s_t$ classes such that no pair in R contains two elements from the same class, and each class is of size s or $s + 1$. We remove one element from each classes of size $s + 1$, choose an arbitrary total order for the removed elements, and make each larger than everything in their former antichain. Within each antichain, we choose an arbitrary total order for the classes, thus converting each antichain into $3^{4t+6}n/k_t s_t$ smaller nontrivial antichains, plus at most one new trivial antichain for each class. Thus

$$k_{t+1} \leq k_t \left\lceil \frac{3^{4t+6}n}{k_t s_t} \right\rceil \leq 3^{4t+7} k_t \text{ since } a_t \leq \frac{2n}{3}$$

$$\leq 3^{2(t+1)^2 + 5(t+1) + 3} \text{ as required.}$$

We can thus define an appropriate Π_{t+1} . In the last step, we may have increased a_{t+1} by one element for each new class, and so in total,

$$\begin{aligned} a_{t+1} &\leq a_t + 2 \left(\frac{n}{3^{t+2}} \right) + k_{t+1} \\ &\leq \frac{n}{3} - \frac{n}{3^{t+2}} + 3^{2(t+1)^2 + 5(t+1) + 4} \text{ as required.} \end{aligned}$$

This concludes the description of the construction of objects of index $t + 1$. We continue this construction to index $T = \epsilon\sqrt{\log n}$, for ϵ a suitably chosen small constant.

Let us choose ϵ sufficiently small ($\epsilon = 1/4$ will do), and n sufficiently large, so that the following conditions hold:

$$n > 3^{2T^2 + 6T + 8}, \quad (10)$$

$$3^T \leq \frac{(n/3)^{1/6}}{\log(n/3)}, \quad (11)$$

$$k_T \leq \frac{(n/3)^{1/6}}{\log(n/3)}, \quad (12)$$

$$(T + 1)^2 < \frac{1}{12} \log\left(\frac{n}{3}\right). \quad (13)$$

Condition (10) ensures that $a_t \leq 2n/3$ for $t \leq T$, which is required in order that the construction be possible up to index T . Conditions (11) and (12) ensure that Theorem 4 is applicable to the tuple system $\mathcal{C}_T$ and the nontrivial antichains of Π_T . To apply Theorem 4, we must know how many elements of X we can select so that no two are in the same nontrivial antichain (this is k_T , the number of nontrivial antichains) and we must know an upper bound on tuple length (this is 3^T). By the theorem, any tuple system that covers all pairs in all nontrivial antichains of Π_T must be of size at least $n(\frac{1}{4} \log(n/3) - \frac{1}{2} \log((n/3)^{1/3} / \log^2(n/3)) - \log \log(n/3))$, or at least $(1/12)n \log(n/3)$. But by condition (13), $\text{size}(\mathcal{C}_T) < (1/12)n \log(n/3)$. Thus at least one pair of variables in some antichain of Π_T remains uncovered by $\mathcal{C}_T$, thereby ensuring a lower bound of $T + 1$ steps as discussed above. This concludes the proof of Theorem 6. $\square$

We conclude by mentioning some implications of this proof for separating the power of PRAM models with the same (finite) amount of shared memory. We can define a bounded version of the element distinctness problem, where the variables $\{x_i\}$ are chosen from $[m]$ for some $m \in \mathbb{N}$. An ARBITRARY PRAM with m shared memory cells can solve this version of element distinctness in $O(1)$ steps, by using the algorithm of Theorem 1. It is also possible to prove a bound of $\Omega(\sqrt{\log n})$ steps to solve bounded element distinctness on a COMMON PRAM with m memory cells in exactly the same fashion as given here, by using the finite version of the Erdős-Rado theorem from [ER2] to prove a finite version of Lemma 7. However, this only works for m larger than a very rapidly growing function of n , since a lower bound on the size of the set S mentioned in Lemma 7 is required, and this bound is expressed by a recurrence involving generalized Ramsey numbers. We can conclude, however, that for n sufficiently large, there exists $m = m(n)$ such that there is an $\Omega(\sqrt{\log n})$ separation between the COMMON and ARBITRARY PRAMs with m memory cells. This also held true for the $\Omega(\log \log \log n)$ separation given in [FMW], except that in that proof m was only required to grow as fast as $2^{n^{0(1)}}$.

REFERENCES

- [B] C. BERGE. *Graphs and Hypergraphs*. North-Holland, Amsterdam, 1973.
- [CDR] S. COOK, C. DWORK, AND R. REISCHUK. *Upper and lower bounds for parallel random access machines without simultaneous writes*, SIAM J. Comput., 15 (1986), pp. 87-97.
- [ER] P. ERDŐS AND R. RADO. *A combinatorial theorem*, J. London Math. Soc., 25 (1950), pp. 376-382.
- [ER2] ———, *Combinatorial theorems on classifications of subsets of a given set*, Proc. London Math. Soc., 3 (1952), pp. 417-439.
- [FK] M.L. FREDMAN AND J. KOMLÓS. *On the size of separating systems and families of perfect hash functions*, SIAM. J. Algebraic Discrete Methods, 5 (1984), pp. 61-68.
- [FMW] F.E. FICH, F. MEYER AUF DER HEIDE, AND A. WIGDERSON. *Lower bounds for parallel random access machines with unbounded shared memory*, Adv. in Comput., 4 (1987), pp. 1-15.
- [FRW] F.E. FICH, P. RAGDE, AND A. WIGDERSON. *Simulations Among Concurrent-Write PRAMs*, Algorithmica, 3 (1988), pp. 43-51.
- [H] G. HANSEL. *Nombre minimal de contacts de fermeture nécessaires pour réaliser une fonction booléenne symétrique de n variables*, Comptes Rendus Acad. Sci. Paris, 258 (1964), pp. 6037-6040.
- [HS] A. HAJNAL AND E. SZEMERÉDI. *Proof of a conjecture of P. Erdős*, Combin. Theory Appl., 2 (1970), pp. 601-623.
- [K] L. KUCERA. *Parallel computation and conflicts in memory access*, Inform. Process. Lett., 14 (1982), pp. 93-96.
- [MW] F. MEYER AUF DER HEIDE AND A. WIGDERSON. *The complexity of parallel sorting*, Proc. 26th Annual IEEE Symposium on Foundations of Computer Science, pp. 532-540.
- [P] N. PIPPENGER. *An information-theoretic method in combinatorial theory*, J. Combin. Theory, ser. A, 23 (1977), pp. 99-104.

ON THE INFLUENCE OF SINGLE PARTICIPANT IN COIN FLIPPING SCHEMES*

BENNY CHOR† AND MIHÁLY GERÉB-GRAUS‡

Abstract. This paper proves that in a one-round fair coin flipping scheme with n participants, either the average influence of all participants is at least $3/n - o(1/n)$, or there is at least one participant whose influence is $\Omega(n^{-5/6})$.

Key words. voting, influence, distributed in flip

AMS(MOS) subject classifications. 05A20, 68R05, 90D10

1. Introduction. A single round voting scheme with n participants is a function $v: \{-1, 1\}^n \rightarrow \{-1, 1\}$. A *fair* voting scheme (coin flip) is one whose expected value, under the uniform probability distribution of the input space, is zero.

A fair voting scheme is considered robust if none of the participants can significantly bias the outcome, first, by observing the choices of other participants, and then, by choosing (deterministically) its bit instead of picking its bit at random. Formally the influence of participant i is the number of points where a change in the i th participant bit causes a change in the value of v , divided by the total number of points, that is,

$$\frac{|\{(c_1, \dots, c_{i-1}, c_{i+1}, \dots, c_n): v(c_1, \dots, c_{i-1}, -1, c_{i+1}, \dots, c_n) \neq v(c_1, \dots, c_{i-1}, +1, c_{i+1}, \dots, c_n)\}|}{2^{n-1}}.$$

The least robust system is the exclusive-or scheme, where each participant can completely control the output. Surprisingly, the most robust is not the majority scheme, where each participant has a $\Theta(1/\sqrt{n})$ influence. Ben-Or and Linial [BL], who proposed the question, gave a fair voting scheme where each participant has $O(\log n/n)$ influence. (They partition the participants into roughly $\log n - \log \log n$ groups of size. The output is 1, if there is a group, where everybody chooses 1; otherwise it is -1 .) Ben-Or and Linial conjecture that this system is asymptotically optimal.

The problem has the following combinatorial interpretation. Given a two-coloring of all 2^n vertices of the n -dimensional hypercube, with 2^{n-1} vertices colored red and 2^{n-1} vertices colored blue, look at all 2^{n-1} pairs of vertices that differ only in the i th coordinate. What is the maximum number of such pairs (for $i = 1, 2, \dots, n$) that have their two members colored differently?

In this paper we prove that in any fair voting scheme with n participants, either the average influence of all participants is at least $3/n - o(1/n)$, or there is at least one participant whose influence is $\geq \Omega(n^{-5/6})$. Previously, the best lower bound was the intuitive $1/n$.

2. Proof. We prove that if there is no participant whose influence is $\Omega(n^{-5/6})$ in an n -participants fair voting scheme, then the sum of the n influences is at least $3 - o(1)$. The assumption means that there is a function f with $\lim_{n \rightarrow \infty} f(n) = \infty$ so that for an infinite sequence of n 's, the influence of all participants is $\leq n^{-5/6}/f(n)$. In addition, there

* Received by the editors October 1, 1987; accepted for publication February 12, 1988. This research was supported by National Science Foundation grant MCS81-21431 at Harvard University, Cambridge, Massachusetts 02138.

† Computer Science Department, Technion, Haifa 32000, Israel. Most of this work was performed while the author was at Harvard University, Cambridge, Massachusetts 02138.

‡ Aiken Computation Laboratory, Harvard University, 33 Oxford St., Cambridge, Massachusetts 02138.

is a constant $d > 0$ so that for all other n 's not in the sequence above, at least one participant has influence $\geq n^{-5/6}/d$. Thus, for the proof, we need only consider n 's in the infinite sequence.

We say that a voting scheme v is *monotone* if for every

$$\bar{c} = (c_1, \dots, c_{i-1}, c_{i+1}, \dots, c_n) \in \{-1, 1\}^{n-1},$$

$v(c_1, \dots, c_{i-1}, -1, c_{i+1}, \dots, c_n) \leq v(c_1, \dots, c_{i-1}, 1, c_{i+1}, \dots, c_n)$. Ben-Or and Linial showed that any voting scheme can be modified to a monotone one, where the influence of every participant is not increased, and the expected value of the scheme (under the uniform distribution) is preserved. Thus it is enough to consider monotone schemes.

Let us extend v from a function of $\{-1, 1\}^n$ to an n variable multilinear polynomial as follows: On the coin flipping sequence $(c_1, \dots, c_n)$ the value of the polynomial

$$p_{\bar{c}}(x_1, \dots, x_n) = \prod_{i=1}^n (1 + x_i c_i)$$

is 2^n , but on any other input sequence its value is zero. The sum

$$\frac{1}{2^n} \sum_{\bar{c} \in \{-1, 1\}^n} v(\bar{c}) \cdot p_{\bar{c}}(x_1, \dots, x_n)$$

will be equal to the function v on $\{-1, +1\}^n$.

Let P denote the set of participants, $\{1, 2, \dots, n\}$. Any multilinear polynomial in the variables $x_1, \dots, x_n$ can be written in a unique canonical form

$$\sum_{S \subset P} a_S x_S$$

where a_S are real numbers and

$$x_S = \prod_{i \in S} x_i.$$

Now $\alpha_{\emptyset} = 0$ because

$$0 = E(v(\bar{c})) = E\left(\sum_{S \subset P} a_S x_S\right) = a_{\emptyset} + E\left(\sum_{\emptyset \neq S \subset P} a_S x_S\right) = a_{\emptyset}.$$

(Inputs that differ only in x_i cancel each other.) For $S, S' \subset P$, $x_S x_{S'} = x_{S \Delta S'}$, and thus for $S \neq S'$ $E(x_S x_{S'}) = 0$. Hence

$$1 = E(v^2(\bar{c})) = \sum_{S \subset P} a_S^2 + 2E \sum_{S \neq S' \subset P} a_S a_{S'} x_S x_{S'} = \sum_{S \subset P} a_S^2,$$

and so (from now on, summing over S will mean over all $S \subset P$)

$$\sum_S a_S^2 = 1.$$

For any participant i , the function v can be written as $g_i + x_i f_i$, where

$$g_i = \sum_{S \ni i} a_S x_S$$

and

$$f_i = \sum_{S \ni i} a_S x_{S - \{i\}}.$$

So for every $\bar{c} = (c_1, \dots, c_{i-1}, c_{i+1}, \dots, c_n) \in \{-1, 1\}^{n-1}$,

$$v(c_1, \dots, c_{i-1}, -1, c_{i+1}, \dots, c_n) = g_i(\bar{c}) - f_i(\bar{c}),$$

$$v(c_1, \dots, c_{i-1}, 1, c_{i+1}, \dots, c_n) = g_i(\bar{c}) + f_i(\bar{c}).$$

Since the function v has ± 1 values on $\{-1, 1\}^n$, $2g_i(\bar{c})$, the result of adding the last two lines is either -2 , 0 , or $+2$. We thus conclude that either $g_i(\bar{c}) = 0$ and $|f_i(\bar{c})| = 1$ or $|g_i(\bar{c})| = 1$ and $f_i(\bar{c}) = 0$. Furthermore, the monotonicity of v implies that if $f_i(\bar{c}) \neq 0$ then $f_i(\bar{c}) = 1$. The i th player can determine the outcome at $\bar{c}$ if and only if $f_i(\bar{c}) = 1$. From this it follows that the influence of the i th player is the probability that $f_i(\bar{c}) = 1$. Since $f_i(\bar{c})$ can only take the values zero and 1, $\Pr(f_i(\bar{c}) = 1) = E(f_i(\bar{c}))$. From the representation

$$f_i = a_{\{i\}} + \sum_{S \ni i, |S| \geq 2} a_S x_{S - \{i\}}$$

it follows that $E(f_i(\bar{c})) = a_{\{i\}}$ is the influence of player i . Since $f_i(\bar{c}) = f_i^2(\bar{c})$ and $E(f_i^2(\bar{c})) = \sum_{S \ni i} a_S^2$, we get $a_{\{i\}} = \sum_{S \ni i} a_S^2$. The sum of the influences is then

$$\begin{aligned} \sum_{i=1}^n a_{\{i\}} &= \sum_{i=1}^n \sum_{S \ni i} a_S^2 \\ &= \sum_{j=0}^n j \sum_{|S|=j} a_S^2 \\ &\geq \sum_{j=0}^2 j \sum_{|S|=j} a_S^2 + 3 \sum_{|S|>2} a_S^2 \\ &= 3 \sum_S a_S^2 - 3a_{\emptyset}^2 - 2 \sum_{|S|=1} a_S^2 - \sum_{|S|=2} a_S^2. \end{aligned}$$

Since $\sum_S a_S^2 = 1$ and $a_{\emptyset}^2 = 0$ it suffices to show that

$$\sum_{|S|=1} a_S^2, \sum_{|S|=2} a_S^2 = o(1).$$

By assumption, the influence of every participant is $o(n^{-5/6})$. Thus for every i , $a_{\{i\}} = o(n^{-5/6})$, $a_{\{i\}}^2 = o(n^{-5/3})$, and so

$$\sum_{|S|=1} a_S^2 = o(1).$$

In the remaining part of the proof, we show that the second sum $\sum_{|S|=2} a_S^2$ is $o(1)$ as well.

We denote by A_r ($r = 1, 2, \dots, n$) the average of size r coefficients

$$A_r = 1 / \binom{n}{r} \sum_{|R|=r} |a_R|.$$

Combining

$$\sum_{|R|=r} a_R^2 \leq 1$$

and the Cauchy-Schwartz inequality

$$\left(\sum_{i=1}^m |b_i| \right)^2 \leq m \sum_{i=1}^m b_i^2,$$

we get

$$A_r \leq \binom{n}{r}^{-1/2}.$$

Let k be an arbitrary integer in the range $3 \leq k \leq n$. Let $S \subset \{1, 2, \dots, n\}$ be a set of size k with $S \ni i$. We calculate the conditional expected value of f_i under the condition $c_j = -\text{sign}(a_{i,j})$ for all $j \in S, j \neq i$. Since f_i is nonnegative this expected value has to be nonnegative. As

$$\begin{aligned} f_i &= \sum_{S \ni i} a_S x_{S - \{i\}} \\ &= a_{\{i\}} + \sum_{j \neq i, j \in S} a_{\{i,j\}} x_{\{j\}} + \sum_{|R| \geq 3, i \in R \subset S} a_R x_{R - \{i\}}, \end{aligned}$$

the contribution of the second term to the conditional expected value is $-\sum_{j \in S} |a_{i,j}|$. For every set R (for $i \in R \subset S$ of size ≥ 3), the contribution of $a_R x_{R - \{i\}}$ to the conditional expected value sum is at most $|a_R|$. The sum of all contributions must be nonnegative, that is,

$$0 \leq a_{\{i\}} - \sum_{i \neq j \in S} |a_{\{i,j\}}| + \sum_{i \in R \subset S, |R| \geq 3} |a_R|.$$

For every r , the number of sets $R \subset S$ of size r with a_R occurring in the last inequality is exactly $\binom{k-1}{r-1}$. Similarly, there are exactly $k-1$ indices j 's with $i \neq j \in S$. Thus, by summing this inequality over all sets S of size k and all choices of i ($i = 1, 2, \dots, n$), and averaging, we get

$$\begin{aligned} (1) \quad (k-1)A_2 &\leq A_1 + \sum_{r=3}^k \binom{k-1}{r-1} A_r \\ &\leq A_1 + \sum_{r=3}^k \binom{k-1}{r-1} \bigg/ \binom{n}{r}^{1/2}. \end{aligned}$$

It is not hard to check that for $k < \frac{1}{2}n^{1/3}$, the sequence $\binom{k-1}{r-1}/\binom{n}{r}^{1/2}$ is monotonically decreasing in r , where $3 \leq r < k$. For k in this range we have

$$\begin{aligned} (2) \quad \sum_{r=3}^k \binom{k-1}{r-1} \bigg/ \binom{n}{r}^{1/2} &\leq k \cdot \binom{k-1}{4} \bigg/ \binom{n}{5}^{1/2} \\ &= O\left(\frac{k^5}{n^{5/2}}\right). \end{aligned}$$

Also

$$(3) \quad \binom{k-1}{3} \bigg/ \binom{n}{4}^{1/2}, \quad \binom{k-1}{2} \bigg/ \binom{n}{3}^{1/2} = O\left(\frac{k^2}{n^{3/2}}\right).$$

Substituting $k = n^{1/3}/\sqrt{f(n)}$ in (1) and using (2), (3), and $A_1 \leq n^{-5/6}/f(n)$, we get

$$\frac{n^{1/3}}{f^{1/2}(n)} A_2 \leq O\left(\frac{1}{n^{5/6}f(n)}\right)$$

and so

$$A_2 = O\left(\frac{1}{n^{7/6}f^{1/2}(n)}\right).$$

Also for every i, j , $|a_{\{i,j\}}| \leq a_{\{i\}}$ because otherwise $E(f_i | (c_j = -\text{sign}(a_{\{i,j\}}))) = a_{\{i\}} - |a_{\{i,j\}}|$ would be negative. So for every i, j , $|a_{i,j}| \leq n^{-5/6}/f(n)$.

The expression

$$\sum_{|S|=2} a_S^2$$

is the sum of the square of $O(n^2)$ positive numbers $|a_S|$; all are $\leq n^{-5/6}/f(n)$. Since their sum $\sum_{|S|=2} |a_S|$ is $\binom{n}{2} A_2 = O(n^{5/6}/f^{1/2}(n))$, the maximum value for the sum of squares is attainable when as many $|a_S|$ as possible are at their maximum value, $O(n^{-5/6}/f(n))$, and the rest are zero. The number of such nonzero a_S is at most $O(n^{5/3}f^{1/2}(n))$, and so

$$\sum_{|S|=2} a_S^2 = o(1). \quad \square$$

3. Concluding remarks. The proof can be simply modified to yield the following related result. Either the average influence of all participants is at least $2/n - o(1/n)$, or there is at least one participant whose influence is $\Omega(n^{-1/2})$. We prove this by noting that $a_{\{i\}} = o(n^{-1/2})$ is sufficient to guarantee

$$\sum_{|S|=1} a_S^2 = o(1),$$

and using

$$\sum_{i=1}^n a_{\{i\}} \geq 2 \sum_S a_S^2 - 2a_{\emptyset}^2 - \sum_{|S|=1} a_S^2.$$

From the proof we also get that in an unfair coin flipping scheme there is a participant whose influence is at least $(3 - 3a_{\emptyset}^2 - o(1))/n$, where $a_{\emptyset}$ is the bias of the scheme if everybody plays correctly.

Alon [Al] used the eigenvalues and eigenvectors of the n -dimensional hypercube to show independently that for any two-coloring of the vertices, with the same number of vertices colored by each of the two colors, there is a coordinate i with $(2/n - o(1/n)) \cdot 2^{n-1}$ pairs of vertices colored differently. This corresponds to our result of the existence of a participant with $(2/n - o(1/n))$ influence. The proof using the eigenvector method is essentially equivalent to our proof, but it has the potential advantage of being applicable to graphs other than the hypercube.

REFERENCES

- [Al] N. ALON, private communication, 1987.
- [BL] M. BEN-OR AND N. LINIAL, *Collective coin flipping, robust voting schemes and minima of Banzhaf values*, in Proc. 26th Annual ACM Symposium on Theory of Computing, 1985, pp. 408–416.

MATROIDS AND SUBSET INTERCONNECTION DESIGN*

DING-ZHU DU† AND ZEVI MILLER‡

Abstract. A problem arising in the design of vacuum systems and having applications to some natural problems of interconnection design is described as follows.

(1) Given a set X and subsets X_i, Y_i of $X, i = 1, \dots, n$, satisfying $X_i \cap Y_i = \emptyset$, find a graph G with vertex set X and the minimum number of edges such that for any i , the subgraph induced by $X \setminus Y_i$ has a connected component containing X_i .

Two other problems related to this one are the following ones.

(2) Given a set X and subsets $X_1, X_2, \dots, X_n$ such that $X = \cup_{i=1}^n X_i$, find a graph G with vertex set X and the minimum number of edges such that for any i the subgraph G_i induced by X_i in G is connected.

(3) Given a set X and subsets $X_1, X_2, \dots, X_n$ such that $X = \cup_{i=1}^n X_i$, find a graph G with vertex set X and the minimum number of edges such that for any subset I of $\{1, \dots, n\}$, the subgraph induced by $\cap_{i \in I} X_i$ is connected.

This paper shows that (3) is polynomial-time solvable while (1) and (2) are NP-complete. Also, some heuristics for (1) and (2) are given. The solution of (3) is an interesting application of matroid theory.

Key word. matroids

AMS(MOS) subject classifications. 05B35, 05C65, 05C35

1. Introduction. In a gas, liquid, or vacuum system, valves are used to make various subsets of the system connected. Here, we consider the following problem: Given required connected subsets, how do we design the system by using the minimum possible number of valves?

This problem appeared first in the design of vacuum systems [1], [2]. More generally, it relates to any situation where specified subsets of a superset must be made internally connected. It can be formalized as follows.

PROBLEM 1. *Given a set X and subsets $X_1, Y_1, \dots, X_n, Y_n$ of X satisfying $X_i \cap Y_i = \dots = X_n \cap Y_n = \emptyset$, find a graph G with the vertex set X and the minimum number of edges such that*

(P1) *For every $i = 1, \dots, n$, the subgraph G_i induced by $X \setminus Y_i$ has a connected component containing X_i .*

The elements of our vacuum system correspond to the points of X . A valve corresponds to an edge, and the connected subsets correspond to the subgraphs G_i . Our working requirement is that G_i should interconnect the elements that form X_i and should avoid the elements that form the Y_i .

This problem has been shown to be NP-complete [4] and remains NP-complete for the special case $X_1 \cup Y_1 = \dots = X_n \cup Y_n = X$. In this special case, we can assume $X = \cup_{i=1}^n X_i$ without loss of generality since every graph satisfying (P1) has no edge incident to a point in $X \setminus \cup_{i=1}^n X_i$ in this case. Thus, the problem in this case can be restated as follows.

* Received by the editors August 25, 1986; accepted for publication May 13, 1988.

† Mathematical Sciences Research Institute, 1000 Centennial Drive, Berkeley, California 94720. Part of this work was performed while the author was at the Department of Mathematics, University of California at Santa Barbara, Santa Barbara, California 93106 and was supported in part by the National Science Foundation under grants DCR83-12472 and 8210790.

‡ Department of Mathematics and Statistics, Miami University, Oxford, Ohio 45056 and Department of Electrical Engineering and Computer Science, University of California, Berkeley, California 94720. This work was partially supported by the Office of Naval Research under grant N00014-85-K-0621.

PROBLEM 2. Given a set X and subsets $X_1, X_2, \dots, X_n$ with $X = \bigcup_{i=1}^n X_i$, find a graph G with vertex set X and the minimum number of edges such that

(P2) Every subgraph G_i induced by X_i is connected.

Since neither Problem 1 nor Problem 2 is polynomial-time solvable, we have to study heuristics for them. A strategy to deal with an intractable problem is to find a related problem the solution of which could be a heuristic for the original one. We therefore introduce a new problem.

PROBLEM 3. Given a set X and subsets $X_1, X_2, \dots, X_n$ with $X = \bigcup_{i=1}^n X_i$, find a graph G with vertex set X and the minimum number of edges such that

(P3) For any subset I of $\{1, \dots, n\}$, the subgraph G_I induced by $\bigcap_{i \in I} X_i$ is connected. Here $G_i = G_{\{i\}}$.

The input size for each of these problems is therefore $O(n|X|)$, since we must store X_i and Y_i for each i , $1 \leq i \leq n$.

In this paper we will prove that Problem 3 is polynomial-time solvable. As an application, we obtain a polynomial-time solvable special case for Problem 2. These results will form § 2.

A heuristic is called *bounded* if it is within a constant factor from optimal. Although finding a bounded heuristic for Problem 2 is still open, we will show in § 3 that under the restriction that $|X_i| \leq 3$ for all i , Problem 2 remains NP-complete but has a bounded heuristic with factor two.

We let (P) denote a property. For simplicity, a graph satisfying (P) is also called a (P)-graph. For example, a graph satisfying (P2) is also called a (P2)-graph. We will sometimes specify our input by $(\cdot)$, so, for example, we will speak of a (P2)-graph for $(X_1, X_2, \dots, X_n)$, etc. A (P)-graph is *minimum* if it has the minimum number of edges over all (P)-graphs for the same input. For a graph G , $\|G\|$ denotes the number of edges of G . For a set Y , $|Y|$ denotes the number of elements of Y .

The operations on graphs are defined in the following way. Let $G = (V, E)$ and $G' = (V', E')$ be two graphs. Then, the union, the intersection, and the difference of G and G' are the graphs $G \cup G' = (V \cup V', E \cup E')$, $G \cap G' = (V \cap V', E \cap E')$, and $G \setminus G' = (V, E \setminus E')$, respectively.

A spanning tree for a set X is a graph with vertex set X , which is a tree. If $X \subset Y$ and T is a spanning tree for Y all of whose endpoints are in X , then T is a *Steiner tree* for X . (P1) then implies that G_i contains a Steiner tree for X_i , and (P2) implies that G_i contains a spanning tree for X_i . A problem on Steiner trees is usually more difficult than the corresponding problem on spanning trees. The allowed extra vertices make the search for minimality more difficult. The same thing happens here. We will make some remarks on that in the last section, and also show that there is a heuristic for Problem 1 within $2\sqrt{2n}$ from optimal.

Clearly, (P3) $\Rightarrow$ (P2) $\Rightarrow$ (P1). The paper is organized in this direction.

Finally, we make two conventions. Throughout this paper, we assume $|X_i| \geq 2$ for all i since X_i with $|X_i| = 1$ can always be deleted. We also assume from now on that every graph G we consider satisfies $G = \bigcup_{i=1}^n G_i$ since an edge not in $\bigcup_{i=1}^n G_i$ can always be deleted when we consider the above three problems.

2. Analysis of Problem 3. We first show that Problem 3 is polynomial-time solvable.

Define $I(x) = \{i: 1 \leq i \leq n, x \in X_i\}$ for any $x \in X$ and $I(H) = \bigcap_{x \in V} I(x)$ for a graph H with the vertex set $V \subseteq X$.

LEMMA 2.1. *A graph G satisfies (P3) if and only if for any two vertices x and y satisfying $I(x) \cap I(y) \neq \emptyset$, there is a path l from x to y in G such that $I(l) = I(x) \cap I(y)$.*

Proof. The “only if” part is trivial. For the “if” part, consider any subset I of $\{1, \dots, n\}$. We will show that G_I is connected. Let x and y be any two vertices of G_I . Then, there is a path l from x to y in G such that $I(l) = I(x) \cap I(y)$. Clearly, $G_I = \bigcap_{i \in I} G_i$ and $I(l) = (I(x) \cap I(y)) \supseteq I$. Thus $G_{I(l)} \subseteq G_I$, so l is in G_I . Therefore, G_I is connected. $\square$

We say that a (P3)-graph is minimal if it does not contain a proper subgraph satisfying (P3). For a graph G , an edge u of G is called a *bridge* if $G \setminus u$ has one more connected component than G does. The following lemma states a necessary and sufficient condition for a (P3)-graph to be minimal.

LEMMA 2.2. *A (P3)-graph G is minimal if and only if every edge u of G is a bridge in the subgraph $G_{I(u)}$.*

Proof. Note that a graph with a (P3)-subgraph must satisfy (P3). Thus, a (P3)-graph G is minimal if and only if for every edge u of G , $G \setminus u$ does not satisfy (P3), which, by Lemma 2.1, is equivalent to that u is a bridge in the subgraph $G_{I(u)}$. $\square$

The next lemma shows that a minimal (P3)-graph is in fact minimum.

LEMMA 2.3. *All minimal (P3)-graphs have the same number of edges.*

Proof. Let G and G' be two minimal (P3)-graphs. By symmetry, to show $\|G\| = \|G'\|$, it suffices to prove $\|G\| \leq \|G'\|$. We prove this by induction on $\|G \setminus G'\|$. For $\|G \setminus G'\| = 0$, it is obvious that $\|G\| \leq \|G'\|$. For $\|G \setminus G'\| > 0$, choose an edge u of $G \setminus G'$. Since G' satisfies (P3), by Lemma 2.1, there is a path l in G' connecting the two endpoints of u such that $I(l) = I(u)$.

We claim that l contains an edge v such that $I(v) = I(l)$. For otherwise, suppose that such a v does not exist. Consider the subgraph $H = \bigcup_{j \in \bar{I}} G_{I \cup \{j\}}$ and $H' = \bigcup_{j \in \bar{I}} G'_{I \cup \{j\}}$ where $I = I(u)$ and $\bar{I} = \{1, \dots, n\} \setminus I$. Since $G_{I \cup \{j\}}$ and $G'_{I \cup \{j\}}$ are connected graphs with the same vertex set, there is a one-to-one correspondence between the connected components of H and the connected components of H' such that corresponding components have the same vertex set.

Now observe that l is in H' . We have $H' = G'_I \cap [\bigcup_{j \in \bar{I}} G'_j]$. Since $l \subset G'_I$ and by assumption $I(v) \cap I \neq \emptyset$ for all edges v in l , it follows that l is in both terms of the intersection describing H' , and hence $l \subset H'$.

Thus, the two endpoints of u lie in the same connected component of H' , and hence in the same connected component of H . However, H is a subgraph of $G_I \setminus u$. Therefore, u is not a bridge in G_I , contradicting the assumption that G is a minimal (P3)-graph. Hence, such a v exists.

Now, define $G'' = (G' \setminus v) \cup u$. Then $\|G \setminus G''\| = \|G \setminus G'\| - 1$ and $\|G'\| = \|G''\|$. It is easy to verify that G'' is still a minimal (P3)-graph. By the inductive hypothesis, $\|G\| \leq \|G''\| = \|G'\|$. $\square$

Let $\bar{G}$ denote the complement of a graph G . Define $\mathcal{C} = \{\bar{G} : G \text{ satisfies (P3)}\}$. Clearly, $H_1 \subseteq H_2$ and $H_2 \in \mathcal{C}$ imply $H_1 \in \mathcal{C}$. Furthermore, by Lemma 2.3, we have the following theorem.

THEOREM 2.4. *$\mathcal{C}$ is the family of independent sets of a matroid. (See [5] for the definition of the matroid.)*

Since a maximum independent set in a matroid can be found by the greedy algorithm, Theorem 2.4 yields the following greedy algorithm for finding minimal (P3)-graphs.

ALGORITHM 1.

$G :=$ the complete graph of X ;

for each edge u of G do
 if $G \setminus u$ satisfies (P3) then $G := G \setminus u$.

Thus, we have Corollary 2.5.

COROLLARY 2.5. *Problem 3 can be solved in $O(n|X|^3 \log |X|)$ time.*

Proof. The validity of the algorithm follows from the matroid property.

Consider the time bound. It suffices to show that the time for each iteration of the loop is bounded by $O(n|X| \log |X|)$ since there are at most $|X|^2$ edges, and hence $|X|^2$ iterations.

We now analyze an iteration in which some edge $u = xy$ is removed from the current G , which is assumed to be (P3).

We claim that $G \setminus u$ is a (P3)-graph $\Leftrightarrow (G \setminus u)_I$ is connected, where $I = I(u)$. The $\Rightarrow$ direction is immediate by definition, so we prove the $\Leftarrow$ direction. By definition of (P3) it suffices to show that $(G \setminus u)_{I'}$ is connected for any index set I' . If $I' \subsetneq I$, then $u \notin G_{I'}$, so $(G \setminus u)_{I'} = G_{I'}$, and hence $(G \setminus u)_{I'}$ is connected. Suppose $I' \subseteq I$. Since by assumption $(G \setminus u)_I$ is connected, u is not a bridge of G_I , and since $G_I \subset G_{I'}$ it also follows that u is not a bridge of $G_{I'}$. Thus $(G \setminus u)_{I'}$ is connected.

Hence in each iteration we only need to check that $(G \setminus u)_I$ is connected for $I = I(u)$. First we must find the vertex set $V(G_I) = \bigcap_{i \in I} X_i$. We can do this by distinguishing some $i_0 \in I$, computing for each $x \in X_{i_0}$ which j satisfy $x \in X_j$, and then including x in G_I if and only if the j for which this holds are precisely the ones in I . This takes $\leq \log(|X_j|)$ time for a fixed j , and hence $\leq \sum_j \log |X_j| \leq n \log |X|$ time over all j . So the vertex set is found in time $O(n|X| \log |X|)$. Finally, to check that $(G \setminus u)_I$ is connected requires linear time (as a function of $|G_I| \leq O(|X|)$), so the total time per iteration is $O(n|X| \log |X|)$. $\square$

Unfortunately, so far we do not know any nontrivial upper bound for the ratio of the minimum (P3)-graph to the minimum (P2)-graph. This would be an interesting topic for future research.

Although such a nontrivial upper bound has not been found, we can give some reason for the minimum (P3)-graph to yield a good heuristic for the minimum (P2)-graph. First of all, Du and Chen [2] have given two operations for simplifying a (P2)-graph.

(i) If a (P2)-graph has a cycle Q containing an edge u such that $I(Q) = I(u)$, then u can be deleted.

(ii) Suppose a (P2)-graph has a cycle Q containing two vertices x and y such that each of the two paths from x to y in Q has an edge u satisfying $I(u) = I(I)$. Then the graph obtained by deleting both u 's and inserting the edge xy is still a (P2)-graph.

It is easy to see that these operations preserve the property (P3). Thus, a minimum (P3)-graph cannot be simplified by these operations.

In the next two lemmas we will show that in a special case, a minimum (P3)-graph is also a minimum (P2)-graph.

First, Du [3] has shown the following. (For the sake of completeness, we include the proof here.)

LEMMA 2.6. (1) *If a graph G satisfies*

(P4) *For any $i, j = 1, \dots, n$, the subgraph G_{ij} induced by $X_i \cap X_j$ is a tree, (note: $G_{ii} = G_i$) then G is a minimum (P2)-graph for $(X_1, \dots, X_n)$.*

(2) *If there is a (P4)-graph G for $(X_1, \dots, X_n)$ then every minimum (P2)-graph for $(X_1, \dots, X_n)$ satisfies (P4).*

Proof. (1) We prove it by induction on n . It is trivial for $n = 1$. Assume $n > 1$. By the induction hypothesis, $\bigcup_{i=1}^{n-1} G_i$ is a minimum (P2)-graph for $(X_1, \dots, X_{n-1})$. Let G' be any (P2)-graph for $(X_1, \dots, X_n)$. Then

$$\left\| \bigcup_{i=1}^{n-1} G_i \right\| \geq \left\| \bigcup_{i=1}^{n-1} G_i \right\|.$$

Let k be the number of connected components of X_n obtained by joining all pairs of vertices in $X_i \cap X_n$ for each $i = 1, \dots, n-1$. Then $\|G' \setminus \bigcup_{i=1}^{n-1} G_i\| \geq k-1$. However, the property (P4) implies that $\|G \setminus \bigcup_{i=1}^{n-1} G_i\| = k-1$. Therefore

$$\|G'\| = \left\| G' \setminus \bigcup_{i=1}^{n-1} G_i \right\| + \left\| \bigcup_{i=1}^{n-1} G_i \right\| \geq \left\| G \setminus \bigcup_{i=1}^{n-1} G_i \right\| + \left\| \bigcup_{i=1}^{n-1} G_i \right\| = \|G\|.$$

(2) We will use the fact that for $n \leq 2$ a (P2)-graph G' is minimum if and only if G' satisfies (P4). Suppose then that a (P2)-graph G' does not satisfy (P4). By rearranging indices of the X_i 's, we can assume that either G'_1 or G'_{12} is not a tree. Letting G satisfy (P4) (as in (1)), the case $n \leq 2$ gives $\|G'_1 \cup G'_2\| > \|G_1 \cup G_2\|$. From the proof of (1), we have $\|\bigcup_{j=1}^i G'_j \cup \bigcup_{j=1}^{i-1} G'_j\| \geq \|\bigcup_{j=1}^i G_j \cup \bigcup_{j=1}^{i-1} G_j\|$. Hence

$$\begin{aligned} \|G'\| &= \sum_{i=3}^n \left\| \bigcup_{j=1}^i G'_j \setminus \bigcup_{j=1}^{i-1} G'_j \right\| + \|G'_1 \cup G'_2\| \\ &> \sum_{i=3}^n \left\| \bigcup_{j=1}^i G_j \setminus \bigcup_{j=1}^{i-1} G_j \right\| + \|G_1 \cup G_2\| = \|G\|. \end{aligned}$$

But since G satisfies (P4) we know by (1) that G also satisfies (P2). Hence G' is not a minimum (P2)-graph, a contradiction. $\square$

We have the following similar result.

LEMMA 2.7. (1) A (P4)-graph is a minimum (P3)-graph.

(2) If there is a (P4)-graph for $(X_1, \dots, X_n)$ then every minimum (P3)-graph for $(X_1, \dots, X_n)$ satisfies (P4).

Proof. We first prove that (P4) implies (P3). Let G be a (P4)-graph for $(X_1, \dots, X_n)$. Consider any two vertices x and y of G . For any $i \in I(x) \cap I(y)$, since $G_i = G_{ii}$ is a tree, there is a unique path l_i from x to y in G_i . Similarly, for any $i, j \in I(x) \cap I(y)$, since G_{ij} is a tree, there is a path l_{ij} from x to y in G_{ij} . By the uniqueness of l_i and l_j , we have $l_i = l_{ij} = l_j$. From this fact, we can see that all l_i for $i \in I(x) \cap I(y)$ are identical, say to l . Hence we have $I(l) = I(x) \cap I(y)$. Therefore by Lemma 2.1, G satisfies (P3).

Consider (1). Clearly every (P3)-graph is a (P2)-graph. Hence, if a (P3)-graph is a minimum (P2)-graph then it is also a minimum (P3)-graph. Now, a (P4)-graph satisfies (P3) by the paragraph above and is also a minimum (P2)-graph by Lemma 2.6. Hence it is a minimum (P3)-graph.

To see (2), we only need to notice that the existence of a (P4)-graph implies the existence of a minimum (P3)-graph which is also a minimum (P2)-graph. But a minimum (P2)-graph is a (P4)-graph by Lemma 2.6 and the existence of a (P4)-graph. $\square$

By Lemmas 2.7 and 2.6, we now have the following corollary of Theorem 2.4.

COROLLARY 2.8. The question of whether a (P4)-graph for $(X_1, \dots, X_n)$ exists can be answered in $O(n |X|^3 \log |X|)$ time, and if a (P4)-graph for $(X_1, \dots, X_n)$ exists then a minimum (P2)-graph can be found in $O(n |X|^3 \log |X|)$ time.

An idea for constructing a heuristic can be motivated from the above results. Start by choosing a largest subset I_1 of $\{1, \dots, n\}$ such that there exists a (P4)-graph for

$(X_i, i \in I_1)$. In each subsequent step k , choose a largest subset I_k of $\{1, \dots, n\} \setminus (\cup_{i=1}^{k-1} I_i)$ such that there exists a (P4)-graph for I_k . For each I_k , construct a minimum (P2)-graph for $(X_i, i \in I_k)$ and then join them. As will be seen below, this idea results in a heuristic for Problem 2 within a factor of $\sqrt{2n}$ from optimal.

A simple sufficient condition for an input $(X_1, \dots, X_n)$ to have a (P4)-graph is that all $X_i, i = 1, \dots, n$, have a point x in common. Then the tree consisting of an internal vertex, which is x , and $|X| - 1$ leaves, which form the set $X \setminus \{x\}$, where $X = \cup_{i=1}^n X_i$, is a (P4)-graph, and hence a minimum (P2)-graph. Now, for an input $(X_1, \dots, X_n)$, we choose $I_1, \dots, I_k$ in the following way.

ALGORITHM 2.

Choose x_1 such that $|I(x_1)| = \max_{x \in X} |I(x)|$; $I_1 := I(x_1)$; $I := I(x_1)$; $k := 2$;
 while $\{1, \dots, n\} \setminus I \neq \emptyset$ do begin
 choose x_k such that
 $|I(x_k) \setminus I| = \max_{x \in X} |I(x) \setminus I|$;
 $I_k := I(x_k) \setminus \cup_{j=1}^{k-1} I_j$;
 $I := I \cup I_k$;
 $k := k + 1$;
 end.

From the algorithm, we can see that the $I_i, i = 1, \dots, k$, form a partition of $\{1, \dots, n\}$, and for each $i = 1, \dots, k$ and all $j \in I_i$ the sets X_j have the point x_i in common. In addition, $|I_i| = \max_{x \in X} |I(x) \setminus \cup_{j=1}^{i-1} I_j|$. Let H be the union over all $i = 1, \dots, k$ of minimum (P2)-graphs for $(X_j, j \in I_i)$. Clearly,

$$\|H\| \leq \sum_{i=1}^k \left(\left| \bigcup_{j \in I_i} X_j \right| - 1 \right).$$

The next lemma states that $\|H\|$ is within a factor of $\sqrt{2n}$ from an optimal solution to Problem 2.

LEMMA 2.9. *Let G^* be a minimum (P2)-graph for $(X_1, \dots, X_n)$. Then*

$$\sum_{i=1}^k \left(\left| \bigcup_{j \in I_i} X_j \right| - 1 \right) < \sqrt{2n} \|G^*\|.$$

Proof. Let $G^{(i)} = \cup_{j \in I_i} G_j^*$ and $V^{(i)} = \cup_{j \in I_i} X_j$. Since $X_j, j \in I_i$, have a point in common, $G^{(i)}$ is a connected graph with the vertex set $V^{(i)}$. Therefore,

$$|V^{(i)}| - 1 \leq \|G^{(i)}\|.$$

Let t be the maximum number of $V^{(i)}$'s that can have a point in common. We first prove $t < \sqrt{2n}$. Let x be a common point of $V^{(i_1)}, V^{(i_2)}, \dots, V^{(i_t)}$ with $i_1 < i_2 < \dots < i_t$. Since all $I_i, i = 1, \dots, t$, are disjoint, $I(x)$ contains at least t indices not in $\cup_{j=1}^{i_1-1} I_j$. Hence, $|I_{i_1}| = \max_{x \in X} |I(x) \setminus \cup_{j=1}^{i_1-1} I_j| \geq t$. Similarly, we can prove that $|I_{i_2}| \geq t - 1, \dots, |I_{i_t}| \geq 1$. Therefore, $n \geq |I_{i_1}| + |I_{i_2}| + \dots + |I_{i_t}| \geq t + (t - 1) + \dots + 1 = \frac{1}{2}t(t + 1) > \frac{1}{2}t^2$. Hence, $t < \sqrt{2n}$.

Now, since every point x can be contained by at most t of the $V^{(i)}$'s, every edge of G^* can belong to at most t of the $G^{(i)}$'s. Thus, we have

$$\sum_{i=1}^k (|V^{(i)}| - 1) \leq \sum_{i=1}^k \|G^{(i)}\| \leq t \|G^*\| < \sqrt{2n} \|G^*\|. \quad \square$$

THEOREM 2.10. *There is a heuristic for Problem 2 that runs in $O(n|X|^2)$ time and is within a factor $\sqrt{2n}$ from optimal.*

It is worth mentioning that so far we do not know how to find a largest subset I of $\{1, \dots, n\}$ such that a (P4)-graph for $(X_i, i \in I)$ exists. A better heuristic for Problem 2 may be found by solving this problem.

3. On a bounded heuristic for Problem 2. We have presented a heuristic for Problem 2 within a factor $\sqrt{2n}$ from optimal. Is there a heuristic within a constant factor from optimal? We know of such a heuristic only for the case that for every $i = 1, \dots, n$, $|X_i| \leq 3$. There is an interesting relation between this case and the vertex covering problem (denoted VC). A vertex cover of a graph G is a subset S of $V(G)$ such that every edge in G is incident on at least one point of S . The problem VC is as follows. Given a graph G and positive integer k , does G have a vertex cover of size at most k ?

THEOREM 3.1. *Problem 2 is NP-complete, even when all subsets satisfy $|X_i| \leq 3$.*

Proof. We reduce the vertex covering problem (VC) to Problem 2. Let G and $k \in \mathbb{Z}^+$ be an instance of VC. Let x^* be a new point not in $V(G)$, and for any edge $u = xy \in E(G)$ let $X_u = \{x, y\}$ and $X_u^* = \{x, y, x^*\}$. The corresponding instance of Problem 2 is the set system $(X_u, X_u^*; u \in E(G))$ and integer $\|G\| + k$ (where the question is whether the system has a (P2)-graph with at most $\|G\| + k$ edges).

It is easy to verify that a subset Y of V is a vertex-covering of G if and only if the graph $G \cup (\cup_{x \in Y} (x^*, x))$ is a (P2)-graph for $(X_u, X_u^*, u \in E)$. $\square$

THEOREM 3.2. *There is a heuristic within a factor of 2 from optimal for Problem 2 with the restriction that all X_i 's are of size at most 3.*

Proof. We provide a polynomial time reduction preserving the approximation performance from this special case of Problem 2 to VC. Since there is a polynomial approximation for VC to within a factor of 2 [7], the theorem follows.

For our instance of Problem 2 let $X_1, X_2, \dots, X_n$ be sets of points of size at most 3. Without loss of generality, assume $|X_1| = \dots = |X_m| = 3$ and $|X_{m+1}| = \dots = |X_n| = 2$. The graph G for the corresponding instance of VC is constructed as follows.

(1) The vertices of G are all subsets of size 2 of $X = \cup_{i=1}^n X_i$.

(2) Two vertices A, B of G are joined by an edge if and only if neither A nor B is in $\{X_{m+1}, \dots, X_n\}$ and there is an X_i , $1 \leq i \leq m$, such that $A \subseteq X_i$ and $B \subseteq X_i$.

Let S be a subset of vertices of G disjoint from $\{X_{m+1}, \dots, X_n\}$. Define H to be the graph with the vertex set X and the edge set $\{X_{m+1}, \dots, X_n\} \cup \{(x, y): \{x, y\} \in S\}$. We will prove that S is a vertex-covering of G if and only if H is a (P2)-graph for $(X_1, \dots, X_n)$.

First, suppose that S is a vertex-covering. To see that H satisfies (P2), it suffices to show that every X_i of size 3 contains at least two subsets belonging to

$$S \cup \{X_{m+1}, \dots, X_n\}.$$

Let A, B , and C be the three subsets of size two of X_i . If two of A, B , and C belong to $\{X_{m+1}, \dots, X_n\}$ then we are done. If one of A, B , and C , say A , belongs to $\{X_{m+1}, \dots, X_n\}$, then the edge BC exists in G , so either B or C is in S since S is a vertex-covering. Hence X_i contains the two required subsets A , and B or C . If none of A, B , and C belongs to $\{X_{m+1}, \dots, X_n\}$, then the edges AB, BC , and CA all exist in G . To cover them, S must contain at least two of A, B , and C .

Next, suppose that H is a (P2)-graph for $(X_1, \dots, X_n)$. Consider each edge AB of G . There exists X_i , $1 \leq i \leq m$, such that $A \subseteq X_i$ and $B \subseteq X_i$. Since H_i is connected and X_i is of size three, H_i has at least two edges of which at least one corresponds to A or B . Hence, either A or B belongs to $S \cup \{X_{m+1}, \dots, X_n\}$. However, neither A nor B is in

$\{X_{m+1}, \dots, X_n\}$. Therefore, either A or B is in S . It follows that S is a vertex-covering of G .

Note that $\|H\| = n - m + |S|$. Hence, S is a minimum vertex-covering of G if and only if H is a minimum (P2)-graph for $(X_1, \dots, X_n)$ and if S is a heuristic within two from optimal for VC then so is H for the minimum (P2)-graph. This completes our reduction and the theorem is proved. $\square$

4. Remarks on Problem 1. As we have indicated in the Introduction, Problem 1 seems much more difficult than Problem 2. A result similar to Lemma 2.6 has been given by Chao and Du [1]. (See Theorem 4.1 below.) However, the proof is different and more difficult. Surprisingly, it seems the simpler method in the proof of Lemma 2.6 does not work here.

THEOREM 4.1 (Chao and Du [1]). *If G is a (P4)-graph for $(X \setminus Y_1, \dots, X \setminus Y_n)$ and for every edge u of G , there exists an index i such that two endpoints of u are in X_i , then G is a minimum (P1)-graph for $(X, X_1, Y_1, \dots, X_n, Y_n)$.*

Another example that can explain the difference between these two problems is the special case where all $X \setminus Y_i$ have a point in common. For Problem 2, this is equivalent to saying that all X_i have a point in common, so an optimal solution can be found easily as shown in § 2. However, for Problem 1, it is still an unsolved case. Let x^* be the common point of all $X \setminus Y_i$, and let G be the graph formed by unioning all edges xx^* for $x \in \bigcup_{i=1}^n X_i$. The trouble is that G may not be a minimum (P1)-graph for $(X, X_1, Y_1, \dots, X_n, Y_n)$. In fact, let G^* be a minimum (P1)-graph, then $\|G\| - \|G^*\|$ can be made to be any positive integer by choosing the input properly. However, G is a bounded heuristic for G^* .

LEMMA 4.2. $\|G\| \leq 2\|G^*\|$.

Proof. Let G^* be a minimum (P1)-graph such that the degree of x^* in G^* is a maximum over all minimum (P1)-graphs. Let H and K be connected components of G^* such that x^* is in H and is not in K . It is easy to see that H is a tree with only one internal vertex that is x^* . Since every X_i is of size at least two, K contains at least one edge. Deleting all edges in K and connecting x^* to every vertex in K increases the total number of edges by one per edge in K . Therefore, $\|G\| \leq 2\|G^*\|$. $\square$

We have Theorem 4.3, which is similar to Theorem 2.10.

THEOREM 4.3. *There is a heuristic for Problem 1 that runs in*

$$O(\min\{n|X|^2, n^2|X|\})$$

time and is within the factor $2\sqrt{2n}$ from optimal.

Proof. Define $I(x) = \{i: 1 \leq i \leq n, x \in X \setminus Y_i\}$ for any $x \in X$ and $I(H) = \bigcap_{x \in V} I(x)$ for a graph H with the vertex set $V \subseteq X$. Choose the index sets $I_1, \dots, I_k$ by Algorithm 2. Let G^* be a minimum (P1)-graph. Let $G^{(i)} = \bigcup_{j \in I_i} G_j^*$ where G_j^* is the subgraph of G^* induced by $X \setminus Y_j$. By the same argument as in the proof of Lemma 2.9, we can obtain $\sum_{i=1}^k \|G^{(i)}\| < \sqrt{2n}\|G^*\|$. By Lemma 4.2, for each I_i , we can construct a (P1)-graph $H^{(i)}$ for $(X, X_j, Y_j, j \in I_i)$ such that $\|H^{(i)}\| \leq 2\|G^{(i)}\|$. Therefore, the heuristic solution $H = \bigcup_{i=1}^k H^{(i)}$ satisfies

$$\|H\| \leq \sum_{i=1}^k \|H^{(i)}\| \leq 2 \sum_{i=1}^k \|G^{(i)}\| < 2\sqrt{2n}\|G^*\| \quad \square$$

Acknowledgments. Some ideas included in this paper were obtained in a class offered by Professor Ronald V. Book at the University of California, Santa Barbara, CA during 1984–1985. Thus the authors thank Professor Book and several classmates for helpful discussions.

REFERENCES

- [1] S.-C. CHAO AND D.-Z. DU, *A sufficient condition for optimal solutions of the valve-placement problem*, J. North-East Heavy Industry Institute, 4 (1983). (In Chinese.)
- [2] D.-Z. DU AND Y.-M. CHEN, *Placement of valves in vacuum systems*, J. Light Sources, 4 (1976). (In Chinese.)
- [3] D.-Z. DU, *An optimization problem on graphs*, Discrete Appl. Math., 14 (1986), pp. 101–104.
- [4] ———, *Generalized complexity cores and levelability of intractable sets*, Ph.D. thesis, University of California, Santa Barbara, CA, 1985.
- [5] M. GONDRAAN AND M. MINOUX, *Graphs and Algorithms*, John Wiley, New York, 1984.
- [6] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability, A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, CA, 1979.
- [7] C. H. PAPADIMITRIOU AND K. STEIGLITZ, *Combinatorial Optimization, Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, NJ, 1982.

LAGUERRE POLYNOMIALS, WEIGHTED DERANGEMENTS, AND POSITIVITY*

DOMINIQUE FOATA† AND DORON ZEILBERGER‡

Abstract. A calculation of the linearization coefficients of the (generalized) Laguerre polynomials $L_n^{(\alpha)}(x)$ is proposed by means of analytic and combinatorial methods. This paper extends to the case of an arbitrary α a combinatoric and analytic result due to Askey, Ismail, and Koornwinder and Even and Gillis.

Key words. Laguerre polynomials, linearization coefficients, weighted derangements, MacMahon Master Theorem (β -extension)

AMS(MOS) subject classifications. 33A65, 05A15

1. Introduction. Let $(p_n(x))$ be a sequence of polynomials, orthogonal with respect to a weight function w . One of the aspects of the linearization of the product of the $p_n(x)$'s is the evaluation of integrals of the form

$$\mathcal{J} = \int \prod_{i=1}^m n_i dw,$$

for the classical polynomials, such as the Jacobi, Meixner, Charlier, Laguerre, and Hermite polynomials. What is meant by evaluation is either the determination of a formula for $\mathcal{J}$ in terms of the classical hypergeometric series (see, e.g., the fantastic formula found by Rahman [Ra] for the Jacobi polynomials involving the series ${}_9F_8$), or the *geometric* interpretation of $\mathcal{J}$ as a generating polynomial for some combinatorial objects, such as permutations or partitions (see, e.g., the present paper for the Laguerre polynomials, or the article by Zeng [Ze] for the Meixner, Krawtchouk, and Charlier polynomials). Of course, for many problems it is essential that $\mathcal{J}$ be positive. In Rahman's formula, for instance, all the hypergeometric series involved in the formula are positive—this is the easy part—the difficult part is the derivation of the formula itself. In the combinatorial approach the positivity of $\mathcal{J}$ also appears as a byproduct, the essential part being played by the construction of the geometric setup for the integral $\mathcal{J}$.

We want to illustrate this in this paper by making a systematic study of integrals of products of (general) *Laguerre polynomials* $L_n^{(\alpha)}(x)$. Recall that those polynomials are orthogonal with respect to the weight function $x^\alpha e^{-x}$ over $\mathbb{R}_+$. They may also be defined by their generating function

$$(1.1) \quad \sum_{n=0}^{\infty} L_n^{(\alpha)}(x) u^n = (1-u)^{-\alpha-1} \exp \frac{-xu}{1-u}.$$

Let m be a positive integer, $\mathbf{n}$ be a sequence $\mathbf{n} = (n_1, \dots, n_m)$ of nonnegative integers, Λ be a sequence $\Lambda = (\lambda_1, \dots, \lambda_m)$ of real numbers and let $A(\mathbf{n}; \alpha)$, $B(\mathbf{n}; \Lambda; \alpha)$ be the

* Received by the editors January 14, 1988; accepted for publication May 5, 1988.

† Département de mathématique, Université Louis-Pasteur, 7, rue René-Descartes, F-67084 Strasbourg, France. The work of this author was supported by the French Coordinated Research Program in Mathematics and Computer Science.

‡ Department of Mathematics, Drexel University, Philadelphia, Pennsylvania 19104. The work of this author was partially supported by the National Science Foundation.

integrals

$$(1.2) \quad A(\mathbf{n}; \alpha) = (-1)^{n_1 + \cdots + n_m} \int_0^\infty \left(\prod_{i=1}^m L_{n_i}^{(\alpha)}(x) \right) x^\alpha e^{-x} dx;$$

$$(1.3) \quad B(\mathbf{n}; \Lambda; \alpha) = \int_0^\infty \left(\prod_{i=1}^m L_{n_i}^{(\alpha)}(\lambda_i x) \right) x^\alpha e^{-x} dx.$$

Also introduce the expressions:

$$(1.4) \quad \mathcal{J}(\mathbf{n}; \alpha) = \frac{1}{\Gamma(\alpha + 1)} \left(\prod_{i=1}^m n_i! \right) A(\mathbf{n}; \alpha)$$

and

$$(1.5) \quad J(\mathbf{n}; \Lambda; \alpha) = \frac{1}{\Gamma(\alpha + 1)} \left(\prod_{i=1}^m n_i! \right) B(\mathbf{n}; \Lambda; \alpha).$$

The first goal of the present paper is to give a *combinatorial interpretation* to $\mathcal{J}(\mathbf{n}; \alpha)$ and $J(\mathbf{n}; \Lambda; \alpha)$, that is, to show that those two expressions are generating polynomials for permutations by certain statistics. In particular, $\mathcal{J}(\mathbf{n}; \alpha)$ will be shown to be the generating function for a special class of permutations, called *m-derangements* by the number of cycles.

To make it more precise let us now define those combinatorial objects. Let $A_1 = \{a_{1,1}, \dots, a_{1,n_1}\}, \dots, A_m = \{a_{m,1}, \dots, a_{m,n_m}\}$ be m mutually disjoint finite sets. The group of the permutations of $A = A_1 + \cdots + A_m$ is denoted by $\mathcal{P}(\mathbf{n})$. Consider any permutation belonging to $\mathcal{P}(\mathbf{n})$. An element $a_{i,j}$ of A_i is said to be (π) -incestuous, if it is sent by π to one of its own kind, i.e., if $\pi(a_{i,j}) \in A_i$. Denote by $\text{Inc}_i \pi$ the set of the incestuous elements of π in A_i and let $\text{Inc } \pi = \sum_i \text{Inc}_i \pi$. If a permutation π has no incestuous element, (i.e., if $\pi(A_i) \cap A_i = \emptyset$ for all i), it is called an *m-derangement*. Denote by $\mathcal{D}(\mathbf{n})$ the subset of $\mathcal{P}(\mathbf{n})$ consisting of all the *m-derangements*. Finally, for each permutation π in $\mathcal{P}(\mathbf{n})$ denote by $\text{cyc } \pi$ its *number of cycles* and define its (w) -weight by

$$w(\pi) = (\alpha + 1)^{\text{cyc } \pi}.$$

The first result is the following theorem.

THEOREM 1. *For each variable α we have*

$$(1.6) \quad \mathcal{J}(\mathbf{n}; \alpha) = \sum_{\pi} w(\pi) = \sum_{\pi} (\alpha + 1)^{\text{cyc } \pi} \quad (\pi \in \mathcal{D}(\mathbf{n})).$$

As for the combinatorial interpretation of $J(\mathbf{n}, \Lambda; \alpha)$, introduce another (W) -weight as follows:

$$W(\pi) = w(\pi) \prod_{i=1}^m (1 - \lambda_i)^{|\text{Inc}_i \pi|} (-\lambda_i)^{|A_i \setminus \text{Inc}_i \pi|}.$$

The corresponding result for $J(\mathbf{n}; \Lambda; \alpha)$ is then as follows in Theorem 2.

THEOREM 2. *For each variable α we have*

$$(1.7) \quad J(\mathbf{n}; \Lambda; \alpha) = \sum_{\pi} W(\pi) \quad (\pi \in \mathcal{P}(\mathbf{n})).$$

When $\Lambda = \mathbf{1} = (1, 1, \dots, 1)$, the W -weight reduces to

$$W(\pi) = \begin{cases} (-1)^{n_1 + \cdots + n_m} w(\pi) & \text{if } \pi \text{ is an } m\text{-derangement,} \\ 0 & \text{otherwise.} \end{cases}$$

Thus, Theorem 1 is just a particular case of Theorem 2. Accordingly, only Theorem 2 will be proved in this paper.

The second goal of this paper is to study the *positivity* of $\mathcal{J}$ and J . For $\mathcal{J}$ it is easy; the positivity property of $\mathcal{J}(\mathbf{n}; \alpha)$ for $\alpha > -1$ follows immediately from our combinatorial interpretation (1.6). As such, it is in no case a new result. It follows from more general theorems that cover the case of several classical polynomials, including the Laguerre polynomials (see, e.g., the basic paper by Askey [As1], or his monograph [As2]). The positivity property can also be proved by means of a simple analytic argument, as derived by Askey and [As-Is], [As-Is-Ko], [As-Ga]. As they have noted, the generating function (1.1) yields the following identity:

$$(1.8) \quad \sum_{\mathbf{n}} A(\mathbf{n}; \alpha) x_1^{n_1} \cdots x_m^{n_m} = \frac{\Gamma(\alpha + 1)}{(1 - e_2 - 2e_3 - \cdots - (m-1)e_m)^{\alpha+1}},$$

where $\mathbf{n} = (n_1, \dots, n_m)$ runs over all sequences of m nonnegative integers and where e_j denotes the j th elementary symmetric function in $x_1, \dots, x_m$. The positivity of $\mathcal{J}(\mathbf{n}; \alpha)$ for $\alpha > -1$ is then clear from (1.8).

As far as positivity is concerned, the combinatorial approach derived in this paper refines the analytic result in the following sense. Not only is $\mathcal{J}(\mathbf{n}; \alpha)$ shown to be positive, but it is, in fact, a *polynomial* in $(\alpha + 1)$ with positive *integral* coefficients.

When $\alpha = 0$, the integrand in $\mathcal{J}(\mathbf{n}; \alpha)$ is a product of *simple* Laguerre polynomials and Theorem 1 implies that $\mathcal{J}(\mathbf{n}; 0)$ is equal to the *number* of m -derangements, a result due to Even and Gillis [Ev-Gi]. Other proofs can be found in [Ja] and [Sa-Vi]. In [As-Is, pp. 857–858] the authors were very close to finding a combinatorial interpretation of $\mathcal{J}(\mathbf{n}; \alpha)$ for an arbitrary α . What was missing in their derivation was an appropriate extension of the “Master Theorem” of MacMahon [Mac, pp. 97–98]. It is also the purpose of this paper to state and prove such a theorem (see the β -extension of the Master Theorem in § 3). As shown in § 5, that β -extension, together with the calculations made by Askey and his coauthors, suffice to establish Theorem 2. We also give a truly combinatorial proof in § 4, after having recorded the material on injection counting in § 2.

The positivity of $J(\mathbf{n}; \Lambda; \alpha)$ for other values of Λ is more difficult to handle. The combinatorial interpretation (1.7) brings no evidence of the positivity for an arbitrary $\alpha \geq 0$. Koornwinder’s inequality [Ko] says that when $m = 3$, $\Lambda = (\lambda, 1 - \lambda, 1)$, then

$$(1.9) \quad \alpha \geq 0, \quad 0 \leq \lambda \leq 1 \Rightarrow J(\mathbf{n}, \Lambda, \alpha) \geq 0.$$

Also the strict inequality holds when $n_1 = 0$ and $0 < \lambda < 1$. Inequality (1.9) is certainly a deep result. To derive further extensions for $m > 3$, Askey, Ismail, and Koornwinder [As-Is-Ko] have used the orthogonality property of the Laguerre polynomials and also the so-called “old expansion” of the same polynomials. As it is (too) easy to prove both the orthogonality relation and the old expansion by combinatorial methods, we shall not concern ourselves with the general extension. We shall concentrate on re-proving Koornwinder’s inequality (Theorem 3 of § 6). The argument developed is very similar to the one developed by Ismail and Tamhankar [Is-Ta] or Gillis and Zeilberger [Gi-Ze]. We may say that the proof of Theorem 3 is the rewriting of the latter authors’ paper using the spirit and method of the former ones.

2. Cycles. We will need three results that are fundamental in the current combinatorial interpretation of special functions. First, the generating function for the set $\mathcal{S}_n$ of all the permutations on n elements by number of cycles is given by (see, e.g., [Ri, p. 78]):

$$(2.1) \quad w(\mathcal{S}_n) = \sum_{\pi \in \mathcal{S}_n} w(\pi) = (\alpha + 1)_n = (\alpha + 1)(\alpha + 2) \cdots (\alpha + n).$$

Let $1 \leq k \leq n$ and S be a $(n - k)$ -element subset of the n -element

$$[n] = \{1; 2; \dots, n\}.$$

The set of *injections* from S into $[n]$ will be denoted by $\text{Inj}(S, n)$. An injection from S to $[n]$ consists of a (possibly empty) collection of cycles within S and some simple paths that wander in S , but terminate at a point outside S . Similarly, denote by $\text{cyc } \pi$ the number of cycles of π and define its *weight* by $w(\pi) = (\alpha + 1)^{\text{cyc } \pi}$.

For example, if $S = \{1, 2, 3, 4, 5, 6\}$ and $n = 9$, then $(1, 3), (2), 4 \rightarrow 5 \rightarrow 7, 6 \rightarrow 8$ is an injection with weight $(\alpha + 1)^2$.

The result analogous to (2.1) reads (see [Fo-St, Lemma 2.1]) as follows: if $\text{card } S = n - k$, then

$$(2.2) \quad w(\text{Inj}(S, n)) = \sum_{\pi} w(\pi) = (\alpha + 1 + k)_{n-k} \quad (\pi \in \text{Inj}(S, n)).$$

The third result is concerned with the calculation of the generating function for a particular class of permutations of the set $A = A_1 + \dots + A_m$ (see the notation introduced in § 1). For each $i = 1, \dots, m$ let T_i be a given subset of A_i of cardinality $(n_i - k_i)$ and denote by $\mathcal{P}(T \subset \text{Inc})$ the set of all permutations π of A satisfying $T_i \subset \text{Inc}_i \pi$ for all i .

LEMMA. *We have*

$$(2.3) \quad w(\mathcal{P}(T \subset \text{Inc})) = (\alpha + 1)_{k_1 + \dots + k_m} \prod_{i=1}^m (\alpha + 1 + k_i)_{n_i - k_i}.$$

Proof. From (2.1) and (2.2) it follows that the right-hand side of (2.3) is the generating function for the product $\mathcal{P}(\mathbf{k}) \times \prod_{i=1}^m \text{Inj}(T_i, n_i)$ by w . To prove the lemma it then suffices to construct a w -weight preserving bijection $\pi \rightarrow (\pi_1, \dots, \pi_m, \sigma)$ of $\mathcal{P}(T \subset \text{Inc})$ onto that product. Write π in cycle form. Then in each cycle of π delete all the elements of $T = T_1 \cup \dots \cup T_m$. What remains is a permutation written in cycle form. Call it σ .

To get π_i take all the cycles of π consisting *only* of elements of T_i . Also take the connected portions of T_i lying in other cycles. Doing this will result in a certain number of *paths* that wander through T_i but terminate in an element not in T_i .

Clearly, σ belongs to $\mathcal{P}(\mathbf{k})$ and each π_i is an injection of S_i into A_i . Moreover, the total number of cycles of $\sigma, \pi_1, \dots, \pi_m$ is equal to $\text{cyc } \pi$. Thus, the mapping is w -preserving. The reverse construction is immediate. $\square$

Example. Take the following:

$$\begin{aligned} n_1 &= 6, & n_2 &= 6, & n_3 &= 6, \\ k_1 &= 3, & k_2 &= 3, & k_3 &= 3, \\ T_1 &= \{a_1, a_2, a_3\}, & T_2 &= \{b_1, b_2, b_3\}, & T_3 &= \{c_1, c_2, c_3\}, \\ \pi &= (a_1 a_2)(a_4 b_1 b_5 a_5 a_3)(b_2 b_4 c_1 c_4 c_3 c_2 c_6)(c_5 a_6)(b_3)(b_6). \end{aligned}$$

Then

$$\begin{aligned} \sigma &= (a_4 b_5 a_5)(b_4 c_4 c_6)(c_5 c_6)(b_6), \\ \pi_1 &= (a_1 a_2), & a_3 &\rightarrow a_4, \\ \pi_2 &= (b_3), & b_1 &\rightarrow b_5, & b_2 &\rightarrow b_4, \\ \pi_3 &= c_1 \rightarrow c_4, & c_3 &\rightarrow c_2 \rightarrow c_6. \end{aligned}$$

3. The β -extension of the MacMahon Master Theorem. Let V_m be the determinant $\det(\delta_{ij} - b(i, j)x_j)(1 \leq i, j \leq m)$. The MacMahon Master Theorem asserts that the coefficient of $x_1^{n_1} \cdots x_m^{n_m}$ in the expansion of V_m^{-1} is equal to the coefficient of $x_1^{n_1} \cdots x_m^{n_m}$ in the product

$$(3.1) \quad (b(1, 1)x_1 + \cdots + b(1, m)x_m)^{n_1} \cdots (b(m, 1)x_1 + \cdots + b(m, m)x_m)^{n_m}.$$

It will be convenient to restate this statement in a slightly different form. Let $\mathcal{R}(\mathbf{n})$ denote the set of all the rearrangements

$$r = r(1, 1) \cdots r(1, n_1) \cdots r(m, 1) \cdots r(m, n_m)$$

of the word $1^{n_1} \cdots m^{n_m}$ and let

$$v(r) = \prod_{i,j} b(i, r(i, j)) \quad (1 \leq i \leq m; 1 \leq j \leq n_i).$$

Clearly, the coefficient of $x_1^{n_1} \cdots x_m^{n_m}$ in (3.1) is equal to the sum of all the $v(r)$ with r running over all the rearrangements of $1^{n_1} \cdots m^{n_m}$.

Next, consider a permutation π belonging to $\mathcal{P}(\mathbf{n})$ (defined in § 1). If π sends (i, j) over (i', j') , write $i' = c\pi(i, j)$. Furthermore, define

$$v(\pi) = \prod_{i,j} b(i, c\pi(i, j)) \quad (1 \leq i \leq m; 1 \leq j \leq n_i).$$

For instance, if

$$\pi = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{2,1} & a_{2,2} & a_{2,3} & a_{3,1} \\ a_{2,2} & a_{3,1} & a_{2,3} & a_{2,1} & a_{1,2} & a_{1,2} \end{pmatrix}$$

belongs to $\mathcal{P}(2, 3, 1)$, then, $v(\pi) = b(1, 2)b(1, 3)b(2, 2)b(2, 2)b(2, 1)b(3, 1)$.

To each rearrangement r in $\mathcal{R}(\mathbf{n})$ there correspond exactly $n_1! \cdots n_m!$ permutations π in $\mathcal{P}(\mathbf{n})$ with the property that $v(\pi) = v(r)$. Therefore, the coefficient of $x_1^{n_1} \cdots x_m^{n_m}$ in (3.1) is also equal to

$$(3.2) \quad \frac{1}{n_1! \cdots n_m!} \sum_{\pi \in \mathcal{P}(\mathbf{n})} v(\pi) \left(= \frac{1}{n_1! \cdots n_m!} v(\mathcal{P}(\mathbf{n})) \right).$$

The MacMahon Master identity can then be restated as

$$(3.3) \quad \sum \frac{x_1^{n_1}}{n_1!} \cdots \frac{x_m^{n_m}}{n_m!} v(\mathcal{P}(\mathbf{n})) = V_m^{-1}.$$

Next define the β -weight $v(\beta; \pi)$ of each permutation π in $\mathcal{P}(\mathbf{n})$ by

$$(3.4) \quad v(\beta; \pi) = \beta^{\text{cyc } \pi} v(\pi).$$

THEOREM (β -extension of the MacMahon Master Theorem). *The following identity holds:*

$$(3.5) \quad \sum \frac{x_1^{n_1}}{n_1!} \cdots \frac{x_m^{n_m}}{n_m!} v(\beta; \mathcal{P}(\mathbf{n})) = V_m^{-\beta}.$$

Proof. Consider the partitional complex (see, e.g., [Fo], [Fo-Sch]) of the permutations and denote by $\mathcal{C}\mathcal{P}(\mathbf{n})$ the subset of the *connected* permutations in $\mathcal{P}(\mathbf{n})$. As the weight $v(\beta; \cdot)$ is *multiplicative* [Fo], [Fo-Sch], the following identity holds:

$$(3.6) \quad \sum \frac{x_1^{n_1}}{n_1!} \cdots \frac{x_m^{n_m}}{n_m!} v(\beta; \mathcal{P}(\mathbf{n})) = \exp \sum \frac{x_1^{n_1}}{n_1!} \cdots \frac{x_m^{n_m}}{n_m!} v(\beta; \mathcal{C}\mathcal{P}(\mathbf{n})).$$

From (3.3) and (3.6) with $\beta = 1$

$$\sum \frac{x_1^{n_1}}{n_1!} \cdots \frac{x_m^{n_m}}{n_m!} v(\mathcal{C} \mathcal{P}(\mathbf{n})) = -\log V_m.$$

But $v(\beta; \pi) = \beta v(\pi)$ for every connected permutation, so that

$$\sum \frac{x_1^{n_1}}{n_1!} \cdots \frac{x_m^{n_m}}{n_m!} v(\beta; \mathcal{C} \mathcal{P}(\mathbf{n})) = -\beta \log V_m,$$

and finally (3.5) holds because of (3.6). $\square$

4. A combinatorial proof of Theorem 2. By the definition of the Laguerre polynomials

$$n_i! L_{n_i}^{(\alpha)}(x) = \sum_{k_i=0}^{n_i} (-1)^{k_i} \binom{n_i}{k_i} (\alpha + 1 + k_i)_{n_i - k_i} x^{k_i} \quad (i = 1, \dots, m).$$

In (1.3) and (1.5) expand each Laguerre polynomial and integrate term by term using the fact that $(1/\Gamma(\alpha + 1)) \int_0^\infty e^{-x} x^{n+\alpha} dx = (\alpha + 1)_n$. This leads to:

$$(4.1) \quad J(\mathbf{n}; \Lambda; \alpha) = \sum_{\mathbf{k}} (\alpha + 1)_{k_1 + \dots + k_m} \prod_{i=1}^m (-\lambda_i)^{k_i} \binom{n_i}{k_i} (\alpha + 1 + k_i)_{n_i - k_i},$$

where the first summation is over all sequences $\mathbf{k} = (k_1, \dots, k_m)$ with $0 \leq k_i \leq n_i, \dots, 0 \leq k_m \leq n_m$. From the lemma of § 2, $J(\mathbf{n}; \Lambda; \alpha)$ can also be written as

$$J(\mathbf{n}; \Lambda; \alpha) = \sum_{\mathbf{T}} \prod_{i=1}^m (-\lambda_i)^{|A_i \setminus T_i|} \sum_{\pi} w(\pi),$$

where the first summation is over all sequences $\mathbf{T} = (T_1, \dots, T_m)$ such that $T_i \subset A_i$ for all i and the second one over all permutations π satisfying $T_i \subset \text{Inc}_i \pi$ for all i . Hence,

$$\begin{aligned} J(\mathbf{n}; \Lambda; \alpha) &= \sum_{\pi} w(\pi) \sum_{\mathbf{T}} \prod_{i=1}^m (-\lambda_i)^{|A_i \setminus T_i|} \quad [\text{with } T_i \subset \text{Inc}_i \pi] \\ &= \sum_{\pi} w(\pi) \prod_{i=1}^m (-\lambda_i)^{|A_i \setminus \text{Inc}_i \pi|} \sum_{\mathbf{T}} \prod_{i=1}^m (-\lambda_i)^{|\text{Inc}_i \pi \setminus T_i|} \quad [\text{with } T_i \subset \text{Inc}_i \pi] \\ &= \sum_{\pi} w(\pi) \prod_{i=1}^m (-\lambda_i)^{|A_i \setminus \text{Inc}_i \pi|} \prod_{i=1}^m (1 - \lambda_i)^{|\text{Inc}_i \pi|}, \end{aligned}$$

the binomial identity being used in the last step of that derivation. Thus,

$$J(\mathbf{n}; \Lambda; \alpha) = \sum_{\pi} W(\pi),$$

which is exactly the statement of Theorem 2. $\square$

5. An analytical proof of Theorem 2. There is another way of expressing the W -weight. For each ordered pair (i, j) of integers lying between one and m and each permutation π in $\mathcal{P}(\mathbf{n})$ let $\gamma_{ij} = \gamma_{ij}(\pi)$ be the number of elements x in A_i that are mapped into A_j (i.e., $\gamma_{ij} = |\pi(A_i) \cap A_j|$). Define the W' -weight of π by

$$(5.1) \quad W'(\pi) = w(\pi) \prod_{i=1}^m (1 - \lambda_i)^{\gamma_{ii}} \prod_{\substack{i,j \\ i \neq j}} (-\sqrt{\lambda_i \lambda_j})^{\gamma_{ij}}.$$

LEMMA. We have $W'(\pi) = W(\pi)$.

Proof. As $\sum_{i,j,i \neq j} \gamma_{ij} = \sum_i |\text{Inc}_i \pi|$, it suffices to show that

$$\prod_i \lambda_i^{|\text{Inc}_i \pi|} = \prod_{\substack{i,j \\ i \neq j}} (\sqrt{\lambda_i \lambda_j})^{\gamma_{ij}}.$$

For each $x \in A$ let $c(x) = i$, if and only if $x \in A_i$. Then

$$\prod_i \lambda_i^{|\text{Inc}_i \pi|} = \prod_x \lambda_{c(x)} \quad (x \in A \setminus \text{Inc } \pi).$$

For each nonincestuous element x let x^* be the nearest nonincestuous element to x in the following sense. Let $k = \min \{l : l \geq 1, \pi^l(x) \in A \setminus \text{Inc } \pi\}$ and put $x^* = \pi^k(x)$. Clearly, x^* is uniquely defined and $x \rightarrow x^*$ is a bijection of $A \setminus \text{Inc } \pi$ onto itself. Then

$$\prod_x \lambda_{c(x)} = \prod_x \sqrt{\lambda_{c(x)} \lambda_{c(x^*)}} \quad (x \in A \setminus \text{Inc } \pi).$$

As $c(x^*) = c(\pi(x))$, we also have

$$\prod_x \lambda_{c(x)} = \prod_x \sqrt{\lambda_{c(x)} \lambda_{c(\pi(x))}} = \prod_{\substack{i,j \\ i \neq j}} (\sqrt{\lambda_i \lambda_j})^{\gamma_{ij}}. \quad \square$$

In the notation of § 3 take

$$(5.2) \quad b(i, j) = \begin{cases} 1 - \lambda_i & \text{if } i = j, \\ -\sqrt{\lambda_i \lambda_j} & \text{if } i \neq j. \end{cases}$$

The determinant $V_m = \det(\delta_{ij} - b(i, j)x_j)$ has been calculated by Askey et al. [As-Is-Ko, Thm. 4]. They showed that a simple induction argument gives

$$(5.3) \quad V_m = \prod_{i=1}^m (1 - x_i) + \sum_{i=1}^m \lambda_i x_i \prod_{j \neq i} (1 - x_j).$$

On the other hand, the generating function (1.1) for the Laguerre polynomials yields the identity

$$(5.4) \quad \sum_{\mathbf{n}} J(\mathbf{n}; \Lambda; \alpha) \frac{x_1^{n_1}}{n_1!} \cdots \frac{x_m^{n_m}}{n_m!} = V_m^{-(\alpha+1)},$$

almost immediately, as noticed by the same authors [As-Is-Ko, Thm. 2]. By comparing (5.4) with (3.5), we conclude that

$$J(\mathbf{n}; \Lambda; \alpha) = v(\alpha + 1; \mathcal{P}(\mathbf{n})).$$

Finally, expand $v(\alpha + 1; \pi)$ in terms of the $b(i, j)$'s:

$$v(\alpha + 1; \pi) = (\alpha + 1)^{\text{cyc } \pi} \prod_{i,j} b(i, c\pi(i, j)) = W'(\pi).$$

Then, Theorem 2 is proved by using the previous lemma. $\square$

6. Positivity. As mentioned in the Introduction, the positivity of $J(\mathbf{n}; \Lambda; \alpha)$ for $m = 3$ (Koornwinder's inequality) implies a general result for an arbitrary m by means of the orthogonality relation and the old expansion of the Laguerre polynomials [As-Is-Ko]. However, that inequality remains the crucial point. It is next restated and re-proved.

THEOREM 3. Let $m = 3$ and $\Lambda = (\lambda_1 = \lambda, \lambda_2 = 1 - \lambda, \lambda_3 = 1)$. Then, for $\alpha \geq 0$ and $0 \leq \lambda \leq 1$ we have $J(\mathbf{n}, \Lambda; \alpha) \geq 0$.

Proof. For $m = 3$ the summation $\sum_{\pi} W(\pi)$ in (1.7) becomes

$$\sum_{\pi} W(\pi) = \sum_{\pi} (\alpha + 1)^{\text{cyc } \pi} \prod_{i=1}^3 (1 - \lambda_i)^{|\text{Inc}_i \pi|} (-\lambda_i)^{|A_i \setminus \text{Inc}_i \pi|},$$

where the summation is over the permutations of π of $A = A_1 + A_2 + A_3$ having no incestuous elements in A_3 (because $\lambda_3 = 1$). As in § 5 let $\gamma_{ij} = |\pi(A_i) \cap A_j|$ be the number of elements that go from A_i to A_j . The sign of the W -weight of π is equal to $(-1)^{s(\pi)}$, where

$$s(\pi) = \gamma_{12} + \gamma_{13} + \gamma_{21} + \gamma_{23} + \gamma_{31} + \gamma_{32} = 2(n_1 + n_3 - \gamma_{11}) - (\gamma_{13} + \gamma_{31}).$$

Thus

$$(6.1) \quad \text{sgn } W(\pi) = (-1)^{\gamma_{13} + \gamma_{31}}.$$

Furthermore, the exponent of λ_1 in $W(\pi)$ is equal to

$$\gamma_{12} + \gamma_{13} + \gamma_{22} = n_2 - n_3 + \gamma_{13} + \gamma_{31},$$

while the exponent of λ_2 is equal to $\gamma_{11} + \gamma_{21} + \gamma_{23} = n_1 + n_3 - \gamma_{13} - \gamma_{31}$. Hence

$$(6.2) \quad W(\pi) = (\alpha + 1)^{\text{cyc } \pi} (-1)^{\gamma_{13} + \gamma_{31}} \lambda_1^{n_1 - n_3 + \gamma_{13} + \gamma_{31}} \lambda_2^{n_1 + n_3 - \gamma_{13} - \gamma_{31}}.$$

Thus, when $\alpha = 0$, the weight $W(\pi)$ depends only on γ_{13} and γ_{31} . Assume that $\alpha = 0$ and compute the number of permutations π such that $\gamma_{13} = a$ and $\gamma_{31} = b$. This is easy if we have in mind the representation of each permutation as a two-line matrix. The number of ways of placing the elements of the set $\pi(A_1) \cap A_3$ and $\pi(A_2) \cap A_3$ is equal to $\binom{n_1}{a} \binom{n_3}{n_3-a} n_3!$. In the same manner, $\binom{n_1}{b} \binom{n_3}{n_3-b} n_3!$ is the number of ways of placing the elements of $\pi(A_3) \cap A_1$ and $\pi(A_3) \cap A_2$. There remain $(n_1 + n_2 - n_3)!$ positions for the other elements to be placed. Thus,

$$\begin{aligned} \sum_{\pi} W(\pi) &= \sum_{a,b} (-1)^{a+b} \binom{n_1}{a} \binom{n_2}{n_3-a} n_3! \binom{n_1}{b} \binom{n_2}{n_3-b} n_3! \\ &\quad \times (n_1 + n_2 - n_3)! \lambda_1^{n_1 - n_3 + a + b} \lambda_2^{n_1 + n_3 - a - b} \\ &= (n_3!)^2 \lambda^{n_2 - n_3} \lambda_2^{n_1 + n_3} (n_1 + n_2 - n_3)! \\ &\quad \times \left(\sum_a \binom{n_1}{a} \binom{n_2}{n_3-a} (-1)^a \left(\frac{\lambda_1}{\lambda_2} \right)^a \right)^2 \\ &\geq 0. \end{aligned}$$

This proves Koornwinder's inequality for $\alpha = 0$.

For an arbitrary $\alpha > 0$ let us make use again of the argument of Gillis, Reznick, and Zeilberger [Gi-Re-Ze, Prop. 1]. The determinant V_3 corresponding to the previous W is easy to calculate, either directly, or by means of (5.3):

$$V_3 = 1 - \lambda_2 x_1 - \lambda_1 x_2 - x_3 [\lambda_1 x_1 + \lambda_2 x_2 - x_1 x_2].$$

Put $X = 1 - \lambda_2 x_1 - \lambda_1 x_2$ and $Y = \lambda_1 x_1 + \lambda_2 x_2 - x_1 x_2$, so that $V_3 = X - x_3 Y$. Hence

$$V_3^{-(\alpha+1)} = X^{-(\alpha+1)} [1 - x_3 Y X^{-1}]^{-(\alpha+1)}$$

$$= X^{-\alpha} \sum_r \frac{(\alpha+1)_r}{r!} \frac{Y^r}{X^{r+1}} x_3^r.$$

But $V_3^{-1} = (X - x_3 Y)^{-1} = \sum_r (Y^r / X^{r+1}) x_3^r$ has positive coefficients. This is another way of saying that Koornwinder's inequality holds for $\alpha = 0$. Hence the same inequality is true for every $\alpha \geq 0$. $\square$

Remark. The idea of expressing the summation $\sum_{\pi} W(\pi)$ for $\alpha = 0$ as a square of a polynomial is basically due to Ismail and Tamhankar [Is-Ta], even though they made their calculations with the determinant V_3 itself. The proof given above is only an adaptation of the derivation of Gillis and Zeilberger [Gi-Ze] to the permutation combinatorial set-up developed in this paper.

Acknowledgment. The first author thanks Jiang Zeng for bringing his attention to the fact that Theorem 1 is just a particular case of Theorem 2.

REFERENCES

- [As1] R. ASKEY, *Linearization of the product of orthogonal polynomials*, in Problems in Analysis (A symposium in honor of Salomon Bochner), Robert C. Gunning, ed., Princeton University Press, Princeton, NJ, 1970, pp. 131–138.
- [As2] ———, *Orthogonal and Special Functions*, CBMS–NSF Regional Conference Series in Applied Mathematics 21, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1975.
- [As-Is] R. ASKEY AND M. E. H. ISMAIL, *Permutation problems and special functions*, Canad. J. Math. 28 (1976), pp. 853–874.
- [As-Is-Ko] R. ASKEY, M. E. H. ISMAIL, AND T. KOORNWINDER, *Weighted permutation problems and Laguerre polynomials*, J. Combin. Theory Ser. A, 25 (1978), pp. 277–287.
- [As-Ga] R. ASKEY AND G. GASPER, *Convolution structures for Laguerre polynomials*, J. d'Anal. Math., 31 (1977), pp. 46–48.
- [Ev-Gi] S. EVEN AND J. GILLIS, *Derangements and Laguerre polynomials*, Proc. Cambridge Phil. Soc., 79 (1976), pp. 135–143.
- [Fo] D. FOATA, *La périe génératrice exponentielle dans les problèmes d'énumération*, Presses de l'Université de Montréal, Montréal, Quebec, Canada, 1974.
- [Fo-Sch] D. FOATA AND M.-P. SCHÜTZENBERGER, *Théorie géométrique des polynômes eutériens*, Lecture Notes in Mathematics 138, Springer-Verlag, Berlin, 1970.
- [Fo-St] D. FOATA AND V. STREHL, *Combinatorics of Laguerre polynomials*, in Enumeration and Design, D. M. Jackson and S. A. Vanstone, eds., Waterloo, Ontario, Canada, June–July 1982, pp. 123–140; Academic Press, Toronto, 1984.
- [Gi-Ze] J. GILLIS AND D. ZEILBERGER, *A direct combinatorial proof of a positivity result*, European J. Combin., 4 (1983), pp. 221–223.
- [Gi-Re-Ze] J. GILLIS, B. REZNICK, AND D. ZEILBERGER, *On elementary methods in positivity theory*, SIAM J. Math. Anal., 14 (1983), pp. 396–398.
- [Is-Ta] M. E. H. ISMAIL AND M. V. TAMHANKAR, *A combinatorial approach to some positivity problems*, SIAM J. Math. Anal., 10 (1979), pp. 478–485.
- [Ja] D. M. JACKSON, *Laguerre polynomials and derangements*, Proc. Cambridge Phil. Soc., 80 (1976), pp. 213–214.
- [Ko] T. KOORNWINDER, *Positivity proofs for linearization and connection coefficients of orthogonal polynomials satisfying an addition theorem*, J. London Math. Soc., 18 (1978), pp. 101–114.
- [Mac] PERCY ALEXANDER MACMAHON, *Combinatory Analysis*, Vol. 1, Cambridge, University Press, 1915; reprinted by Chelsea, New York, 1955.
- [Ra] M. RAHMAN, *A non-negative representation of the linearization coefficients of the product of Jacobi polynomials*, Canad. J. Math., 33 (1981), pp. 915–928.
- [Ri] J. RIORDAN, *An Introduction to Combinatorial Analysis*, New York, John Wiley, 1959.
- [Sa-Vi] M. DE SAINTE-CATHERINE AND G. VIENNOT, *Combinatorial interpretation of integrals of products of Hermite, Laguerre and Tchebycheff polynomials*, in Polynômes orthogonaux et applications, C. Brezinski, et al. eds., Bar-le-Duc, 1984, pp. 120–128; Lecture Notes in Mathematics 1175, Springer-Verlag, Berlin, 1985.
- [Ze] J. ZENG, *Linéarisation de produits de polynômes de Meixner*, Krawtchouk et Charlier, Strasbourg, 1987, submitted.

PARALLEL SYMMETRY-BREAKING IN SPARSE GRAPHS*

ANDREW V. GOLDBERG[†], SERGE A. PLOTKIN[‡], AND GREGORY E. SHANNON[§]

Abstract. This paper describes efficient deterministic techniques for breaking symmetry in parallel. These techniques work well on rooted trees and graphs of constant degree or genus. The primary technique allows us to 3-color a rooted tree in $O(\lg^* n)$ time on an EREW PRAM using a linear number of processors. These techniques are used to construct fast linear processor algorithms for several problems, including the problem of $(\Delta + 1)$ -coloring constant-degree graphs and 5-coloring planar graphs. Lower bounds for 2-coloring directed lists and for finding maximal independent sets in arbitrary graphs are also proved.

Key words. parallel algorithms, graph coloring, maximal independent set, planar graphs

1. Introduction. Some problems for which trivial sequential algorithms exist appear to be much harder to solve in a parallel framework. When converting a sequential algorithm to a parallel one, at each step of the parallel algorithm we must choose a set of operations that may be executed in parallel. Often, we must choose these operations from a large set of symmetrical operations where interdependencies prevent simultaneous execution of all the operations in the set. Symmetry-breaking techniques enable the algorithm to select a large subset of independent operations.

Finding a maximal independent set (MIS) of a graph is a good example of the necessity of symmetry-breaking. At any step, a parallel MIS algorithm might have many candidate nodes to add to the independent set. However, due to adjacency constraints, not all of these nodes can be added simultaneously. A symmetry-breaking technique is therefore needed to find a large set of nodes to add, as has been done in previous parallel MIS algorithms [GS87], [KW85], [Lub86].

Previous symmetry-breaking techniques have focused on randomization. It is often desirable, however, to have a deterministic algorithm. Karp and Wigderson [KW85], and Luby [Lub86] proposed methods to convert certain randomized algorithms into deterministic ones. Their methods, however, significantly increase the number of processors used.

In many cases it is sufficient to break symmetry in sparse graphs. In this paper we introduce deterministic symmetry-breaking techniques for sparse graphs that use a linear number of processors. Our primary technique allows us to 3-color a rooted tree in $O(\lg^* n)$ time on a CREW PRAM. This technique was motivated by the deterministic coin-flipping technique developed by Cole and Vishkin [CV86].

* Received by the editors October 1, 1987; accepted for publication (in revised form) April 6, 1988. A preliminary version of this paper appeared in the Proceedings of the 19th Annual ACM Symposium on Theory of Computing, May 1987.

[†] Department of Computer Science, Stanford University, Stanford, California 94305. Part of this research was done while the author was at the Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts. This research was supported in part by a Fannie and John Hertz Foundation Fellowship and Defense Advanced Research Projects Agency contract N00014-80-C-0622.

[‡] Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139. This work was supported by Defense Advanced Research Projects Agency contract N00014-80-C-0622. Part of this work was done while the author was at AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

[§] Computer Sciences Department, Purdue University, West Lafayette, Indiana 47907. This work was supported in part by Hewlett-Packard's Faculty Development Program, National Science Foundation grant DCR-8320124, and Office of Naval Research contract N00014-86-K-0689.

We use our techniques to develop the linear-processor algorithms listed below.

- For graphs whose maximum degree is Δ , we give an $O((\lg \Delta)(\Delta^2 + \lg^* n))$ -time EREW PRAM algorithm for $(\Delta + 1)$ -coloring and for finding a maximal independent set.
- For planar graphs, we give 7-coloring, MIS, and maximal matching algorithms that run in $O(\lg n)$ time on a CRCW PRAM and in $O(\lg^2 n)$ time on an EREW PRAM.
- We give an $O(\lg n \lg^* n)$ -time CRCW PRAM algorithm for 5-coloring an embedded planar graph.

The above results improve the running time and processor bounds for the respective problems. The best deterministic linear-processor algorithm for finding MIS [GS87] runs in $O(\lg^4 n)$ time on constant-degree graphs, compared to $O(\lg^* n)$ time of our algorithm. The 5-coloring algorithms for planar graphs described in [BK87], [Nao86] use $O(\lg^3 n)$ time and the same (large) number of processors as needed by Luby's MIS subroutine [Lub86]. The $O(\lg^3 n)$ running time of the maximal matching algorithm due to Israeli and Shiloach [IS86] can be reduced to $O(\lg^2 n)$ in the restricted case of planar graphs, but our algorithm is faster.

Although in this paper we have limited ourselves to the application of our techniques for the design of parallel algorithms for the PRAM model of computation, the same techniques can be applied in a distributed model of computation [Awe85], [GHS83]. Moreover, the $\Omega(\lg^* n)$ lower bound for the MIS problem on a chain in the distributed model implies that our symmetry-breaking technique is optimal in this model [Awe87], [Lin87].

Since we can 3-color a rooted tree in $O(\lg^* n)$ time, it is natural to ask if a rooted tree can be 2-colored as quickly. We answer this question by giving an $\Omega(\lg n / \lg \lg n)$ lower bound for 2-coloring of a rooted tree. We also present an $\Omega(\lg n / \lg \lg n)$ lower bound for finding a maximal independent set in a general graph, thus answering the question posed by Luby [Lub86].

This paper is organized as follows. In § 2 we present definitions, notation, and computation model details. In § 3 we present the algorithm for 3-coloring rooted trees. In § 4 we use this algorithm to $(\Delta + 1)$ -color constant-degree graphs. In § 5 we use results of § 4 to develop algorithms for planar graphs. In § 6 we prove the lower bounds mentioned earlier.

2. Preliminaries. This section describes the assumptions about the computational model and introduces the notation used throughout the paper. We consider simple, undirected graphs with n vertices and m edges. The maximum degree of a graph is denoted by Δ . The graph induced by a set of nodes X is denoted by $G[X]$.

Given a graph $G = (V, E)$, we say that a subset of nodes $I \subseteq V$ is *independent* if no two nodes in I are adjacent. A *coloring* of a graph G is an assignment $C: V \rightarrow I^+ \cup \{0\}$ of nonnegative (not necessarily consecutive) integers (colors) to nodes of the graph. A coloring is *valid* if no two adjacent nodes have the same color. The bits are numbered from zero, and the i th bit in the color of a node v is denoted by $C_v(i)$. A subset of edges $M \subseteq E$ is a *matching* if each pair of distinct edges in M has no nodes in common.

The following problems are discussed in the paper:

- The node-coloring problem: find a valid coloring of a given graph that uses at most $\Delta + 1$ colors.
- The maximal independent set (MIS) problem: find a maximal independent set of vertices in a given graph.
- The maximal matching (MM) problem: find a maximal matching in a given graph.

We make a distinction between *unrooted* and *rooted* trees. In a rooted tree, each nonroot node knows which of its neighbors is its parent.

The following notation is used:

$$\begin{aligned}\lg x &= \log_2 x, \\ \lg^{(1)} x &= \lg x, \\ \lg^{(i)} x &= \lg \lg^{(i-1)} x, \\ \lg^* x &= \min \{ i \mid \lg^{(i)} x \leq 2 \}.\end{aligned}$$

We assume a PRAM model of computation [BH85], [FW78] where each processor is capable of executing simple word and bit operations. The word width is assumed to be $O(\lg n)$. The word operations we use include bit-wise Boolean operations, integer comparisons, and unary-to-binary conversion. Each processor P has a unique *identification number* $O(\lg n)$ bits wide, which we denote by $\text{PE-ID}(P)$. We use adjacency lists to represent the graph, assigning a processor to each edge and each node of the graph. We use exclusive-read exclusive-write (EREW) PRAM, concurrent-read exclusive-write (CREW) PRAM, and concurrent-read concurrent-write (CRCW) PRAM, as appropriate. The write conflicts in CRCW PRAM are assumed to be resolved arbitrarily. All lower bounds are proven for a CRCW PRAM with a polynomial number of processors.

3. Coloring rooted trees. This section describes an $O(\lg^* n)$ -time algorithm for 3-coloring rooted trees. We first describe an $O(\lg^* n)$ -time algorithm for 6-coloring rooted trees and then show how to transform a 6-coloring of a rooted tree into a 3-coloring in constant time.

The procedure *6-Color-Rooted-Tree* is shown in Fig. 1. This procedure accepts a rooted tree $T = (V, E)$ and 6-colors it in time $O(\lg^* n)$. Starting from the valid coloring given by the processor IDs, the procedure iteratively reduces the number of bits in the color descriptions by recoloring each nonroot node v with the color obtained by concatenating the index of a bit in which C_v differs from $C_{\text{parent}(v)}$ and the value of this bit. The root r concatenates zero and $C_r[0]$ to form its new color.

```

Procedure 6-Color-Rooted-Tree ( $T$ )
   $N_c \leftarrow n$ ;
  for all  $v \in V$  in parallel do  $C_v \leftarrow \text{PE-ID}(v)$ ;
  while  $N_c > 6$  do
    for all  $v \in V$  in parallel do begin
      if  $v$  is the root then begin
         $i_v \leftarrow 0$ ;
         $b_v \leftarrow C_v(0)$ ;
      end;
      else begin
         $i_v \leftarrow \min \{ l \mid C_v(l) \neq C_{\text{parent}(v)}(l) \}$ ;
         $b_v \leftarrow C_v(i_v)$ ;
      end;
       $C_v \leftarrow i_v b_v$ ;
       $\langle \langle i_0 \text{ concatenated with } b_v \rangle \rangle$ 
    end;
     $N_c \leftarrow \max \{ C_v \mid v \in V \} + 1$ ;
  end;
end.

```

FIG. 1. *The Coloring Algorithm for rooted trees.*

THEOREM 1. *The algorithm 6-Color-Rooted-Tree produces a valid 6-coloring of a tree in $O(\lg^* n)$ time on a CREW PRAM using a linear number of processors.*

Proof. First we prove by induction that the coloring computed by the algorithm is valid, and then we prove the upper bound on the execution time.

Assuming that the coloring C is valid at the beginning of an iteration, we show that the coloring at the end of the iteration is also valid. Let v and w be two adjacent nodes with v being the parent of w . In the algorithm, w chooses some index i such that $C_v(i) \neq C_w(i)$ and v chooses some index j such that $C_v(j) \neq C_{\text{parent}(v)}(j)$. The new color of w is $\langle i, C_w(i) \rangle$ and the new color of v is $\langle j, C_v(j) \rangle$. If $i \neq j$, the new colors are different and we are done. On the other hand, if $i = j$, then $C_v(i)$ cannot be equal to $C_w(i)$ by the definition of i , and again the colors are different. Hence, the validity of the coloring is preserved.

Now we show that the algorithm terminates after $O(\lg^* n)$ iterations. Let L_k denote the number of bits in the representation of colors after k iterations. For $k = 1$ we have

$$L_1 = \lceil \lg L \rceil + 1.$$

$\lceil \lg^{(k-1)} L \rceil \geq 4$. Assume for some k we have $L_{k-1} \leq 2\lceil \lg^{(k-1)} L \rceil + 2$ and $\lceil \lg^{(k)} L \rceil \geq 2$. Then

$$\begin{aligned} L_k &= \lceil \lg L_{k-1} \rceil + 1 \\ &\leq \lceil \lg (2 \lg^{(k-1)} L) \rceil + 1 \\ &\leq \lceil \lg^{(k)} L \rceil + 2. \end{aligned}$$

Therefore, as long as $\lceil \lg^{(k)} L \rceil \geq 2$,

$$L_k \leq \lceil \lg^{(k)} L \rceil + 2.$$

Hence, the number of bits in the representation of colors L_k decreases until, after $O(\lg^* n)$ iterations, $\lceil \lg^{(k)} L \rceil$ becomes one and L_k reaches the value of three (the solution of $L = \lceil \lg L \rceil + 1$). Another iteration of the algorithm produces a 6-coloring: three possible values of the index i_v and two possible values of the bit b_v . The algorithm terminates at this point.

When we use concurrent-read, each node determines its parent's color in constant time. Given two colors, C_v and C_w , we can compute the smallest index j such that the j th bit of C_v differs from the j th bit of C_w by computing $j = \text{unary-to-binary}(|C_v - C_w| \text{ XOR } (|C_v - C_w| - 1))$. Hence, each node can compute the new color independently in constant time. Therefore, each iteration takes constant time and the algorithm uses $O(\lg^* n)$ time overall. Note that no concurrent-write capabilities are required; for constant-degree trees the concurrent-read capability is not needed either. $\square$

We now describe the algorithm 3-Color-Rooted-Tree that 3-colors a rooted tree. The algorithm first applies 6-Color-Rooted-Tree to produce a valid 6-coloring of the tree. Then it executes three stages, each time reducing the number of colors by one.

Each stage works as follows. By *shifting down* the coloring we mean recoloring each nonroot node with the color of its parent and recoloring the root with a color different from its current color. To remove the color $c \in \{3, 4, 5\}$, first shift down the current coloring. Then, recolor each node of color c with the smallest color different from its parent's and children's colors.

THEOREM 2. *Given a rooted tree T , the algorithm 3-Color-Rooted-Tree constructs a valid 3-coloring of T using n processors and $O(\lg^* n)$ time on a CREW PRAM.*

Proof. After a shift of colors, the children of any node have the same color. Thus each node is adjacent to nodes of at most two different colors. Therefore, each stage of

the algorithm reduces the number of colors by one, as long as the number of colors is greater than three. Each stage takes a constant time on a CREW PRAM. The theorem follows from Theorem 1. $\square$

To describe the subsequent algorithms, we introduce the concept of a pseudoforest [PQ82]. A *pseudoforest* of $G = (V, E)$ is a directed graph $G' = (V, E')$, such that $(u, v) \in E' \Rightarrow \{u, v\} \in E$ and outdegree of any node is at most one. A *maximal pseudoforest* of $G = (V, E)$ is a directed graph $G' = (V, E')$, such that $(u, v) \in E' \Rightarrow \{u, v\} \in E$ and outdegree of any node in G' is one, unless this node is zero-degree in G . Nodes with zero out-degree are *roots* of the pseudoforest. We assume that graphs are represented by adjacency lists, and therefore a maximal pseudoforest can be constructed in (parallel) constant time by choosing an arbitrary adjacent edge for every node and directing this edge outward.

The coloring algorithms presented in this section work for pseudoforests as well as for rooted trees. Therefore, a pseudoforest can be 3-colored in $O(\lg^* n)$ time on an CRCW PRAM using a linear number of processors. We shall call the procedure for 3-coloring pseudoforests *3-Color-Pseudoforest*. Note that an odd cycle is a pseudoforest that cannot be colored in less than three colors, and therefore the number of colors used by the procedure *3-Color-Pseudoforest* is optimal in this case.

Any tree can be 2-colored. In fact, it is easy to 2-color a tree in polylogarithmic time. For example, we can use tree prefix operations [LM86], [MR85] to compute the distance from each node to the root, and color even level nodes with one color and odd level nodes with the other color. It is harder to find a 2-coloring of a rooted tree in parallel, however, than it is to find a 3-coloring of a rooted tree. In § 6 we show a lower bound of $\Omega(\lg n / \lg \lg n)$ on 2-coloring of a directed list on a CRCW PRAM with a polynomial number of processors, which implies the same lower bound for rooted trees.

4. Coloring constant-degree graphs. The method for coloring rooted trees, described in the previous section, is a generalization of the deterministic coin-flipping technique described in [CV86]. The method can be generalized even further [GP87b] to color constant-degree graphs in a constant number of colors. In the generalized algorithm, a current color of a node is replaced by a new color obtained by looking at each neighbor, appending the index of a bit in which the current color of the node is different from the neighbors' color to the value of the bit in the node color, and concatenating the resulting strings. This algorithm runs in $O(\lg^* n)$ time, but the number of colors, although constant as a function of n , is exponential in the degree of the graph.

In this section we show how to use the procedure *3-Color-Pseudoforest*, described in the previous section, to color a constant-degree graph with $(\Delta + 1)$ colors.

The algorithm *Color-Constant-Degree-Graph* that colors a constant-degree graph $G = (V, E)$ with $(\Delta + 1)$ colors is presented in Fig. 2. The algorithm consists of two phases. In the first phase we iteratively construct a maximal pseudoforest and remove its edges from G . This phase continues until no edges remain, at which point we color all the nodes with one color. Then we color all the pseudoforests with three colors in parallel.

In the second phase we iteratively return the edges of the current pseudoforest, each time recoloring the nodes to maintain a consistent coloring. At the beginning of each iteration of this phase, the edges E' of the current pseudoforest are added, making the existing $(\Delta + 1)$ -coloring inconsistent. The forest E' is already colored with three colors. Now, each node has two colors—one from the coloring at the previous iteration and one from the coloring of the forest. The pairs of colors form a valid $3(\Delta + 1)$ -coloring of the graph. The iteration finishes by enumerating the color classes, recoloring each node of the current color with a color from $\{0, \dots, \Delta\}$ that is different from the colors of its

Procedure *Color-Constant-Degree-Graph*.

```

 $E' \leftarrow \{(v, w) \mid \{v, w\} \in E\};$ 
for  $i = 0$  to  $\Delta$  do begin                                     <<the first phase>>
    for all  $v \in V$  in parallel do
        if  $\exists (v, u) \in E'$  then  $E_i \leftarrow E_i + (v, u);$ 
         $E' \leftarrow E' - E_i;$                                      << $E_i$  are edges of a maximal pseudoforest>>
    end;
    for all  $v \in V$  in parallel do                                     <<initial coloring>>
         $C(v) \leftarrow 0;$ 
    for all  $0 \leq i \leq \Delta$  in parallel do                             <<color the pseudoforests>>
         $C_i \leftarrow \text{3-Color-Pseudoforest}(V, E_i);$ 
    for  $i \leftarrow \Delta$  down to  $0$  do begin                             <<the second phase>>
         $E' \leftarrow E' + E_i;$ 
        for  $k \leftarrow 1$  to  $2, j \leftarrow 0$  to  $\Delta$  do
             $V' \leftarrow V;$ 
            for all  $v \in V'$  in parallel do
                if  $C(v) = j$  and  $C_i(v) = k$ 
                    then begin
                         $C(v) \leftarrow \max \{ \{0, 1, \dots, \Delta\} - \{C(w) \mid (v, w) \in E'\} \};$ 
                         $V' \leftarrow V' - \{v\};$ 
                    end;
            end;
        end;
    end;
end.

```

FIG. 2. The Coloring Algorithm for constant degree graphs.

neighbors. We can recolor all the nodes of the same color in parallel because they are independent.

THEOREM 3. *The algorithm Color-Constant-Degree-Graph colors the graph with $(\Delta + 1)$ colors and runs in $O((\lg \Delta)(\Delta^2 + \lg^* n))$ time on an EREW PRAM using a linear number of processors.*

Proof. At each iteration all edges of the maximal pseudoforest are removed. The definition of a maximal pseudoforest implies that each node that still has neighbors in the beginning of an iteration has at least one edge removed during that iteration and therefore its degree decreases. After at most Δ iterations, E' is empty. The running time of each iteration is determined by the time required to select an unused edge out of an edge list. On an EREW PRAM, an unused edge can be selected in $O(\lg \Delta)$ time. The pseudoforests are edge-disjoint and therefore can be colored in parallel. By Theorem 2, this takes $O(\lg \Delta \lg^* n)$ time on an EREW PRAM. The $\lg \Delta$ factor appears because we do not use the concurrent-read capability; a node must broadcast its color to its children using, for example, recursive doubling. The total time bound for the first stage is therefore $O((\lg \Delta)(\Delta + \lg^* n))$.

The second phase terminates in at most Δ iterations as well. For each pseudoforest we execute $O(\Delta)$ steps, one step per each possible color. Using sorting [Col86], each step can be implemented to run in $O(\lg \Delta)$ time. Hence, one iteration of the second phase takes $O(\Delta \lg \Delta)$ time, leading to an overall $O((\lg \Delta)(\Delta^2 + \lg^* n))$ running time for the second stage of the algorithm and for the algorithm itself. $\square$

Given a $(\Delta + 1)$ -coloring of a graph, we can find an MIS of the graph by iterating over the colors, taking all the remaining nodes of the current color, adding them to the independent set, and removing them and all their neighbors from the graph. (We refer to this procedure as *Constant-Degree-MIS* in the subsequent sections.) The running time of this algorithm is dominated by the running time of the *Color-Constant-Degree-Graph* procedure. The following theorem states this fact formally.

THEOREM 4. *An MIS in constant-degree Δ graphs can be found in*

$$O((\lg \Delta)(\Delta^2 + \lg^* n))$$

time on an EREW PRAM using a linear number of processors.

Remark. The proofs of Theorems 3 and 4 imply that the algorithms *Color-Constant-Degree-Graph* and *Constant-Degree-MIS* have polylogarithmic running times for graphs with polylogarithmic maximum degrees. For graphs with arbitrary maximum degree we can use the following algorithm. First, the graph is partitioned into two subgraphs with approximately equal number of nodes, and the subgraphs are recursively colored in $\Delta + 1$ colors. Then we iterate through all the colors of one of the subgraphs, recoloring each node with a color different from the colors of all of its neighbors. We can find this color using sorting in $O(\lg \Delta)$ time [Col86]. This algorithm colors a graph with a maximum degree of Δ with $\Delta + 1$ colors in $O(\Delta \lg \Delta \lg n)$ time.

The above algorithms can be implemented in the distributed model of computation [Awe85], [GHS83], where processors have fixed connections determined by the input graph. The algorithms in the distributed model achieve the same $O(\lg^* n)$ bound as in the EREW PRAM model. It was recently shown that $\Omega(\lg^* n)$ time is required in the distributed model to find a maximal independent set on a chain [Awe87], [Lin87]. Our algorithms are therefore optimal (to within a constant factor) in the distributed model.

In [Sha86], *flat forests* are used to develop a linear processor constant-degree MIS algorithm that uses time exponential in Δ . A forest is *flat* if each of its trees, when properly oriented, has a height of at most one, and any zero-degree node in the forest is zero-degree in the input graph. Using the techniques introduced in this section, we can find a flat forest of a graph by proceeding as follows. Find a maximal pseudoforest $P = G(V, E')$. Note that there exists a flat forest $F = (V, E'')$, such that $E'' \subseteq E'$. Use the algorithm *3-Color-Pseudoforest* to find a 3-coloring of the pseudoforest P and subsequently find an MIS I of P . Each node $v \notin I$ adds an edge (v, u) to E'' such that $u \in I$. Each node in I with no adjacent edges in E'' , but some adjacent edges in E' , chooses one adjacent edge in E' and adds it to E'' . The graph F induced by the edges in E'' is almost a flat forest—each tree has a height of at most two. Now we split trees of height two in F into trees of height one to produce a flat forest. All operations take constant time except the operation of finding the 3-coloring of P , which takes $O(\lg^* n)$ time. Therefore, we can find a flat forest in $O(\lg^* n)$ time on a CREW PRAM using n processors.

5. Coloring and matching in planar graphs. Euler's formula [Har72] implies that every planar graph has a constant fraction of nodes of degree six or less. In this section we use this property in conjunction with the techniques developed above to construct efficient algorithms for coloring and finding maximal matchings in planar graphs.

First we present the algorithm *7-Color-Planar-Graph* that finds a 7-coloring of a planar graph in $O(\lg n)$ time. The algorithm is shown in Fig. 3. The first stage of the algorithm partitions the nodes of the graph into sets V_i , such that the degree of any node $v \in V_i$ in $G[V_i + V_{i+1} + V_{i+2} + \dots]$ is at most six (V_i consists of all nodes of degree at most six in $G[V - (V_0 \cup V_1 \cup \dots \cup V_{i-1})]$). Then, the algorithm colors all the subgraphs induced by the node-sets $\{V_i\}$. These graphs are node-disjoint and therefore the coloring can be done in parallel. The last stage of the algorithm adds the subgraphs back in reverse order, updating the coloring.

THEOREM 5. *The algorithm 7-Color-Planar-Graph constructs a valid 7-coloring using n processors and $O(\lg n)$ time on a CRCW PRAM.*

Proof. By Euler's formula, at least a constant fraction of any planar graph's nodes are of degree six or less. Therefore, the first stage partitions the nodes of G into at most $O(\lg n)$ sets V_i . We use concurrent reads and writes to determine whether the degree of a node is at most six, and hence each iteration of the first stage is done in constant time.

Procedure 7-Color-Planar-Graph

```

 $V' \leftarrow V;$ 
 $V_1, V_2, \dots, V_{5 \lceil \lg n \rceil} \leftarrow \emptyset;$ 
 $i \leftarrow 0;$ 
while  $V' \neq \emptyset$  for all  $v \in V'$  do in parallel                                <<first stage>>
    if  $\text{Degree}(v) \leq 6$ 
        then begin
             $V_i \leftarrow V_i \cup v;$ 
             $V' \leftarrow V' - v;$ 
        end;
     $i \leftarrow i + 1;$ 
end;
 $k \leftarrow i - 1;$ 
for all  $0 \leq i \leq k$  do in parallel                                <<color the pseudoforests>>
     $E_i \leftarrow \{\{v, w\} \mid v, w \in V_i; \{v, w\} \in E\};$ 
     $C_i \leftarrow \text{Color-Constant-Degree-Graph}(V_i, E_i);$ 
end;
for  $i \leftarrow k$  down to  $0$  do                                <<second stage>>
     $V'' \leftarrow V_i;$ 
    for  $j \leftarrow 0$  to  $6$  do
        for all  $v \in V''$  do in parallel
            if  $C_v = j$ 
                then begin
                     $C_v \leftarrow \max \{ \{0, \dots, 6\} - \{C_w \mid w \in V'; \{v, w\} \in E\} \};$ 
                     $V'' \leftarrow V'' - v;$ 
                     $V' \leftarrow V' \cup v;$ 
                end;
            end;
        end;
    end;
end.

```

FIG. 3. The 7-Coloring Algorithm for planar graphs.

By Theorem 3, the second stage uses only $O(\lg^* n)$ time. In the i th iteration of the third stage, the graph $G[V_i]$ is already 7-colored and the maximum degree of each node in V_i in the graph $G[V_i + V_{i+1} + V_{i+2} + \dots]$ is at most six. Only constant time is then needed to add in V_i and produce a valid 7-coloring of $G[V_i + V_{i+1} + V_{i+2} + \dots]$. Therefore, only $O(\lg n)$ time is used in all three stages. $\square$

Remark. If at the first stage, instead of removing from the graph all the nodes with degree of at most six, we remove all nodes with degree of at most c times the average degree (for $c > 1$), the algorithm described above runs in polylogarithmic time for any graph G such that the average degree of any node-induced subgraph G' of G is polylogarithmic in the size of G' . This class contains many important subclasses including graphs that are unions of a polylogarithmic number of planar graphs (i.e., graphs with polylogarithmic thickness [Har72]).

Given a valid 7-coloring of a planar graph, we can find an MIS in the graph by iterating through colors as in our *Constant-Degree-MIS* algorithm. With concurrent reads and writes, only constant time is needed for each color class. Hence, we can find an MIS in a planar graph in $O(\lg n)$ time on a CRCW PRAM using a linear number of processors.

The deterministic parallel algorithms for 5-coloring planar graphs, described in [BK87] and in [Nao86], use $\Omega(\lg^3 n)$ time and $O(n^3)$ processors. These algorithms require a large number of processors because they use Luby's MIS algorithm [Lub86]. Using the *Constant-Degree-MIS* algorithm described in the previous section, we can reduce the number of processors to linear, while increasing the running time by a factor of $O(\lg^* n)$ [GP87a], [Gol87].

The 5-coloring algorithm presented below is essentially a parallelization of the sequential algorithms in [CNS81], [MST80]. Given an embedding (which can be computed in $O(\lg^2 n)$ time [KR86]), our algorithm runs in $O(\lg n \lg^* n)$ time on a CRCW PRAM using a linear number of processors. Given a graph $G = (V, E)$, the algorithm finds a special large independent set I of nodes in G , merges some of the neighbors of I (as described below) and removes the nodes in I to create a new graph G' , recursively colors G' , and uses this coloring to color the nodes in G .

The special independent set I is constructed as follows. Let Q be the set of all nodes in G of degree greater than 42. Let V_4 be the set of all nodes of degree four or less. Let V_5 and V_6 be the set of all nodes of degree five with at most one neighbor in Q and the set of all nodes of degree six with no neighbors in Q , respectively. Let $S = V_4 \cup V_5 \cup V_6$. Let $G^s = (S, E^s)$ be the graph induced by the nodes in S in the graph which is the square of $G[V - Q]$. The set I is a maximal independent set in the graph $G^s \cup G[V - Q]$. Since G^s and $G[V - Q]$ are of constant degree, we can find I using the procedure *Constant-Degree-MIS*.

To construct the graph G' , the algorithm proceeds as follows. Start with $G' = G$. For each node in $I \cap V_5$ we find two of its nonadjacent neighbors that have low degree (42 or less), and merge them into a single supernode. For each node in $I \cap V_6$ we either merge three of its nonadjacent neighbors into a single supernode, or merge two nonadjacent pairs of its neighbors into two supernodes. The embedding information is used as in [CNS81], [MST80] to find the neighbors that can be merged while preserving planarity after all nodes in I are removed. Then we remove all the nodes in I to get the graph G' .

After recursively 5-coloring the graph G' , we obtain the coloring of G as follows. First we color all the nodes of G that correspond to nodes or supernodes of G' with the same color they were colored in G' . Now we add all the nodes in I and in parallel color every one of them with a color different from the colors of its neighbors.

To bound the running time of the 5-coloring algorithm we need the following lemma, which is similar to Lemma 3 in [CNS81].

LEMMA 6. *The size of $S = V_4 \cup V_5 \cup V_6$ is at least a constant fraction of the total number of nodes in the graph.*

Proof. Let $R = V - S$. Denote by s_i and r_i the number of nodes of degree i in the sets S and R , respectively. Let $r_* = \sum_{i=7}^{42} r_i$, and let $r_Q = \sum_{i=43}^{\infty} r_i$. By Euler's formula, $r_Q \leq (6/43)n$.

We prove the lemma by a counting argument. Definitions of r_5 and r_6 imply that $2r_5 + r_6 \leq \sum_{i=43}^{\infty} ir_i$. Euler's formula implies that $3n \geq m$, and therefore

$$\begin{aligned}
 6n &\geq \sum_{i=1}^6 is_i + 5r_5 + 6r_6 + \sum_{i=7}^{42} ir_i + \sum_{i=43}^{\infty} ir_i \\
 &\geq \sum_{i=1}^6 is_i + 7r_5 + 7r_6 + \sum_{i=7}^{42} ir_i \\
 &\geq 7r_5 + 7r_6 + 7 \sum_{i=7}^{42} r_i \\
 &\geq 7 \left(\sum_{i=1}^6 s_i + r_5 + r_6 + r_* + r_Q \right) - 7 \sum_{i=1}^6 s_i - 7r_Q \\
 &\geq 7n - 7|S| - 7 \cdot \frac{6}{43}n.
 \end{aligned}$$

Thus, $|S| \geq n/301$. $\square$

THEOREM 7. *Given an embedded planar graph, the algorithm 5-Color-Planar-Graph 5-colors it using n processors in $O(\lg^* n \lg n)$ time on a CRCW PRAM, and $O((\lg^* n + \lg \Delta) \lg n)$ time on an EREW PRAM.*

Proof. Correctness of the algorithm follows from [CNS81] and from the fact that the nodes in I are independent in $G^S \cup G[V - Q]$.

Lemma 6 implies that the size of S is $\Omega(n)$. The graph G^S has a constant maximum degree, and hence the size of the set I is $\Omega(n)$ as well. Therefore the depth of recursion is at most $O(\lg n)$.

On a CRCW PRAM, we can find S and Q in constant time as in the algorithm 7-Color-Planar-Graph. The construction of $G^S \cup G$ takes constant time because G^S has constant degree. The algorithm Constant-Degree-MIS finds I in $O(\lg^* n)$ time. In constant time nodes in I can merge appropriate neighbors and delete themselves from G to form G' . Edge lists in G' need not be compacted when we are using the CRCW PRAM. After recursively coloring G' , we can color G in constant time.

On the EREW PRAM, $O(\lg \Delta)$ additional time per recursion level is needed since we must compact edge lists of G' (so that the set S in G' can be found in constant time). $\square$

Remark. Chrobak, Diks, and Hagerup [CDH87] have recently improved the result of Theorem 7 by giving an algorithm for 5-coloring planar graphs that runs in $O(\lg^* n \lg n)$ time on an EREW PRAM and does not need an embedding.

By using the techniques described in this section, it is easy to construct a fast algorithm for finding a maximal matching (MM) in a planar (or a constant-degree) graph. As in the 7-coloring algorithm, the first stage of the MM algorithm separates the nodes of the graph into sets V_i , such that the degree of any node $v \in V_i$ in $G[V_i + V_{i+1} + V_{i+2}, \dots]$ is at most six. Then the graphs $\{G[V_i]\}$ are colored in parallel. The second stage of the algorithm recursively finds MM in the graph $G[V - V_1]$ and removes the matched nodes to get $G[V']$, where V' is the set of the unmatched nodes. The graph $G[V']$ has no edges and the nodes V_1 in the graph $G[V' + V_1]$ have maximum degree of six. Hence, in seven iterations over the colors of $G[V_1]$ we can find the MM of G .

THEOREM 8. *A maximal matching in a planar graph can be found in $O(\lg n)$ time on a CRCW PRAM using a linear number of processors.*

Remark. Using Euler's formula, we can extend our algorithms for 7-coloring and MIS in planar graphs to graphs of bounded genus γ . We apply the algorithm 7-Color-Planar-Graph as before when there are at least $c\gamma$ nodes remaining in the residual graph, for some constant c . The Heawood map-coloring theorem states that any graph can be colored with $O(\sqrt{\gamma})$ colors, and its proof implies a polynomial time algorithm for finding such a coloring [Har72]. Therefore, when less than $c\gamma$ nodes remain in the residual graph, we sequentially color it with $O(\sqrt{\gamma})$ colors. With additional time that is polynomial in γ , we can then $O(\sqrt{\gamma})$ -color the graph using the same time and number of processors as for 7-coloring a planar graph. Note, that the above algorithm does not need embedding. The related result for MIS on bounded-genus graphs follows as before.

6. Lower bounds. In this section we prove two lower bounds for a CRCW PRAM with a polynomial number of processors:

- Finding a maximal independent set in a general graph takes $\Omega(\lg n / \lg \lg n)$ time.
- 2-coloring a directed list takes $\Omega(\lg n / \lg \lg n)$ time.

The first lower bound complements the $O(\lg n)$ CRCW PRAM upper bound for the MIS problem that is achieved by Luby's algorithm [Lub86]. The second lower bound complements Theorem 2 in this paper.

THEOREM 9. *The running time of any MIS algorithm on a CRCW PRAM with a polynomial number of processors is $\Omega(\lg n / \lg \lg n)$.*

Proof. Given an instance of MAJORITY, we construct an instance of MIS in constant CRCW PRAM time. MAJORITY is harder than PARITY [FSS81], which was proven to take $\Omega(\lg n / \lg \lg n)$ time on a CRCW PRAM in [Bea86], [BH87]. Therefore the lower bound claimed in the theorem follows.

Let $x_1, x_2, \dots, x_n$ be an instance of MAJORITY. We construct a complete bipartite graph $G = (V, E)$ with nodes corresponding to “0” bits of the input on one side and nodes corresponding to “1” bits on the other side.

$$V = \{1, \dots, n\}, \quad E = \{(i, j) \mid x_i \neq x_j\}.$$

To construct this graph, assign a processor P_{ij} for each pair $1 \leq i < j \leq n$. Then, each processor P_{ij} writes one into location M_{ij} if $x_i \neq x_j$ and writes zero otherwise.

A maximal matching in a complete bipartite graph is also a maximum one. By constructing a maximal independent set in the line-graph G' of G , we can find a maximal matching in G . To construct the graph G' assign a processor P_{ijk} for each distinct $i, j, k \leq n$. Each P_{ijk} writes one into location $M_{(i,j),(j,k)}$ if $M_{ij} = M_{jk} = 1$, and zero otherwise.

The MAJORITY equals one if and only if there is an unmatched node $i \in G$ such that $x_i = 1$, which can be checked on a CRCW PRAM in constant time. $\square$

THEOREM 10. *The time to 2-color a directed list on a CRCW PRAM with a polynomial number of processors is $\Omega(\lg n / \lg \lg n)$.*

Proof. We show a constant time reduction from PARITY to the 2-coloring of a directed list. First, we show how to construct, in constant time, a directed list with elements corresponding to all the input bits x_i with value of one. Let $x_1, x_2, \dots, x_n$ be an instance of PARITY. We can assume without loss of generality that $x_1 = 1$. With each input cell M_i (that initially holds the value of x_i), associate a processor P_i , a set of processors P_i^k with each index $i, 1 \leq k \leq j < i$, a set of cells $M_i^j, 0 \leq j < i$, and a cell M'_i . Initialize all cells that do not store input bits to zero.

In one step, each processor P_i^k reads the value of M_{i-k} and, if it equals one, writes one into M_i^j , effectively computing the OR-function on the input values $x_{i-j}, x_{i-j+1}, \dots, x_{i-1}$. Assign a processor P'_i to each $M_i^j, 1 \leq j < i$. Each processor P'_i reads M_i^j and M_i^{j+1} and writes $(i-j)$ into M'_i if and only if $M_i^j \neq M_i^{j+1}$. It can be seen that for all $1 \leq i \leq n$, M'_i holds $\max \{j \mid j < i, x_j = 1\}$.

We have constructed a directed list with elements corresponding to all the input bits x_i with value one. Assume this list is 2-colored. Then PARITY equals one if and only if both ends of the list are colored with the same color, which can be checked in constant time. $\square$

7. Conclusion and open problems. We have presented a fast technique for breaking symmetry in parallel and have shown how to apply this technique to improve the running times and processor bounds of a number of important parallel algorithms. We believe that the efficiency of this technique, combined with the simplicity of its implementation, makes it an important tool in designing parallel algorithms.

Our results motivate the following open questions.

- We have proved a lower bound for MIS in general graphs. What is the lower bound for MIS in planar graphs?
- Beame has proposed the following algorithm for coloring rooted trees of constant degree on PRAM. Run the algorithm *3-Color-Rooted-Tree* for $O(\lg \lg n)$ steps. Next, each processor collects the colors of all the descendants on distance $O(\lg n)$ or less and uses this information and a precomputed lookup table (of size $O(\lg n \lg \lg n)$) to compute its final color. Given an $\Omega(\lg n)$ preprocessing time, we can precompute the lookup table; after this preprocessing step, the

time to 3-color a tree (or a pseudoforest) will be $O(\lg \lg^* n)$. Is it possible to 3-color a tree in $o(\lg^* n)$ time on PRAM with no preprocessing?

- Can we compute an MIS in general graphs in $o(\lg n)$ time?
- Recently, several papers [CV86], [Rei85] that present parallel algorithms with “optimal speedup” have been published. (The measure of optimality used in these papers is how close is the processor-time product of the parallel algorithm to the running time of the fastest known sequential one.) The problems we have been studying in this paper can be solved in linear sequential time, but the processor-time products achieved by our algorithms are superlinear (by a $\lg^* n$ or a polylogarithmic factor). How can we reduce the processor requirements of the algorithms without increasing their running time in order to achieve linear time-processor products? Also, will this improved processor efficiency induce significantly more constraints on the model as compared to our current algorithms?

Acknowledgments. We thank Greg Frederickson, Charles Leiserson, and David Shmoys for fruitful and stimulating discussions, and for their valuable comments on a draft of this paper. We are also grateful to the referee whose extremely thorough comments have significantly improved the presentation of the paper.

REFERENCES

- [Awe85] B. AWERBUCH, *Complexity of network synchronization*, J. Assoc. Comput. Mach., 32 (1985), pp. 804–823.
- [Awe87] ———, *A tight lower bound on the time of distributed maximal independent set algorithms*, Unpublished manuscript, February 1987.
- [Bea86] P. BEAME, *Lower bounds in parallel machine computation*, Ph.D. dissertation, University of Toronto, Toronto, Ontario, Canada, 1986.
- [BH85] A. BORODIN AND J. E. HOPCROFT, *Routing, merging, and sorting on parallel models of computation*, J. Comput. System Sci., 30 (1985), pp. 130–145.
- [BH87] P. BEAME AND J. HASTAD, *Optimal bounds for decision problems on the CRCW PRAM*, in Proc. 19th Annual ACM Symposium on Theory of Computing, 1987, pp. 83–93.
- [BK87] J. BOYAR AND H. KARLOFF, *Coloring planar graphs in parallel*, J. Algorithms, 8 (1987), pp. 470–479.
- [CDH87] M. CHROBAK, K. DIKS, AND T. HAGERUP, *Parallel 5-coloring of planar graphs*, in 14th International Colloquium on Automata, Languages, and Programming, July 1987, Lecture Notes in Comput. Sci. 267, Springer-Verlag, New York, Berlin, 1987, pp. 304–313.
- [CNS81] N. CHIBA, T. NISHIZEKI, AND N. SAITO, *A linear 5-color algorithm of planar graphs*, J. Algorithms, 2 (1981), pp. 317–327.
- [Col86] R. COLE, *Parallel merge sort*, in Proc. 27th Annual IEEE Annual Symposium on Foundations of Computer Science, 1986, pp. 511–516.
- [CV86] R. COLE AND U. VISHKIN, *Deterministic coin tossing with applications to optimal parallel list ranking*, Inform. and Control, 70 (1986), pp. 32–56.
- [FSS81] M. FURST, J. SAXE, AND M. SIPSER, *Parity, circuits, and the polynomial time hierarchy*, in Proc. 22nd Annual IEEE Symposium on Foundations of Computer Science, 1981, pp. 260–270.
- [FW78] S. FORTUNE AND J. WYLLIE, *Parallelism in random access machines*, in Proc. 10th Annual ACM Symposium on Theory of Computing, 1978, pp. 114–118.
- [GHS83] R. G. GALLAGER, P. A. HUMBLET, AND P. M. SPIRA, *A distributed algorithm for minimum-weight spanning trees*, ACM Trans. Programming Languages and Systems, 5 (1983), pp. 66–77.
- [Gol87] A. V. GOLDBERG, *Efficient graph algorithms for sequential and parallel computers*, Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, January 1987.
- [GP87a] A. GOLDBERG AND S. PLOTKIN, *Efficient parallel algorithms for $(\Delta + 1)$ -coloring and maximal independent set problems*, Technical Report MIT/LCS/TM-320, Massachusetts Institute of Technology, Cambridge, MA, January 1987.
- [GP87b] ———, *Parallel $(\Delta + 1)$ coloring of constant-degree graphs*, Inform. Process. Lett., 25 (1987), pp. 241–245.

- [GS87] M. GOLDBERG AND T. SPENCER, *A new parallel algorithm for the maximal independent set problem*, in Proc. 28th Annual IEEE Symposium on Foundations of Computer Science, 1987, to appear.
- [Har72] F. HARARY, *Graph Theory*, Addison-Wesley, Reading, MA, 1972.
- [IS86] A. ISRAELI AND Y. SHILOACH, *An improved parallel algorithm for maximal matching*, Inform. Process. Lett., 22 (1986), pp. 57–60.
- [KR86] P. KLEIN AND J. REIF, *An efficient parallel algorithm for planarity*, in Proc. 27th Annual IEEE Symposium on Foundations of Computer Science, 1986, pp. 465–477.
- [KW85] R. M. KARP AND A. WIGDERSON, *A fast parallel algorithm for the maximal independent set problem*, J. Assoc. Comput. Mach., 32 (1985), pp. 762–773.
- [Lin87] N. LINIAL, *Locality as an obstacle to distributed computing*, in Proc. 28th Annual IEEE Symposium on Foundations of Computer Science, October 1987, pp. 331–335.
- [LM86] C. LEISERSON AND B. MAGGS, *Communication-efficient parallel graph algorithms*, in Proc. International Conference on Parallel Processing, 1986, pp. 861–868.
- [Lub86] M. LUBY, *A simple parallel algorithm for the maximal independent set problem*, SIAM J. Comput., 15 (1986), pp. 1036–1052.
- [MR85] G. MILLER AND J. REIF, *Parallel tree contraction and its application*, in Proc. 26th Annual IEEE Symposium on Foundations of Computer Science, October 1985, pp. 478–489.
- [MST80] D. MATULA, Y. SHILOACH, AND R. TARJAN, *Two linear-time algorithms for five-coloring a planar graph*, Technical Report STAN-CS-80-830, Department of Computer Science, Stanford University, Palo Alto, CA, November 1980.
- [Nao86] J. NAOR, *Two parallel algorithms in graph theory*, Technical Report CS-86-6, Department of Computer Science, The Hebrew University of Jerusalem, Jerusalem, Israel, June 1986.
- [PQ82] J. C. PICARD AND M. QUEYRANNE, *A network flow solution to some nonlinear 0-1 programming problems, with applications to graph theory*, Networks, 12 (1982), pp. 141–159.
- [Rei85] J. REIF, *An optimal parallel algorithm for integer sorting*, in Proc. 26th Annual IEEE Symposium on Foundations of Computer Science, October 1985, pp. 496–503.
- [Sha86] G. SHANNON, *Parallel independent set algorithms for sparse graphs*, Technical Report CSD-TR-634, Computer Sciences Department, Purdue University, West Lafayette, IN, October 1986.

RECOGNIZING BELLMAN-FORD-ORDERABLE GRAPHS*

RAMSEY W. HADDAD† AND ALEJANDRO A. SCHÄFFER‡

Abstract. Mehlhorn and Schmidt [*Discrete Appl. Math.*, 15 (1986), pp. 315–327] consider the following problem. Given a directed graph with distinguished source vertex s , is it possible to order the edges so that all simple paths starting at s use edges in increasing order? They show how to solve their problem in $O(|E|^2)$ steps, where $|E|$ is the number of edges. An algorithm that runs in $O(|V|^2)$ steps is given, where $|V|$ is the number of vertices. The new algorithm and its analysis apply and extend previous results on dominators in directed graphs.

Key words. dominators, directed graphs, shortest paths, Bellman–Ford algorithm

AMS(MOS) subject classifications. primary 68R10; secondary 05C20, 05C38, 05C75, 68Q25.

1. Introduction. Let $G = (V, E)$ be a directed graph with one distinguished source vertex s . Mehlhorn and Schmidt [Meh86] define G to be *Bellman–Ford-orderable* or *BF-orderable* if the edges can be labeled with distinct natural numbers, so that on any (vertex) simple path emanating from s , the edge labels are ascending. In such a graph, the sequence of $|E|$ edges sorted by label contains every simple path from s as a subsequence. Therefore the Bellman–Ford algorithms for problems such as single-source shortest path in a graph with no negative weight cycles run in time $O(|E|)$ instead of the bound of $O(|V||E|)$ for general graphs [Bel58], [Law76]. Directed acyclic graphs are an important special class of BF-orderable graphs, but as shown by the example of Fig. 1, a BF-orderable graph may have cycles.

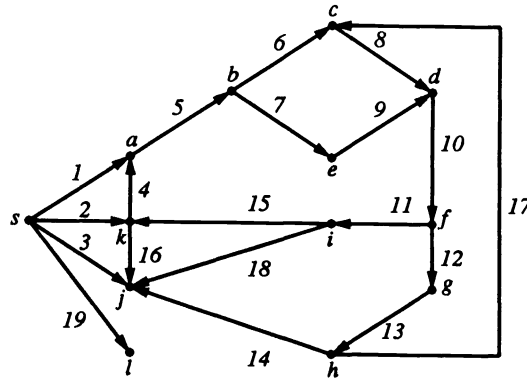
If we want to solve an instance of the single-source shortest path problem where all edge weights are positive, then Dijkstra's algorithm [Dij59], [Law76] can be implemented in time $O(|E| + |V| \log |V|)$ using a Fibonacci heap [Fre84] or a relaxed heap [Dri87]. If there are negative edge weights, however, Dijkstra's algorithm can be modified to work correctly, but it may take exponential time in the worst case even if the graph is acyclic [Joh73], [Shi81].

As motivation for the problem of recognizing BF-orderable graphs and labeling their edges, Mehlhorn and Schmidt cite two applications to VLSI problems [Lei83], [Len86]. For example, Leiserson and Pinter [Lei83] reduce a class of river routing problems to solving single-source longest paths on certain graphs with positive *and* negative edge weights having no positive weight cycles. In the same paper Leiserson and Pinter show not only that all the graphs that their reduction generates are BF-orderable, but also that the structure of their graphs is special enough that they can compute an edge ordering in time $O(|E|)$. Mehlhorn and Schmidt show how to decide in time $O(|E|^2)$ whether an arbitrary digraph with fixed source is BF-orderable and how to compute a labeling if it is. Unfortunately this time bound is somewhat impractical, unless we want to run Bellman–Ford algorithms on the same graph many times, because just copying the list

* Received by the editors August 20, 1987; accepted for publication (in revised form) May 5, 1988.

† Computer Science Department, Stanford University, Stanford, California 94305-2140. The work of this author was supported by National Science Foundation grants DCR 83-08109 and CCR 86-10181.

‡ Computer Science Department, Stanford University, Stanford, California 94305-2140. The work of this author was primarily supported by a Fannie and John Hertz Foundation Fellowship, and in part by National Science Foundation grants DCR 83-20000 and CCR 87-04170. Part of this work was done while this author was a Research Student Associate at IBM Almaden Research Center. Present address, Computer Science Department, Rice University, Houston, Texas 77251.

FIG. 1. A Bellman-Ford-orderable graph G with cycles.

of edges in arbitrary order $|V|$ times produces a list of length only $O(|V||E|)$ that contains all simple paths emanating from s as subsequences.

We show how to recognize BF-orderable graphs and to compute a labeling in time $O(|V|^2)$ improving on the $O(|E|^2)$ bound of Mehlhorn and Schmidt. We use ideas from the theory of dominators in digraphs, which was originally developed to solve problems in the design of optimizing compilers [Pro59], [Low69], [Aho79], in order to decompose the original problem into smaller recognition problems. If the graph is BF-orderable then we translate the labelings on the smaller graphs we derive into a labeling of the entire graph. The hard part is to show that the number of edges in each of the smaller graphs we work on is linear in the number of vertices of that graph. This implies that we can use the Mehlhorn-Schmidt algorithm on the smaller graphs and get the improved bound.

2. Preliminaries. We call any sequence of vertices $W = v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_r$, with an edge between each pair of consecutive vertices, a *walk*. If all of the vertices in W are distinct, then we call W a (simple) *path*, and we call vertices $v_2, \dots, v_{r-1}$ *internal vertices*. If all of the vertices on W are distinct, except that v_1 and v_r coincide, then we call W a (simple) *cycle*. We use the notation $G - v_1 - V_1 - v_2 - V_2 \dots$ to mean the graph induced by $V \setminus (\{v_1, v_2, \dots\} \cup V_1 \cup V_2 \dots)$, where $v_1, v_2, \dots$ represent individual vertices and $V_1, V_2, \dots$ represent either arbitrary sets of vertices or the vertices on a walk; the symbols for individual vertices and sets may be interspersed in any order in such expressions. If $v_1, v_2 \in V$, we use the notation $v_1 \xrightarrow{*} v_2$ for a (simple) path from v_1 to v_2 , where the path contains no edges if the two vertices coincide. In cases where the two endpoints of a path cannot coincide we use the notation $v_1 \xrightarrow{+} v_2$. If W is a walk in G we use V_W to mean the vertices occurring on W ; generally W will be a path or a cycle.

2.1. Previous work. On the way to their $O(|E|^2)$ algorithm, Mehlhorn and Schmidt develop some useful definitions and alternate characterizations of BF-orderable graphs that we repeat here.

DEFINITION 2.1. If $U \subseteq V$ is a set of vertices then $w \in U$ is an *entry point* of U if there is a path from the source to w that does not go through any other vertex of U . The entry points to $G - s$ will be referred to as *contact points*.

DEFINITION 2.2. A graph has *property BF1* if every simple cycle C has at most two entry points.

DEFINITION 2.3. A graph that satisfies BF1 is said to satisfy *property* BF2 if for every simple cycle C with two entry points the two entry points of C are neighbors on C .

DEFINITION 2.4. A graph that satisfies BF1 and BF2 is said to satisfy *property* BF3 if there is no pair of simple cycles C_1 and C_2 such that we have the following:

- C_1 has two entry points u_1 and v_1 and includes the edge $u_1 \rightarrow v_1$; C_2 has two entry points u_2 and v_2 and includes the edge $u_2 \rightarrow v_2$.
- C_1 and C_2 have no common vertices except for the possibility that $u_1 = u_2$.
- The graph contains a path $P = u_1 \rightarrow w_1 \rightarrow w_2 \rightarrow \dots \rightarrow w_k \rightarrow u_2$ and the antiparallel path $P^{-1} = u_2 \rightarrow w_k \rightarrow \dots \rightarrow w_2 \rightarrow w_1 \rightarrow u_1$, traversing precisely the same vertices in reverse order. The paths P and P^{-1} have no vertices in common with the two cycles C_1 and C_2 except their endpoints u_1 and u_2 .
- v_1 and v_2 are entry points of the set $V(C_1) \cup V(C_2) \cup V(P)$.

DEFINITION 2.5. A graph has *property* BF4 if there is no pair e, e' of edges such that there are (simple) paths from s : $P = P_1 e P_2 e'$ and $Q = Q_1 e' Q_2 e$. (See Fig. 2 for a sketch of a violation of BF4.)

Two results of Mehlhorn and Schmidt [Meh86] are the following lemmas.

LEMMA 2.6. The graph G is BF-orderable if and only if G satisfies BF1, BF2, and BF3. $\square$

LEMMA 2.7. The graph G is BF-orderable if and only if G satisfies BF1 and BF4. $\square$

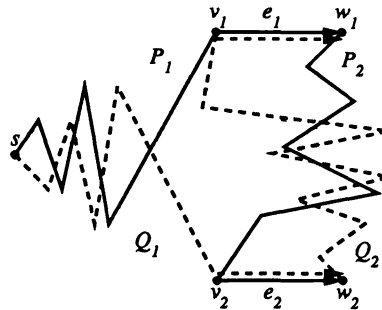


FIG. 2. Sketch of a violation of BF4; P is solid, Q is dashed.

2.2. Mehlhorn and Schmidt's algorithm. Mehlhorn and Schmidt's algorithm to recognize and label a BF-orderable graph works as follows.

(1) Build an auxiliary graph having one vertex for each edge of G . Put an edge in the auxiliary graph from $e_i = u \rightarrow v$ to $e_j = v \rightarrow w$ if there is a path in G from s that ends $e_i e_j$.

The edges of the auxiliary graph can be computed as follows. For each edge $e = v \rightarrow w$ of G do a depth first search of G starting at s , but prohibiting the use of any edge from v , from w , or to w . Then for each edge $e' = u \rightarrow v$ such that vertex u is reached in the depth first search, add the edge $e' \rightarrow e$ to the auxiliary graph. Each depth first search and the subsequent addition of edges to the auxiliary graph takes $O(|E|)$ time. There are $O(|E|)$ searches to do, so the cost of this step is $O(|E|^2)$.

(2) Attempt to compute a topological numbering of the vertices of the auxiliary graph. If there is a cycle in the auxiliary graph then the original graph is not BF-orderable.

able; otherwise, any complete topological ordering is a valid BF-ordering of the edges of G . While Mehlhorn and Schmidt claim only an $O(|E|^2)$ time bound for this step, the auxiliary graph has at most $O(|V||E|)$ edges, so this step really takes only $O(|V||E|)$ time.

2.3. Dominators.

DEFINITION 2.8. In a directed graph with source vertex s , a vertex v *dominates* a vertex $w \neq v$ if every path from s to w passes through v . Furthermore, v is said to be the *immediate dominator* of w if there is no vertex x such that v dominates x and x dominates w .

It can be shown [Aho79], [Tar74] that the immediate dominator relation forms an outtree, with s as the source. This tree is called the *dominator tree* of G . If v is the immediate dominator of w in G , then in the dominator tree v is the parent of w . In the graph G , a vertex v dominates a vertex w if and only if v is an ancestor of w in the dominator tree of G . It will turn out to be useful to compute the dominator tree of the input graph.

We can find the dominator tree in time $O(|E| + |V| \log |V|)$ using the algorithm of Tarjan [Tar74] or we can find it in time $O(|E|\alpha(|E|, |V|))$, where α is the very slowly growing functional inverse of Ackermann's function, using the refinement of Lengauer and Tarjan [Len79]. Experimental data [Len79] suggest that the refined version is quite efficient in practice. But, for the purpose of asymptotic bounds, the $O(|E|)$ time algorithm of Harel [Har85] is best.

3. Outline of algorithm. Our $O(|V|^2)$ algorithm to recognize and label BF-orderable graphs, when given a directed graph G with distinguished vertex s , can be summarized as follows.

(1) Find all self-loops, all edges to s , and all edges from vertices unreachable from s . These edges can be given arbitrary labels, so give them distinct labels larger than the number of other edges. From now on we assume that these edges are deleted.

(2) Find the dominator tree T of G with respect to s . See Fig. 3 for an example.

(3) Construct G' from G as follows. For each vertex v that has dominator tree descendants, create a vertex v' and add the edge $v \rightarrow v'$. For every back edge $u \rightarrow v \in E$ —an edge such that v is closer to s in the dominator tree than u is—let w be the unique ancestor of u in T that is a sibling of v and replace $u \rightarrow v$ with an edge $w' \rightarrow v$. If $w = v$, then the back edge $u \rightarrow v$ cannot occur on any path starting at s , so we may label it immediately instead of doing the replacement. We assume below that such edges are labeled immediately. If there are multiple copies of any $w' - v$ edge, collapse them into one copy—saving the information to allow a later reversal of the collapse. See Fig. 4 for an example. Tarjan [Tar81] uses a similar transformation to solve a different problem related to paths in directed graphs, although in his problem he does not introduce the vertex v' and edge $v \rightarrow v'$.

(4) Construct G'' from G' as follows. For each vertex v that has descendants in the dominator tree T , create a vertex v'' . Replace every edge $v \rightarrow w$ of G , where w is a child of v in the dominator tree, with the edge $v'' \rightarrow w$. Thus G'' consists of a disconnected component G''_v for each $v \in V(G)$ that is not a leaf of the dominator tree, plus an extra *root component* consisting of s and s' . See Fig. 5 for an example.

(5) Construct H from G'' as follows. Add a vertex s_0 and replace the edge $s \rightarrow s'$ with the edge $s_0 \rightarrow s'$. Call the graph induced by $\{s_0, s'\}$, H_0 . For each G''_v , break

$G'' - v''$ into its strongly connected components—remembering their topological ordering. For each strongly connected component H_j , create a new source vertex s_j . For each edge in G'' to vertex w of H_j that either goes between two strong components of the same G'' or is the edge $v'' \rightarrow w$, replace such an edge with $s_j \rightarrow w$. If there are multiple copies of any such $s_j \rightarrow w$ edge, collapse them into one copy—saving the information to allow a later reversal of the collapse. See Fig. 6 for a sample H .

(6) For each H_j , if it is BF-orderable then the size of $E(H_j)$ is linear in the number of vertices in $V(H_j)$ (Corollary 7.22). If any H_j has more than $8|V(H_j)| - 12$ edges then stop now, as G being BF-orderable implies that each H_j is BF-orderable (Theorem 4.2).

(7) Use the original [Meh86] algorithm to locally label each H_j . Their algorithm runs in time $O(|E|^2)$. Since each vertex $v \in G$ spawns at most one v' in H , and since each source vertex s_j is in the same component as at least one v or v' , $\sum_j |V(H_j)| \leq 4|V|$. Since each H_j has $|E(H_j)| \leq 8|V(H_j)| - 12$, the time for this step is $O(|V|^2)$.

(8) Combine the separate labelings and simultaneously label the previously unlabeled edges of the original graph.

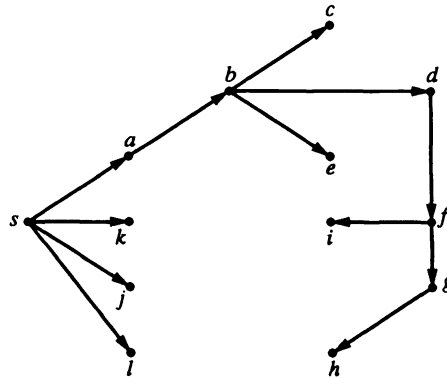


FIG. 3. The dominator tree for G in Fig. 1.

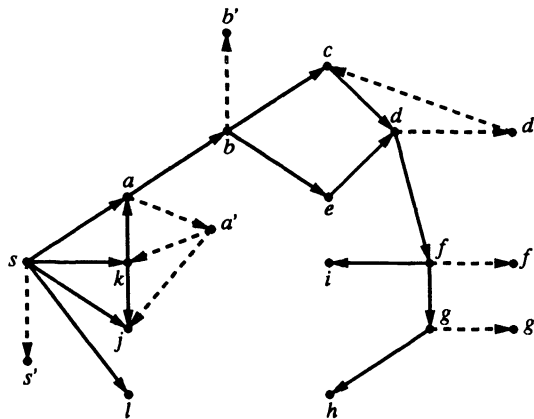


FIG. 4. The graph G' corresponding to the graph G in Fig. 1. Old edges are solid; new edges are dashed.

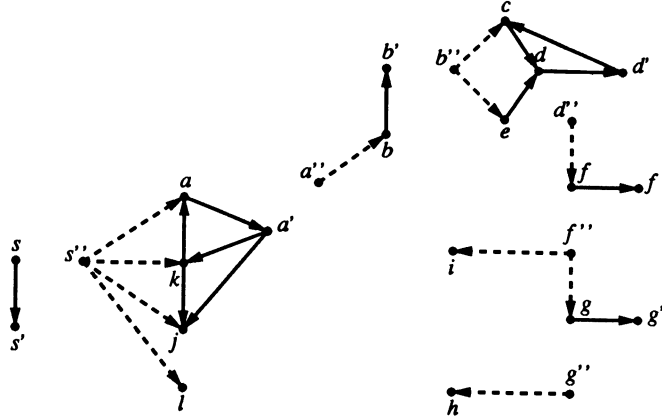


FIG. 5. The graph G'' corresponding to G' in Fig. 4. Old edges are solid; new edges are dashed.

4. Details, time analysis, and algorithm correctness. In this section we fill in more details of the algorithm and prove it correct.

4.1. Details for step 8. The focus of this section is on how to translate the *local* labelings of the H_j 's into a *global* labeling of G . It is easiest to think of the translation as a pair of mutually recursive procedures. They use a global variable L that represents the highest label used so far in G ; initially $L = 0$. In order to explain the two procedures, we use the following definition.

DEFINITION 4.1. The *expansion* of an edge in H_j is:

- $s_j \rightarrow v$: the edges that this edge represents in the appropriate G'' .
- $u \rightarrow v$: the same *base edge* $u \rightarrow v$.
- $v \rightarrow v'$: the edges of G''_v .
- $u' \rightarrow v$: the back edges that this edge represents.
- $u'' \rightarrow v$: the edge $u \rightarrow v$.

If E' is a set of edges, we can apply the appropriate expansion rule to some $e \in E'$, and obtain a new set E'' where e is replaced by its expansion. The *complete recursive expansion* of a set of edges is the set of base edges we obtain by successively expanding members of the current set and replacing the current set with the new set until only base edges remain.

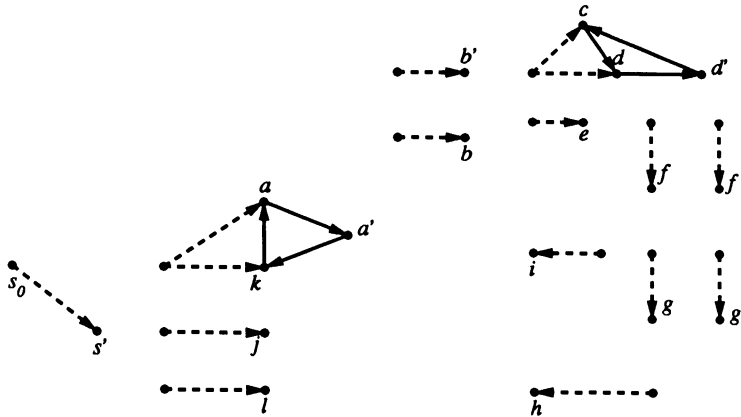


FIG. 6. The graph H corresponding to G'' in Fig. 5. Old edges are solid; new edges are dashed. New source vertices are unnamed.

The main recursive procedure, which we call **label_component**, is given an H_j to label. Using the other procedure, which we call **label_edge**, **label_component** gives global labels to every edge of G in the complete recursive expansion of $E(H_j)$. To bootstrap call **label_component** on H_0 (defined in Step (5) of the main algorithm).

procedure **label_component**:

Iterate through the edges of the H_j in the order of the local labeling, calling **label_edge** on each edge to give it a global label.

procedure **label_edge**: there are five mutually exclusive cases:

(1) If the current edge is from an s_j —the source of an H_j —then it corresponds to a set of edges of G'' . Iteratively call **label_edge** on each of these edges in any order; they may be any combination of case 2, 3, 4, or 5 edges. (Each edge in the corresponding set either has a source of the type v'' , or is between strongly connected components in the same connected component of G'' , or is the edge $s \rightarrow s'$.)

(2) If the current edge connects two vertices of the original G , increment L and then give the current edge the global label L .

(3) If the current edge goes from a vertex $u \in G$ to u' —one of the vertices added in going from G to G' in Step (3) of the main algorithm—then recursively label the edges in the dominator tree's subtree rooted at u as follows. Iterate through the H_j 's corresponding to G'' in topological order, calling **label_component** on each of them.

(4) If the current edge goes from a u' to w , a vertex of G , then it corresponds to a set of edges of G . Label those edges consecutively starting with $L + 1$, incrementing L to be the highest label used. (These edges of G are back edges with respect to the dominator tree.)

(5) If the current edge goes from a u'' to any other vertex, find the edge e in G that it replaced, increment L , and give e the global label L .

Observe that the five rules in the definition of *expansion* are mutually exclusive and that they are exactly analogous to the five cases outlined in the description of **label_edge**.

4.2. Time analysis of algorithm. As mentioned in § 2.3, we can perform Step (2) in $O(|E|)$ time.

In Step (3) we keep only one copy of each $v' \rightarrow w$ edge in G' , but we need to compute and remember all the edges in G that were mapped to each $v' \rightarrow w$ edge in G' , so that we can easily translate the local edge labels for the H_j into global edge labels for G . This replacement of the back edges can be done in time $O(|E|)$ as outlined in § 8.

Steps (4) and (6) do a constant amount of work per vertex, and hence require only $O(|V|)$ time.

For Step (5) we need only $O(|E|)$ time to compute the strongly connected components [Tar72] and to perform the topological ordering [Knu73].

The routines **label_component** and **label_edge** manipulate each vertex and edge in all the graphs we create a constant number of times, so Step (8) takes $O(|E|)$ time.

The only remaining task in the time analysis is to show that Step (7) really takes only $O(|V(H_j)|^2)$ time for each H_j where the constant implicit in the big O can be chosen independent of H_j . That is, we must still show that the H_j 's have at most a linear number of edges. We will do this in §§ 5–7.

4.3. Correctness. We now prove that the composition of all our transformations preserves BF-orderability.

THEOREM 4.2. *The graph G is BF-orderable if and only if every H_j is BF-orderable.* $\square$

Proof. By Lemmas 2.7, 4.4 (below), and 4.5 (below) we know that if G is BF-orderable then so are all of the H_j 's. Theorem 4.7 (below) gives us the implication in the opposite direction by showing the correctness of our method to label G from labels for the H_j 's. $\square$

LEMMA 4.3. *Let $v \in V(G)$ also be in $V(H_j)$, and suppose that $s_j \rightarrow v \in E(H_j)$. There is a (simple) path $s \xrightarrow{+} v$ in G that contains no internal vertex that is in $V(H_j)$ or is dominated by a vertex in $V(H_j)$.*

Proof. Note that $v \neq s$, because s does not appear in H . Let x be the parent of v in the dominator tree. There is a path $P_x = s \xrightarrow{*} x$ in G that uses no vertices dominated by x . If G contains an edge $x \rightarrow v$, then append that edge to P_x to get the desired path. If G does not contain the edge $x \rightarrow v$, then the edge $s_j \rightarrow v$ appears in H_j because in step (5), the graph G_x'' has more than one strongly connected component. In this case there is a path in G_x'' from the source x'' to v that uses no internal vertices from the strongly connected component containing v . Therefore, we can find a corresponding path, $P_v = x \xrightarrow{*} v$, in G that uses no internal vertices belonging to H_j (or dominated by any vertex in H_j). The desired $s \xrightarrow{+} v$ path is $P_x P_v$. $\square$

LEMMA 4.4. *If the graph G satisfies BF1 then every H_j satisfies BF1.*

Proof. We prove the contrapositive. Let C be a cycle in some H_j that has more than two entry points, thus violating BF1. Let its vertices be $v_1, v_2, \dots, v_k$ in that order and let its entry points be $w_1, w_2, \dots$, a subsequence of the vertex sequence. All the entry points must be original vertices of G , as the v' vertices have indegree one, the v'' vertices have been deleted in going from G'' to H_j , and the s_j vertices are the sources and hence are not in the cycle. We will transform C into a cycle C' in G that has more than two entry points. If v_i and v_{i+1} (or v_k and v_1) are original vertices of G then the edge between them is also in $E(G)$ and we leave it in C' . If the cycle contains a two-edge subpath of the form $v_i \rightarrow v'_i \rightarrow v_{i+2}$ there must be a back edge $u_i \rightarrow v_{i+2}$ in G . Replace the two-edge path in H_j with a path $v_i \xrightarrow{+} u_i$ in G that uses only vertices dominated by v_i , followed by the edge $u_i \rightarrow v_{i+2}$.

It is important to observe in the back edge case that because of the dominator tree definition, the path $v_i \xrightarrow{+} v_{i+2}$ in G will include other vertices dominated by v_i , but none of the other two-edge paths of the form $v_l \rightarrow v'_l \rightarrow v_{l+2}$ in C could use vertices from that same subtree when they are replaced. Thus, after all the necessary two-edge paths of H_j are replaced by paths in G , we get a simple cycle C' in G .

For each entry point w_i , there is a path $s_j \xrightarrow{+} w_i$ in H_j not using any other vertex of C . If the path consists of just the edge $s_j \rightarrow w_i$, then Lemma 4.3 shows that there is a $s_j \xrightarrow{+} w_i$ path in G that establishes w_i as an entry point of C' . Otherwise, the path in H_j consists of an edge $s_j \rightarrow y_i$ followed by a path $P = y_i \xrightarrow{+} w_i$. By Lemma 4.3 we can find a path $s \xrightarrow{+} y_i$ in G that contains no other vertex of H_j . Then we can construct a path $y_i \xrightarrow{+} w_i$ in G that proves that w_i is also an entry point of C' ; this path is constructed from P just as we constructed C' from C , replacing one or two edges at a time. $\square$

LEMMA 4.5. *If the graph G satisfies BF4 then every H_j satisfies BF4.*

Proof. We again prove the contrapositive using a similar approach. Suppose BF4 is violated in some H_j with P and Q as the offending paths and e_1 and e_2 as the offending edges in H_j . We construct corresponding walks P' and Q' in G and show that they are paths. Let $s_j \rightarrow v$ be the first edge of path P , and let $s_j \rightarrow w$ be the first edge of path Q ; neither of these edges is e_1 or e_2 . By Lemma 4.3 we can find paths $P_v = s \xrightarrow{+} v$ and $Q_w = s \xrightarrow{+} w$ in G that use no internal vertex belonging to $V(H_j)$. The path P' will start with P_v ; the path Q' will start with Q_w .

Wherever P contains an edge $y \rightarrow z$ belonging to G we leave that edge in P' , and similarly for Q and Q' . Wherever P contains a two-edge subpath of the form $y \rightarrow y' \rightarrow z$, we find a path $P'_{yz} = y \xrightarrow{+} z$ in G using as internal vertices only vertices dominated by y , and then in both P' and Q' we replace $y \rightarrow y' \rightarrow z$ with the path P'_{yz} . Finally we may have to replace an edge of the form $y \rightarrow y'$ at the end of P or Q . Suppose this edge occurs at the end of P ; this means that it is the offending edge e_2 , so it also occurs in the middle of Q as the first edge in a two-edge subpath $y \rightarrow y' \rightarrow z$, and in Q' we replaced this two-edge path with some path P'_{yz} . Then we should make the last edge of P' be the first edge of the path P'_{yz} .

For any $y \in V(H_j) \cap V(G)$, y is not used as an internal vertex in P_v or in Q_w . The vertex y' occurs at most once in P and at most once in Q ; therefore in P' and in Q' vertices of G from the subtree of T rooted at y can only occur once. Thus the walks P' and Q' are paths.

To prove that P' and Q' violate BF4 we need to show that the last edge of P' occurs in Q' (not as its last edge), and it will follow by symmetry that the last edge of Q' occurs in P . If e_2 is in G then it is also in P' and in Q' , and it occurs somewhere in the middle of Q' . If e_2 is of the form $y' \rightarrow z$, then both P and Q contain the two-edge subpath $y \rightarrow y' \rightarrow z$, and it does not occur at the end of Q ; in both P' and Q' we replaced this two-edge path with the same path P'_{yz} , so the back edge at the end of this path occurs in the middle of Q' and at the end of P' . Finally if e_2 is of the form $y \rightarrow y'$, we explained two paragraphs above how to ensure that its replacement in P' is also contained in Q' . $\square$

LEMMA 4.6. *Let G be a BF-orderable graph. Let there be a path from s with $e_1 = v_1 \rightarrow v_2$ and $e_2 = v_2 \rightarrow v_3$ as the last two edges. If there is a G''_v with edges $f_1 = w_1 \rightarrow w_2$ and $f_2 = w_2 \rightarrow w_3$ the complete recursive expansions of which include e_1 and e_2 , respectively, and such that there is a path from v'' with f_1 and f_2 as the last two edges, then running **label_component** on any given set of valid local BF-labelings on the H_j 's will give e_1 a smaller global label than e_2 .*

Proof. In going from G''_v to the H_j 's, the edges f_1 and f_2 get mapped to the edges g_1 and g_2 , respectively.

Case 1. The edges g_1 and g_2 are in the same H_j . In this case, there is a path from s_j to g_1 and then g_2 . Hence g_1 must get a lower local label than g_2 in any valid labeling of H_j . Since the complete recursive expansion of g_1 includes e_1 and the complete recursive expansion of g_2 includes e_2 , e_1 will have a lower global label than e_2 .

Case 2. The edges g_1 and g_2 are in different H_j 's. Say that g_1 is an H_i and g_2 is in H_k . It must be the case that H_i precedes H_k in any valid topological sorting of the H_j 's corresponding to G''_v . Hence, g_1 will be given a lower global label than g_2 . $\square$

All edges in G receive a global label, since during the global labeling we uncollapse all collapsed edges. These labels are distinct because we always increment L . The following theorem shows that the global labeling for G is valid.

THEOREM 4.7. *After we run our algorithm on the H_j 's to label G , every path from s in G has ascending global edge labels.*

Proof. Suppose $e_1 = v_1 \rightarrow v_2$, $e_2 = v_2 \rightarrow v_3$, and there is a path from s with e_1 and e_2 as its last two edges. We split into three cases depending on the type of edge e_1 , and split each of those into subcases depending on the type of edge e_2 . In each case we identify appropriate edges f_1 and f_2 , and an appropriate component G''_v , and then appeal to Lemma 4.6.

(1) e_1 is an edge of the dominator tree T .

(a) e_2 is a tree edge. Let $f_1 = v'_1 \rightarrow v_2$ and $f_2 = v_2 \rightarrow v'_2$ in G''_{v_1} .

(b) e_2 connects siblings of T . Let $f_1 = v''_1 \rightarrow v_2$ and $f_2 = v_2 \rightarrow v_3$ in G''_{v_1} .

(c) e_2 is a back edge. Let w_1 be the sibling of v_3 in the dominator tree that is

an ancestor of v_2 . (If $v_3 = w_1$ then there cannot be a path from s ending with $e_1 e_2$.) Let w_2 be the least common ancestor of v_3 and w_1 . Let $f_1 = w_1 \rightarrow w'_1$ and $f_2 = w'_1 \rightarrow v_3$ in G''_{w_2} .

- (2) e_1 connects siblings of T .
 - (a) e_2 is a tree edge. Let v be the parent of v_1 and v_2 . Let $f_1 = v_1 \rightarrow v_2$ and $f_2 = v_2 \rightarrow v'_2$ in G''_v .
 - (b) e_2 connects siblings. Let $f_1 = v_1 \rightarrow v_2$ and $f_2 = v_2 \rightarrow v_2$ in G''_v , where v is the parent of v_1 (and v_2 and v_3).
 - (c) e_2 is a back edge. Let w_1 be the sibling of v_3 in the dominator tree that is an ancestor of v_1 . Let w_2 be the least common ancestor of v_3 and v_1 ; w_2 is also the parent of v_3 and w_1 . Let $f_1 = w_1 \rightarrow w'_1$ and $f_2 = w'_1 \rightarrow v_3$ in G''_{w_2} .
- (3) e_1 is a back edge.
 - (a) e_2 is a tree edge. Let w_1 be the sibling of v_2 that dominates v_1 . Let $f_1 = w'_1 \rightarrow v_2$ and $f_2 = v_2 \rightarrow v'_2$ in G''_v , where v is the parent of v_2 .
 - (b) e_2 connects siblings. Let w_1 be the sibling of v_2 that dominates v_1 . Let $f_1 = w'_1 \rightarrow v_2$ and $f_2 = v_2 \rightarrow v_3$ in G''_v , where v is the parent of v_2 .
 - (c) e_2 is a back edge. Let w_1 be the sibling of v_3 in the dominator tree that is an ancestor of v_1 . Let w_2 be the least common ancestor of v_3 and v_1 . Let $f_1 = w_1 \rightarrow w'_1$ and $f_2 = w'_1 \rightarrow v_3$ in G''_{w_2} . $\square$

5. Flat dominator trees. The task that remains is to show that any H_j component that is BF-orderable has less than $8|V(H_j)|$ edges. This will be done in the next three sections. We apologize to the reader for the length of the proof. We were unable to find a simpler proof, which could yield more insight into the structure of the H_j components. Furthermore, since the labeling of these components is the only step that takes quadratic time, a different proof could lead to a new and faster BF-ordering algorithm.

DEFINITION 5.1. If the dominator tree for a graph G is of depth one, then we say G has a *flat* dominator tree.

DEFINITION 5.2. If a graph, minus its source vertex, is strongly connected, we say the graph is *strongly connected with an external source*.

DEFINITION 5.3. If G is strongly connected with an external source and has a flat dominator tree, we say G is *porous*. If G is porous and BF-orderable, we say G is *BF-porous*. We call these graphs porous because intuitively they should have many vertex-disjoint paths, and hence would tolerate a large flow from the source to any vertex.

For each H_j , create a graph H'_j as follows. Obtain each H'_j from the corresponding H_j by collapsing each pair of vertices v and v' together into the vertex v , if they both are in H_j . Our goal is to bound the *density*, that is the ratio of edges to vertices, of the graphs H_j under the assumption that they are all BF-orderable. We do this by relating the density of an H_j to the density of its corresponding H'_j , by showing that each graph H'_j with more than two vertices is BF-porous if the corresponding graph H_j is BF-orderable, and then by bounding the density of all BF-porous graphs.

There are two ways of decomposing a BF-ordering problem into independent sub-problems. One way is to break up the graph using dominators, and the other way is to break it up using strongly connected components. Clearly, after applying one of these decompositions nothing is gained by applying it again immediately. Indeed, it turns out that after applying the dominator decomposition followed by the strongly connected component decomposition, nothing is gained by applying the dominator decomposition again. Hence, we cannot just keep decomposing the problem until we have only trivially solvable (constant time) problems. But it turns out that these decompositions yield graphs that are easier than normal to label, since they have very few edges.

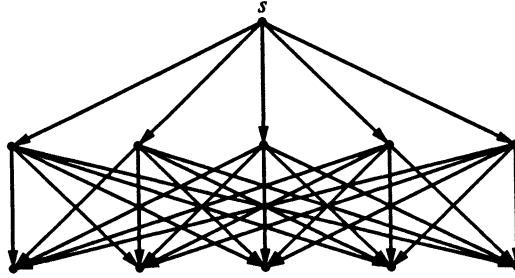


FIG. 7. A Bellman-Ford-orderable graph with high density.

Is this surprising? Yes and no. It seems that BF-orderable graphs should not be able to have many edges because then there would be cycles with many entry points. This intuition fails because of two constructs allowable in BF-orderable graphs that can have lots of edges. These are shown in Figs. 7 and 8. Both can have quadratically many edges. However, BF-porous graphs cannot include these constructs. The graph in Fig. 7 is not strongly connected and the graph in Fig. 8 does not have a flat dominator tree.

LEMMA 5.4. *Let H_j have e edges and n vertices. Let H'_j have e' edges and n' vertices. Then, $e' \geq (e - (n - n'))/2$.*

Proof. In transforming H_j to H'_j , we delete all the v' vertices, of which there are $n - n'$. The edges of the form $v \rightarrow v'$ are deleted, leaving $e - (n - n')$ edges. Then, the edges of the form $v' \rightarrow w$ get transformed to edges of the form $v \rightarrow w$. If the edge $v \rightarrow w$ is already present in the graph, we keep only one copy. This explains the division by two. $\square$

LEMMA 5.5. *Each H'_j has a flat dominator tree.*

Proof. If there is a v' vertex in H'_j , then the corresponding v vertex was not in the graph H_j from which we obtained H'_j . Since H_j was strongly connected with external source, and only v or s_j could have an edge to v' , it follows that H_j and, hence H'_j , consist solely of a source and v' . Hence the dominator tree is flat.

We can now assume that H'_j has no v' vertices. Let s_j be the source of H'_j . Let two distinct vertices in H'_j $v \neq s_j$ and $w \neq s_j$ be given. By construction of H_j , v and w are siblings in T , so there is a path P from s to w in $G' - v$. Let x be the first vertex on P that is in H'_j ; note that $x \neq v$, but they are siblings in T . After reaching x , P never leaves

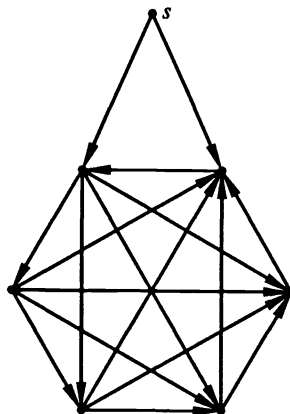


FIG. 8. Another dense BF-orderable graph.

H'_j and never hits v . By construction, there will be an edge from s_j to x . The path going from s_j to x and following P from x is a path from s_j to w in $H'_j - v$. Thus v does not dominate w in H'_j . Since v and w were arbitrary, the dominator tree of H'_j is flat. $\square$

LEMMA 5.6. *If H'_j has more than two vertices, and the corresponding H_j is BF-orderable, then H'_j must be BF-porous.*

Proof. Suppose H'_j has more than two vertices. Then, the graph H_j must have more than two vertices, and by the construction in Step (5) of the main algorithm H_j is strongly connected with external source. Any path of the form $v \rightarrow v' \rightarrow w$ in H_j can be replaced by a path $v \rightarrow w$ connecting the same endpoints in H'_j . Hence, H'_j is also strongly connected with external source. We just showed in Lemma 5.5 that H'_j must have a flat dominator tree. Finally, we construct a BF-ordering for H'_j . For any edge $v \rightarrow w$ in H'_j we choose its label to be the same as either the label of the edge $v \rightarrow w$ in H_j (if it exists) or the edge $v' \rightarrow w$ in H_j (if it exists). If only one of the two candidate edges exist in H_j we use the label of that edge for $v \rightarrow w$; otherwise, we use the smaller of the two labels. $\square$

LEMMA 5.7. *Let G be a graph with at least three vertices that is strongly connected with an external source s . Then, G has a flat dominator tree if and only if for any vertex $v \neq s$, there are two internally vertex-disjoint paths from s to v .*

Proof. If for every v there are two vertex-disjoint paths from s to v , then no vertex other than s can dominate v and G has a flat dominator tree. Proving the other direction takes more work.

Assume that G has a flat dominator tree. We will show that there are two vertex-disjoint paths from s to v by induction on the length of a shortest path from s to v .

For the base case, there is an edge from s to v . Since there is at least one more vertex in the graph, and since $G - s$ is strongly connected, there is at least one edge, say $w \rightarrow v$, to v . Since v does not dominate w , there is a path from s to w in $G - v$. Appending the edge $w \rightarrow v$ to that path yields a path from s to v that is edge disjoint to the edge $s \rightarrow v$.

Assume that every vertex of distance $\leq d$ has two vertex-disjoint paths from s . Let v be at distance $d + 1$ from s . Let P_1 be a shortest path from s to v . Let w be the vertex previous to v on the path P_1 . Since w is at distance d from s , there are two vertex-disjoint paths, P_2 and P_3 , from s to w . Since w does not dominate v , there is a path P_4 from s to v that does not go through w . Follow P_4 backward from v until it first hits either P_2 or P_3 —say P_2 with no loss of generality—at vertex x . Then construct the path P_5 by following P_2 from s to x and then following P_4 from x to v . The path P_3 followed by the edge $w \rightarrow v$ and the path P_5 are vertex-disjoint paths from s to v . (See Fig. 9.) $\square$

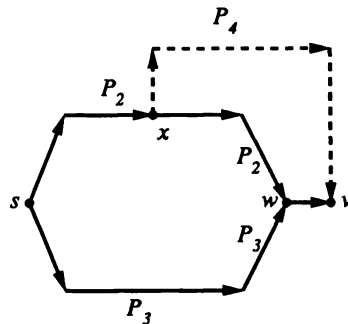


FIG. 9. The induction step in the proof of Lemma 5.7.

COROLLARY 5.8. *A BF-porous graph with at least three vertices has at least two contact points.* $\square$

6. Dividing paths.

DEFINITION 6.1. Consider a path $P = s \rightarrow v_1 \rightarrow v_2 \xrightarrow{*} v_k$ such that there is a contact point w and another vertex x such that neither w nor x are on the path P and such that there is no path $w \xrightarrow{+} x$ in $G - V_P$, where $V_P = \{s, v_1, \dots, v_k\}$ is the set of vertices on the path P . Call such a path a *dividing path*. If the path $P' = s \rightarrow v_1 \rightarrow v_2 \xrightarrow{*} v_{k-1}$ is not a dividing path, then P is a *minimal dividing path*. We sometimes assume implicitly that $s = v_0$.

When given a dividing path P , define the set $L(w)$, for each contact point $w \notin P$, to be the set of vertices reachable from the vertex w in the graph $G - V_P$.

LEMMA 6.2. *Given a minimal dividing path $P = s \rightarrow v_1 \xrightarrow{*} v_k$, and a contact point $w \notin V_P$ such that $V \setminus V_P \setminus L(w) \neq \emptyset$, there is a path $P' = w \xrightarrow{+} v_k$ in the graph induced by $L(w) \cup \{v_k\}$.*

Proof. There is a vertex $x \in G - V_P - L(w)$. Since P is a minimal dividing path, the path $P - v_k$ is not a dividing path. Hence, there is a path $P_1 = w \xrightarrow{+} x$ in the graph $G - V_P \cup \{v_k\}$. Since $x \notin L(w)$, there is no path $w \xrightarrow{+} x$ in $G - V_P$, and hence the path P_1 must go through v_k . Let $P' = w \xrightarrow{+} v_k$ be the prefix of path P_1 up to vertex v_k . Since every vertex of $P' - v_k$ is in $L(w)$ by definition, the path P' is in the graph induced by $L(w) \cup \{v_k\}$. $\square$

When given a dividing path P , define the sets $D(x)$, for each vertex $x \notin P$, to be those vertices from which the vertex x cannot be reached in the graph $G - V_P$. Hence, for example, if $x \notin L(w)$ for some contact $w \notin P$, then $L(w) \subseteq D(x)$.

LEMMA 6.3. *Given a minimal dividing path $P = s \rightarrow v_1 \xrightarrow{*} v_k$, and a vertex $x \notin L(w) \cup V_P$ for some contact point $w \notin P$, there is a path $P' = v_k \xrightarrow{+} x$ with P' contained entirely in the graph $G - V_P - D(x) \cup \{v_k\}$.*

Proof. As in the proof of Lemma 6.2 there is a path $P_1 = w \xrightarrow{+} x$ in the graph $G - V_P \cup \{v_k\}$, since the path P is minimal. And similarly, the path P_1 must go through vertex v_k . This time, however, let $P' = v_k \xrightarrow{+} x$ be the *suffix* of path P_1 starting from v_k . Since every vertex of $P' - v_k$ is not in $D(x)$ by definition, the path P' is in $G - V_P - D(x) \cup \{v_k\}$. $\square$

LEMMA 6.4. *If two contact points $w_1, w_2 \in G - V_P$ are such that*

$$L(w_1) \cap L(w_2) \neq \emptyset,$$

then $L(w_1) = L(w_2)$.

Proof. An easy case is when the contact point $w_1 \in L(w_2)$ and the contact point $w_2 \in L(w_1)$. This implies that $L(w_1) = L(w_2)$ and we are done.

If, however, one of the contacts—without loss of generality say w_1 —is not in the opposite set— $L(w_2)$ —then $L(w_1) \neq L(w_2)$, and we must show that $L(w_1) \cap L(w_2) = \emptyset$. This proof is by contradiction. Hence, assume that there is a vertex $x_1 \in L(w_1) \cap L(w_2)$. By strong connectivity, there is a path $P_0 = x_1 \xrightarrow{+} v_k$. Let v_c be the first vertex where P_0 hits P . Consider the subpath $P_1 = x_1 \xrightarrow{+} v_c$ of path P_0 . By definition, P_1 can pass through neither $L(w_1) \setminus L(w_2)$ nor $L(w_2) \setminus L(w_1)$. Let x_2 be the penultimate vertex of P_1 ; x_2 is the last vertex of $L(w_1) \cap L(w_2)$ along P_1 . Let $P_2 = x_2 \rightarrow v_c$ be the path consisting of the last edge of P_1 .

Since $w_1 \notin L(w_2)$, it follows that $L(w_2) \subseteq D(w_1)$. By Lemma 6.3, there is a path $P_3 = v_k \xrightarrow{+} w_1$ in $G - V_P - D(w_1) \cup \{v_k\} \subseteq G - V_P - L(w_2) \cup \{v_k\}$. By definition of sets $L(w_1)$ and $L(w_2)$, there is a path $P_4 = w_1 \xrightarrow{+} x_2$ in $L(w_1)$ and a path $P_5 = w_2 \xrightarrow{*} x_2$ in $L(w_2)$.

Let C be the simple cycle formed by following P_2 from x_2 to v_c , P from v_c to v_k , P_3 from v_k to where it first hits P_4 (call it x_3) and, last, P_4 from x_3 to x_2 . The cycle C has the three entry points: v_c is an entry point by virtue of the path to it along P ; x_3 is an entry point by virtue of the path to it along P_4 ; and the first vertex of P_4 hit by P_5 is an entry point. The second and third entry points must be distinct because the last entry point is in $L(w_2)$, but $x_3 \notin L(w_2)$. The fact that C has three entry points contradicts our assumption that G is BF-orderable. Hence, our assumption that we could find $x_1 \in L(w_1) \cap L(w_2)$ must be incorrect, and $L(w_1) \cap L(w_2) = \emptyset$. $\square$

7. Density bound. The basic idea behind this proof is to use a dividing path to split a BF-porous graph into two BF-porous pieces. We will take as inductive hypothesis that each BF-porous piece has few edges, and infer that the original BF-porous graph has few edges. Unfortunately, it is rather tedious to show both that splitting the graph does not discard edges and that the two pieces are BF-porous.

THEOREM 7.1. *A BF-porous graph G of $n \geq 2$ vertices with source s and with m contact points has at most $4n - 6 - m$ edges.*

Proof. Let $f(n, m) = 4n - 6 - m$.

If G has only one contact point, then by Corollary 5.8 it has at most two vertices. In this case $n = 2$, $m = 1$, and the number of edges is one, so the bound in the theorem is exact. We assume in the remainder of the proof that G has at least two *distinct* contact points w_1 and w_2 .

The proof is divided into three cases. The first case is when G has no dividing path. In the other two cases, there is a dividing path P , and we choose P to be a shortest (and hence, minimal) dividing path.

Case 1. G has no dividing path.

CLAIM 7.2. *G has exactly two contact points.* Assume to the contrary that there are three contact points w_1 , w_2 , and w_3 . Let $P_1 = w_1 \xrightarrow{*} x_1 \rightarrow w_2$ be a path that does not pass through w_3 ; such a path exists because the edge $s \rightarrow w_3$ cannot constitute a dividing path if we are in this case. Let $P'_1 = w_1 \xrightarrow{*} x_1$ be the initial segment of P_1 . Let $P_2 = w_2 \xrightarrow{*} x_2 \rightarrow w_3$ be a path that does not pass through P'_1 ; such a path exists because the path $s \rightarrow w_1 P'_1$ cannot be a dividing path. Let $P'_2 = w_2 \xrightarrow{*} x_2$ be the initial segment of P_2 . Let $P_3 = w_3 \xrightarrow{+} x_3 \xrightarrow{+} w_1$ be a path that does not pass through P'_2 , and first hits P_1 at x_3 ; let $P'_3 = w_3 \xrightarrow{+} x_3$ be the initial segment of it. Let $P''_1 = x_3 \xrightarrow{+} w_2$ be a segment of P_1 . The simple cycle $C = P'_1 P_2 P'_3$ has entry points w_2 , w_3 , and x_3 . This means that G is not BF-orderable, and hence contradicts the assumption that G is BF-porous. $\square$

CLAIM 7.3. *In $G - s - w_1 - w_2$, there is no simple cycle C of length ≥ 3 .* Assume to the contrary that we have such a cycle C . Let x_1 be an entry point of C such that the path $P_1 = s \rightarrow w_1 \xrightarrow{*} x_1$ establishes that x_1 is an entry point. We can assume without loss of generality that P_1 is in $G - V_C - w_2 \cup \{x_1\}$; in particular, if P_1 were to pass through w_1 and then w_2 , we could replace the $s \xrightarrow{+} w_2$ part of P_1 with $s \rightarrow w_2$ and interchange the roles of w_1 and w_2 . Let $x_3 \rightarrow x_2 \rightarrow x_1$ be the edges of C preceding x_1 . Let $P_2 = x_1 \xrightarrow{+} x_3$ be a segment of C . The path $P_3 = P_1 P_2$ does not include x_2 . Since P_3 cannot be a dividing path, there must be a path $P_4 = s \rightarrow w_2 \xrightarrow{*} x_2$ in $G - V_{P_3} \cup \{s\} \subset G - V_C \cup \{x_2\}$. Thus the vertex x_2 is an entry point of C . A similar argument shows that all vertices of the cycle C are entry points contradicting the assumption that G is BF-porous. $\square$

CLAIM 7.4. *All the vertices in $V \setminus \{s, w_1, w_2\}$ have indegree ≤ 2 .* Assume to the contrary that $x_0 \in V \setminus \{s, w_1, w_2\}$ has indegree at least three. Suppose that x_0 has incoming edges $x_1 \rightarrow x_0$, $x_2 \rightarrow x_0$, and $x_3 \rightarrow x_0$. Since G has a flat dominator tree there are two internally vertex-disjoint paths from s to x_0 . One must go through contact point w_1 and

the other through contact point w_2 . Assume without loss of generality that the shortest path through w_1 subsequently uses the edge $x_1 \rightarrow x_0$, and that the shortest path through w_2 subsequently uses the edge $x_2 \rightarrow x_0$. In summary there is a path $P_1 = s \rightarrow w_1 \xrightarrow{*} x_1 \rightarrow x_0$ in $G - x_3 - x_2 - w_2$ and there is a path $P_2 = s \rightarrow w_2 \xrightarrow{*} x_2 \rightarrow x_0$ in $G - x_3 - x_1 - w_1$. We can assume that neither of the paths P_1 nor P_2 uses the vertex x_3 because if one of them did we could replace the $x_3 \xrightarrow{+} x_0$ suffix with the edge $x_3 \rightarrow x_0$ and instead omit either x_1 or x_2 from both paths. Since the paths P_1 and P_2 are not dividing paths because we are in Case 1, there exist paths $P_3 = s \rightarrow w_1 \xrightarrow{*} x_3 \rightarrow x_0$ in $G - V_{P_2} \cup \{s, x_0\}$ and $P_4 = s \rightarrow w_2 \xrightarrow{*} x_3 \rightarrow x_0$ in $G - V_{P_1} \cup \{s, x_0\}$.

Since $G - s$ is strongly connected there must be paths from x_0 to both contact points. Therefore, there must be a path from x_0 to one of the contact points that does not pass through the other contact point. Without loss of generality, assume that the path is $P_5 = x_0 \xrightarrow{+} w_1$.

- If P_5 does not go through x_3 . Note that P_5 cannot hit P_3 or P_4 except at w_1 or we would have a violation of Claim 7.3. Let P'_3 be the suffix of P_3 with the edge $s \rightarrow w_1$ omitted. Then, the cycle $C = P'_3 P_5$ has w_1 and x_3 as entry points; x_3 is an entry point by virtue of the path to it along P_4 . There is a third entry point at the first place along P_2 where P_2 hits P_5 . The existence of three entry points to C implies that G is not BF-orderable.
- If P_5 does go through x_3 . Clearly P_5 must do so immediately by Claim 7.3, and thus $P_5 = x_0 \rightarrow x_3 \xrightarrow{*} w_1$. Let P'_1 be the suffix of P_1 with the edge $s \rightarrow w_1$ omitted. The cycle $C = P'_1 P_5$ has w_1, x_0 , and the first vertex where P_4 hits P_5 as entry points; the path to x_0 along P_2 makes x_0 an entry point, since if P_2 were to hit P_5 earlier, we would have a simple cycle of length greater than or equal to three. Again, this means that G is not BF-orderable. $\square$

The number of edges of G equals the sum of the indegrees of its vertices. The source vertex s has indegree zero; w_1 and w_2 have indegrees $\leq n - 1$; the other $n - 3$ vertices have indegrees of at most two. Thus G has $|E| \leq 2(n - 1) + (n - 3)2 = 4n - 8 = f(n, 2) = f(n, m)$ edges.

Case 2. The shortest dividing path P is such that in the graph induced by $V \setminus V_P$ every vertex is reachable from a contact point.

For any contact point $w \in V \setminus V_P$ we have already defined $L(w)$ to be the set of vertices reachable from w by a path in the graph induced by $V \setminus V_P$. Define $R(w) = V \setminus V_P \setminus L(w)$. By definition of Case 2, every vertex $x \in G - V_P$ that is not in the set $L(w)$, must be in some other set $L(w')$ such that $w' \notin L(w)$. As a consequence of Lemma 6.4, $L(w') \subseteq D(w)$, and thus $x \in D(w)$. Hence $R(w) \subseteq D(w)$ for all of Case 2.

If $L(w) = V \setminus V_P$ from some contact point w , then by Lemma 6.4, $L(w') = V \setminus V_P$ for all contacts $w' \in V \setminus V_P$. This would imply that V_P is not a dividing path, and so it cannot be the case that $L(w) = V \setminus V_P$. Hence, $R(w) \neq \emptyset$ for every contact point $w \in V \setminus V_P$.

CLAIM 7.5. All edges from $V \setminus V_P$ to V_P are to the vertex v_k , where v_k is the last vertex in the dividing path P . Let $v \in V \setminus V_P$ be given. By definition, $v \in L(w)$ for some contact point $w \notin P$. Since v has been chosen arbitrarily it will suffice to show that there are no edges from $L(w)$ to $V_P \setminus \{v_k\}$.

Assume to the contrary that $e = x \rightarrow v_j$ is an edge with $x \in L(w)$ and $j \neq k$. Let $P_1 = v_j \xrightarrow{+} v_k$ be a segment of path P . By definition of Case 2, there is some contact point w' with $w' \notin L(w)$. Thus, by Lemma 6.3 there is path $P_3 = v_k \xrightarrow{+} x$, in $G - V_P - D(x) \cup \{v_k\}$ which is contained in the graph induced by $L(w) \cup \{v_k\}$. Let $P_4 = w \xrightarrow{*} x$ be a path in the graph induced by $L(w)$. The simple cycle $C = P_1 P_3 e$ has v_j, v_k and the first vertex where P_4 hits P_3 as entry points; v_j is an entry point by virtue of the

path along P and v_k is an entry point by virtue of the path from w' to x that yielded P_2 . This implies that G is not BF-orderable. $\square$

CLAIM 7.6. *All edges from $V_P \setminus \{s\}$ to $V \setminus V_P$ are from the vertex v_k , where v_k is the last vertex in the dividing path P . Let $v \in V \setminus V_P$ be given. By definition, $v \in L(w)$ for some contact point $w \notin P$. Since v was chosen arbitrarily it will suffice to show that there are no edges from $V_P \setminus \{v_k, s\}$ to $L(w)$.*

Assume to the contrary that $e = v_i \rightarrow x$ is an edge with $x \in L(w)$ and $i \neq k$. By Claim 7.5, the only way to get out of $L(w)$ through P is through v_k . Since $G - s$ is strongly connected there is a path $P_2 = x \xrightarrow{*} v_k$ in the graph induced by $L(w) \cup \{v_k\}$. Since $G - s$ is strongly connected there must be a path from v_k to v_i , but this path cannot wander into $V \setminus V_P$ because there is no way out back to V_P . Therefore, there is a path $P_3 = v_k \xrightarrow{+} v_i$ in the graph induced by V_P . Consider the simple cycle $C = P_3 e P_2$. There is a path $s \rightarrow w \xrightarrow{*} x$ in the graph induced by $L(w) \cup \{s\}$; the first place that this path hits C is an entry point. The first place that P hits C is a second entry point. Since $R(w) \neq \emptyset$, there is a contact point $w' \in V \setminus V_P$ such that $w' \notin L(w)$. By Lemma 6.2 there is a path $s \rightarrow w' \xrightarrow{+} v_k$ in $L(w') \cup \{s, v_k\}$. Thus, v_k is a third entry point. This means that G is not BF-orderable. $\square$

CLAIM 7.7. *For any w , the only edges between $L(w) \cup \{v_k\}$ and $R(w)$ are incident to v_k . By definition, there can be no edge between $L(w)$ and $R(w)$.* $\square$

Choose a contact point w . For the remainder of Case 2, let $L = L(w)$ and $R = R(w)$. Let G_1 be the graph induced by $R \cup V_P$. Let G_2 be the graph induced by $L \cup \{s, v_k\}$ with the additional edge $s \rightarrow v_k$. Note that if $s \rightarrow v_k$ is in $E(G)$, then it is in both $E(G_1)$ and $E(G_2)$. We have shown that G contains no edges between a vertex only in G_1 and a vertex only in G_2 , so $|E(G)| = |E(G_1)| + |E(G_2)| - 1$.

CLAIM 7.8. *The graph G_1 is BF-porous.* The graph G_1 is strongly connected with an external source because if two vertices of $R \cup V_P \setminus \{s\}$ had a path between them, then that same path is still in G_1 . The labels that made G BF-orderable can still be used to show that G_1 is BF-orderable.

By Lemma 5.7, for each vertex $x \in G_1$ there are two vertex-disjoint $s \xrightarrow{+} x$ paths in G , call them P_1 and P_2 . By Claims 7.5, 7.6, and 7.7 any $s \rightarrow w_1 \xrightarrow{*} x$ path in G with $w_1 \in G_1$ must be entirely in G_1 . If both paths are in G_1 then we are done. However, some vertex x may have one of its vertex-disjoint $s \xrightarrow{+} x$ paths in G , say P_1 , through a contact point in L and via v_k . If this happens, we split into two cases depending on the location of x .

- *If x is not on P .* Replace the $s \xrightarrow{+} v_k$ segment of P_1 with the path P to obtain a path P'_1 . Since P_1 and P_2 are internally vertex-disjoint, P_2 does not use v_k . By Claims 7.6 and 7.7, path P_2 cannot go through $V_P \setminus \{s, v_k\}$. Thus, P'_1 and P_2 are internally vertex-disjoint $s \xrightarrow{+} x$ paths in G_1 .
- *If x is on P .* Since P_1 uses v_k , Claim 7.5 implies that P_2 must be entirely on P . Furthermore, the segment $P_3 = v_k \xrightarrow{*} x$ of P_1 must be entirely on P . There is at least one contact point $w_1 \notin L(w) \cup P$. By Lemma 6.2 there is a $P_4 = s \rightarrow w_1 \xrightarrow{+} v_k$ path in $L(w_1) \cup \{s, v_k\}$. Thus, P_2 and the path $P_4 P_3$ are vertex-disjoint $s \xrightarrow{+} x$ paths in G_1 .

By Lemma 5.7 the dominator tree of G_1 is flat. $\square$

CLAIM 7.9. *The graph G_2 is BF-porous.* Since there are no edges of G between a vertex only in G_1 and a vertex only in G_2 , and we assumed that $G - s$ is strongly connected, it follows that G_2 is strongly connected with an external source. Temporarily give the edge $s \rightarrow v_k$ the label -1 , keep the labels from G for the other edges of G_2 , and then increment every label by one. This is a valid ordering for $E(G_2)$, so G_2 is BF-orderable. If for a vertex $x \neq s$, the two vertex-disjoint $s \xrightarrow{+} x$ paths that helped make G porous

were through contacts in G_2 , then they both still exist. If one of them was through a contact in $R \cup V_P$, and hence via v_k , then a replacement $s \xrightarrow{+} x$ path can be found starting with the edge $s \rightarrow v_k$. $\square$

Let $n_1 = |L|$. Since $1 \leq n_1 \leq n - 3$ we can use complete induction on n to bound $|E|$. Specifically,

$$\begin{aligned} |E| &\leq f(n_1 + 2, m_1) + f(n - n_1, m - m_1 + 1) - 1 \\ &\leq 4n_1 + 8 - 6 - m_1 + 4n - 4n_1 - 6 - m + m_1 - 1 - 1 \\ &= 4n - 6 - m \\ &= f(n, m), \end{aligned}$$

which is exactly the bound claimed in Theorem 7.1.

Case 3. The shortest dividing path $P = s \rightarrow v_1 \xrightarrow{*} v_k$ is such that in the graph induced by $V \setminus V_P$ some vertices are not reachable from any contact point.

Let L be the set of vertices of $V \setminus V_P$ that are not reachable from any contact point by a path in $G - V_P$. Let $R = V \setminus V_P \setminus L$. Note that R is a union over contact points $w \notin P$ of the sets $L(w)$. Hence for any $x \in L$, we have $R \subseteq D(x)$.

Choose a to be the smallest integer such that there is an edge $x \rightarrow v_a$ for some $x \in L$.

CLAIM 7.10. $k - 1 \leq a \leq k$. By Lemma 6.3, there is a path $P_1 = v_k \xrightarrow{+} x$ in the graph $G - V_P - D(x) \cup \{v_k\}$, which is a subgraph of the graph induced by $L \cup \{v_k\}$. The path P_1 followed by the edge $x \rightarrow v_a$ followed by the segment in P from v_a to v_k form a simple cycle. Lemma 6.2 implies that v_k is an entry point to the cycle via a path from some contact point $w \in R$. The vertex v_a is also an entry point since it can be reached along P . If $a < k - 1$, then we would have a cycle with two nonadjacent entry points violating property BF2 and the assumption that G is BF-orderable. $\square$

Choose b to be the smallest integer such that there is an edge $v_b \rightarrow x$ for some $x \in L$. Note that $v_b \neq s$ because x cannot be a contact point.

CLAIM 7.11. $b < k$ (and $b \leq a$). Since G is BF-porous, for each $x \in L$ there must be two internally vertex-disjoint paths $s \xrightarrow{+} x$. Since there are no contact points in L and since P is a dividing path, both of these paths to x must use an edge from a vertex of P to a vertex of L . The sources of the two edges must be distinct, so $b \leq k - 1$. The claim now follows from Claim 7.10. $\square$

Let $B = \{v_b, v_{b+1}, \dots, v_k\}$. Let $U = V_P \setminus B$.

CLAIM 7.12. *The only edges from R to $L \cup V_P$ are to V_P .* This follows from the definition of R . $\square$

CLAIM 7.13. *There are no edges from $R \cup U \setminus \{s\}$ to L .* By Claim 7.12 there is no edge from R to L . By definition, there is no edge from $U \setminus \{s\}$ to L . $\square$

CLAIM 7.14. *There are no edges from L to $R \cup U$.* There are no edges from L to U by the definition of U . Thus we need to consider only possible edges from L to R . Assume to the contrary that there is an edge $e = x_1 \rightarrow x_2$ with $x_1 \in L$ and $x_2 \in R$. Let w_1 be a contact point with $x_2 \in L(w_1)$. By Lemma 6.2, there is a path $P_3 = w_1 \xrightarrow{+} v_k$ in the graph induced by $L(w_1) \cup \{v_k\}$. By Lemma 6.3 there is a path $P_1 = v_k \xrightarrow{+} x_1$ in the graph $G - V_P - D(x_1) \cup \{v_k\}$ all of the vertices of which are in $L \cup \{v_k\}$. By Lemma 5.7 every vertex has two vertex-disjoint paths from s to it, and hence there is a path $P_2 = s \xrightarrow{+} x_1$ in the graph $G - v_k$. Since there are no edges from R to L , and since there are no entry points in L , it follows that the path P_2 must pass through V_P . Let the last vertex of P_2 on P be v_c . Let $P'_2 = v_c \xrightarrow{+} x_1$. Let $P_6 = w_1 \xrightarrow{*} x_2$ be a path in the graph induced by $L(w_1)$. Strong connectivity implies that there is a path $x_2 \xrightarrow{+} v_k$ in $G - s$.

- If there is a path $P_4 = x_2 \xrightarrow{+} v_k$ in the graph induced by $L(w_1) \cup \{v_k\}$: Thus there is a simple cycle $C = eP_4P_1 = x_1 \rightarrow x_2 \xrightarrow{+} v_k \xrightarrow{+} x_1$. The vertex v_k is an entry point of C by virtue of path P . The vertex in $L(w_1)$ where P_6 first hits P_4 is a second entry point of C . The vertex in L where P'_2 first hits P_1 is a third entry point of C . Since C has three entry points, G cannot be BF-orderable.
- If all paths of the form $x_2 \xrightarrow{+} v_k$ hit $P \setminus v_k$ before they hit v_k : Choosing any one of these, there is a path $P_5 = x_2 \xrightarrow{+} v_i$ in the graph induced by $L(w_1) \cup \{v_i\}$, where $v_i \in V_P \setminus \{v_k\}$ is the first vertex where P_5 hits P . Let $P_7 = v_i \xrightarrow{+} v_k$ be a segment of P . There is a simple cycle $C = eP_5P_7P_1$. The vertex v_i is an entry point of C —via P . The vertex in $L(w_1)$ where P_6 first hits P_5 is a second entry point of C . If path P_3 hit P_5 , then we could find a path $x_2 \xrightarrow{+} v_k$ in the graph induced by $L(w_1) \cup \{v_k\}$ by following P_5 from x_2 until it hit P_3 , and then following P_3 to v_k . The existence of this $x_2 \xrightarrow{+} v_k$ path contradicts the definition of this case, so P_5 cannot hit P_3 , and $s \rightarrow wP_3$ establishes v_k as the third entry point. Since C has three entry points, G cannot be BF-orderable. $\square$

CLAIM 7.15. *For any $x \in L$ and any $j < a$, if there is an edge $v_j \rightarrow x$ then there is a path $x \xrightarrow{+} v_k$ in the graph induced by $L \cup \{v_k\}$. Furthermore, there is no path $x \xrightarrow{+} v_{k-1}$ in the graph induced by $L \cup \{v_{k-1}\}$. By strong connectivity there is a path $P_2 = x \xrightarrow{+} v_k$ in $G - s$. By Claim 7.14, P_2 cannot wander out of L directly into $R \cup U$. By Claim 7.10, if P_2 hits $P - v_k$, it must hit first at v_{k-1} . Thus all we need to show is that P_2 cannot hit v_{k-1} .*

The proof will be by contradiction, so assume that P_2 does hit v_{k-1} . By definition of a dividing path, there must be a contact point $w \in R$. Hence, by Lemma 6.3 there is a path $P_1 = v_k \xrightarrow{+} x$ in the graph induced by $L \cup \{v_k\}$.

Let y be the vertex where P_1 first hits P_2 . Let P'_1 be the segment $v_k \xrightarrow{+} y$ of P_1 . Let P'_2 be the segment $y \xrightarrow{+} v_{k-1}$ of P_2 . Consider the simple cycle $C = P'_2(v_{k-1} \rightarrow v_k)P'_1$. The vertex v_{k-1} is an entry point of C via P and the vertex v_k is an entry point via a path in $R \cup \{v_k, s\}$ (Lemma 6.2). Since there is an edge $v_j \rightarrow x$, with $j < a$, then it is possible to get to x by going along P to v_j and then taking the edge to x . If we then append the part of P_2 from x to y , we get a path $s \xrightarrow{+} y$ that uses no vertex of C except y , so y is a third entry point of C , contradicting the assumption that G is BF-orderable. $\square$

CLAIM 7.16. *The only edges from $R \cup U \setminus \{s\}$ to B are to v_b and v_k . If $b = k - 1$, then $B = \{v_b, v_k\}$, and the claim is satisfied vacuously. Therefore, assume $b \leq k - 2 < a$. By definition of b there is an edge $e = v_b \rightarrow x$ for some $x \in L$. By Claim 7.15 we know that there is a path $P_0 = x \xrightarrow{+} v_k$ in the graph induced by $L \cup \{v_k\}$. Let $P_4 = eP_0$. Suppose, seeking to establish a contradiction, that there is an edge e_2 from $R \cup U \setminus \{s\}$ to $B \setminus \{v_b, v_k\}$. The remainder of the proof is divided into two cases based on the location of the source of e_2 .*

- *From R .* Let $e_2 = x_2 \rightarrow v_i$ be a violating edge with $x_2 \in R$ and $b < i < k$ such that i is maximized. Let $w_1 \in R$ be a contact point such that $x_2 \in L(w_1)$; there is a path $P_2 = w_1 \xrightarrow{+} x_2$ in the graph induced by $L(w_1) \subseteq R$. Let $P_1 = s \xrightarrow{+} v_{i-1}$ be a segment of P . Let $P'_1 = s \xrightarrow{+} v_b$ be a segment of P . A path $P_5 = w_1 \xrightarrow{+} v_k$ in the graph induced by $L(w_1) \cup \{v_k\}$ exists by Lemma 6.2. Let P'_5 be the segment of P_5 that is missing only the last edge to v_k . Choose any path $P_3 = v_k \xrightarrow{+} w_1$; it can only touch P_4 at v_b and v_k by Claims 7.10 and 7.15.
—*Case.* P_3 hits P_2 before it hits $P_1 \cup \{v_i\}$. Let x_3 be the intersection point where P_3 first hits P_2 . Let $P'_3 = v_k \xrightarrow{+} x_3$ be a segment of P_3 . Let g be the smallest integer $> i$ such that v_g is on P'_3 . The simple cycle C formed by following P'_3 from v_g to x_3 , then following P_2e_2 to v_i , then following P to v_g has three entry

- points: x_3 via $s \rightarrow w_1 P_2$; v_i via P ; and the first vertex where the walk $P'_1 P_4 P_3$ hits C .
- Case. P_3 hits P_1 before it hits $P_2 \cup \{v_i\}$. Let v_f be the first vertex where P_3 hits P_1 . Let $P'_3 = v_k \xrightarrow{+} v_f$ be a subpath of P_3 . Break P'_3 into two pieces $P''_3 = v_k \xrightarrow{*} v_g$ and $P'''_3 = v_g \xrightarrow{+} v_f$, where v_g is the last vertex on P'_3 that is in $\{v_{i+1}, \dots, v_k\}$. The simple cycle C formed by following P'''_3 from v_g to v_f , then following P from v_f to v_g has three entry points: v_f via P ; v_i via $P_2 e_2$ (since P'_3 , and hence P'''_3 does not hit P_2); and the first vertex where the walk $P_5 P''_3$ hits C .
 - Case. P_3 hits v_i before P_1 , P_2 , and P'_5 , and P_3 hits P'_5 before it hits P'_1 . The simple cycle formed by following P_3 from v_k until it first hits P'_5 and then following P_5 back to v_k has three entry points: the first vertex where the path $P'_1 P_4$ hits C , which is in $L \cup \{v_k\}$; the vertex where P first hits P_3 ; and the vertex where P_3 first hits P'_5 via $s \rightarrow w_1 P_5$.
 - Case. P_3 hits v_i before P_1 , P_2 , and P'_5 , and P_3 hits P'_1 before it hits P'_5 . The simple cycle formed by following P_3 until P'_1 , and then P'_1 until v_b , and then P_4 until v_k has three entry points: v_k via P_5 ; the vertex where P_3 first hits P'_1 via P ; and the vertex where $P_2 e_2$ first hits the segment of P_3 from v_k to P'_1 via $s \rightarrow w_1 P_2 e_2$.
 - Case. P_3 hits v_i before P_1 and P_2 , and P_3 hits P'_5 at vertex y , before it hits v_i . Let P'_3 be the segment of P_3 from v_k to v_i . Let v_d be the first point along P with $d > i$ that is on P'_3 . Form the simple cycle C by following P from v_i until v_d and then following P_3 back to v_i . Since there is no edge $x \rightarrow v_j$ with $x \in R$ and $i < j < k$, and since there are no edges from R to L the path from y to v_i along P_3 is in $R \cup \{v_i\}$, and hence is part of C . The cycle C has three entry points: v_i via P ; y via $s \rightarrow w_1 P_5$; and the first vertex where the walk $P'_1 P_4 P_3$ hits C .
 - From $U \setminus \{s\}$. Let $e_2 = v_j \rightarrow v_i$ be a violating edge with $j < b$ and $b < i < k$. Let $P_1 = s \xrightarrow{+} v_{b-1}$ be a segment of P . Since P_1 is not a dividing path, there is a path $P_2 = w_1 \xrightarrow{+} v_b$ in the graph induced by $V \setminus V_{P_1}$. As before, let $P_4 = v_b \rightarrow x \xrightarrow{+} v_k$ with $x \in L$. Choose any $P_3 = v_k \xrightarrow{*} w_1$; it can only touch P_4 at v_b and v_k . Let P_5 be an $s \xrightarrow{+} v_k$ path in the graph induced by $R \cup \{s, v_k\}$, which exists by Lemma 6.2.
 - Case. P_3 hits P_1 before P_2 . Let v_g be the first vertex along P_3 that is also on P_1 . The simple cycle formed by following P_3 to v_g , and then following P to v_b , and then following P_4 to v_k has three entry points: v_b via $s \rightarrow w_1 P_2$; v_g via P ; and the first vertex where P_5 hits P_3 .
 - Case. P_3 hits P_2 before P_1 . Let P_6 be the path formed by following P_3 to P_2 and then following P_2 to v_b . If P_6 goes through v_i then the simple cycle C formed by following P_3 to P_2 , P_2 to P_4 (at v_b), and then P_4 to v_k has three entry points: v_b via P ; v_i via the path $s \xrightarrow{+} v_j \rightarrow v_i$; and the first vertex where P_5 hits C . Otherwise P_6 does not go through v_i . The paths P_6 and P intersect in at least two vertices, v_b and v_k , but they may intersect more times. Let v_c be the first intersection point along P_6 such that $c < i$. Let v_d be the previous intersection point along P_6 ; note that $d > i$. The simple cycle C formed by following P_6 from v_d to v_c , and then following P from v_c to v_d has three entry points: v_c via P ; v_i via the path $s \xrightarrow{+} v_j \rightarrow v_i$; and where the walk $P_5 P_3$ first hits C . $\square$

CLAIM 7.17. *None of the vertices in $B' = B \setminus \{v_b, v_k\}$ is a contact point.* Assume to the contrary that $v_i \in B'$ is a contact point. By Claim 7.15 for some $x \in L$ there is a path $P_1 = v_b \rightarrow x \xrightarrow{+} v_k$ in the graph induced by $L \cup \{v_b, v_k\}$. By strong connectivity we can find a path $P_2 = v_k \xrightarrow{+} v_b$. Suppose P_1 and P_2 intersect other than at their endpoints. Any

intersection point must be in L . If P_2 wanders into L , then by Claim 7.10 it has to come out at vertex v_{k-1} (since it started at v_k). Thus, there would be a path from x to the intersection of P_1 and P_2 and then, following P_2 , to v_{k-1} , violating the second part of Claim 7.15. Therefore P_1 and P_2 intersect only at their endpoints and the closed walk P_2P_1 is a simple cycle.

There is a contact point $w \in R$. By Lemma 6.2, there is a path $P_3 = s \rightarrow w \xrightarrow{+} v_k$ in the graph induced by $L(w) \cup \{s, v_k\}$.

If P_2 passes through any v_c with $i \leq c < k$ then the simple cycle $C = P_2P_1$ has three entry points. Vertex v_b is an entry point via P ; the lowest numbered vertex v_c with $i \leq c$ is an entry point via $s \rightarrow v_i$ followed by a piece of P ; since P is a shortest dividing path, P_3 establishes an entry point where it first hits C .

Otherwise, if P_2 does not pass through any v_c with $i \leq c < k$, then instead let $v_c \neq v_k$ be the second vertex where P_2 hits P ; note that $c < i$. Let $P'_2 = v_k \xrightarrow{+} v_c$ be a segment of P_2 . Let $P_4 = v_c \xrightarrow{+} v_k$ be a segment of P . The simple cycle P'_2P_4 has v_i and v_c as entry points. A third entry point is established by path P_3 . $\square$

CLAIM 7.18. *The only edges from B to $R \cup U$ are from v_b and v_k .* Assume to the contrary that there is a violating edge from $v_i \in B'$. By Lemma 5.7 there are two internally vertex-disjoint $s \xrightarrow{+} v_i$ paths. By Claims 7.13, 7.16, and 7.17 there are no edges from $R \cup U \setminus \{s\}$ to $L \cup B'$, and therefore these paths have the form $P_1 = s \xrightarrow{+} v_b \xrightarrow{+} v_i$ and $P_2 = s \xrightarrow{+} v_k \xrightarrow{+} v_i$. Moreover, the segments $P'_1 = v_b \xrightarrow{+} v_i$ and $P'_2 = v_k \xrightarrow{+} v_i$ on those paths must lie entirely in the graph induced by $L \cup B$. Let $P'_1 = s \xrightarrow{+} v_b$ be a segment of P_1 and let $P'_2 = s \xrightarrow{+} v_k$ be a segment of P_2 . Both $V_{P'_1} \setminus \{v_b\}$ and $V_{P'_2} \setminus \{v_k\}$ are in $R \cup U \cup \{v_k\}$ because the internally vertex-disjoint segments P'_1 and P'_2 each end in one of v_b and v_k and because by Claims 7.13 and 7.16 the segments cannot enter $L \cup B$ otherwise. Let $P_4 = v_b \rightarrow x \xrightarrow{+} v_k$ in the graph induced by $L \cup \{v_b, v_k\}$, which exists by Claim 7.15.

If the path P_4 hits P_2 at $y \neq v_k$, then P'_2 must wander into L to reach y . By Claim 7.10 it must return to B at v_{k-1} . Thus there would be a path from $x \xrightarrow{+} y \xrightarrow{+} v_{k-1}$ in the graph induced by $L \cup \{v_{k-1}\}$, which violates Claim 7.15. Hence P_4 only hits P_2 at v_k .

The proof is divided into two cases, depending on the destination of the violating edge.

- *To R .* Let $e = v_i \rightarrow x_1$ be a violating edge with $x_1 \in R$. By strong connectivity, there is a path $P_3 = x_1 \xrightarrow{+} v_k$.

If P_3 hits P_1 at a vertex, x_3 , before it hits P_2 , let $P'_3 = x_1 \xrightarrow{*} x_3$ be a segment of P_3 . The cycle formed by following P'_3 from x_1 to x_3 , then following P_1 from x_3 to v_b , then following P_4 from v_b to v_k , then following P_2 from v_k to v_i and finally following e is a simple cycle. Its entry points x_3 , and v_k are separated by a distance greater than or equal to two, violating conditions BF2 and the assumption that G is BF-orderable.

Otherwise P_3 hits P_2 at a vertex, x_2 , before it hits P_1 . Since $x_1 \in R$, we can find a path $P_5 = s \rightarrow w \xrightarrow{*} x_1$ in the graph induced by $R \cup \{s\}$. The simple cycle C formed by following P_3 from x_1 to x_2 , then following the segment of P_2 from x_2 to v_i , and then following e to x_1 has three entry points: where P first hits C ; where the path P'_1P_4 first hits C ; and where P_5 first hits C .

- *To U .* Let $e = v_i \rightarrow v_g$ be a violating edge with $v_g \in U \setminus \{s\}$. Let $P_3 = v_g \xrightarrow{+} v_b$ be a segment of P . The simple cycle $C = eP_3P_4P'_2$ has v_g and v_k as entry points, which violates BF2. The vertex v_g is an entry point via P and the vertex v_k is an entry point via a $s \xrightarrow{+} v_k$ path in the graph induced by $R \cup \{s, v_k\}$ that exists by Lemma 6.2. $\square$

We now break G into two separate graphs such that each is BF-porous. There are two different possibilities for how we will break the graph. Using induction, we show that in both cases G satisfies the bounds of Theorem 7.1. Let $L' = L \cup B$, and let $R' = R \cup U$.

- *Case 1.* There is no path $v_k \xrightarrow{+} v_b$ in the graph induced by L' . Let G_1 be the graph induced by $R' \cup \{s, v_b, v_k\}$ plus the edge $v_b \rightarrow v_k$. Let G_2 be the graph induced by $L' \cup \{s\}$ plus the edges $s \rightarrow v_k$, $s \rightarrow v_b$ and $v_k \rightarrow v_b$.
- *Case 2.* There is a path $v_k \xrightarrow{+} v_b$ in the graph induced by L' . Let G_1 be the graph induced by $R' \cup \{s, v_b, v_k\}$ plus the edges $v_b \rightarrow v_k$ and $v_k \rightarrow v_b$. Let G_2 be the graph induced by $L' \cup \{s\}$ plus the edges $s \rightarrow v_k$ and $s \rightarrow v_b$.

It is important to observe that some of the new edges such as $v_b \rightarrow v_k$ may already be in G , but if this is the case then the edge is in both G_1 and G_2 . Thus G_1 and G_2 together have exactly four more edges than G , when edges occurring in both graphs are counted twice. In the two claims below let $L'' = L' \setminus \{v_b, v_k\}$.

CLAIM 7.19. *The graph G_1 is BF-porous.* The graph still has s as its external source. If for every $x, y \in G_1 - s$ there is a path $x \xrightarrow{+} y$, then G_1 is strongly connected with external source. Assume to the contrary that there exist x and y such that there is no $x \xrightarrow{+} y$ path in G_1 . Since $G - s$ is strongly connected, we can let $P_0 = x \xrightarrow{+} y$ be a path in G . If P_0 does not exist in G_1 , then in G the path P_0 must pass through L'' . By Claims 7.16 and 7.18, P_0 must pass through P at v_b and v_k . If the segment of P_0 in the graph induced by L' is of the form $v_b \xrightarrow{*} v_k$, then the segment can be replaced by the edge $v_b \rightarrow v_k$ to yield a path $P'_0 = x \xrightarrow{*} y$ in G_1 . Otherwise, the segment of P_0 in the graph induced by L' is of the form $P_1 = v_k \xrightarrow{*} v_b$. Thus there is a path $v_k \xrightarrow{*} v_b$ in the graph induced by L' , and we can modify P_0 by replacing P_1 with the edge $v_k \rightarrow v_b$, thus yielding a path $P'_0 = x \xrightarrow{*} y$ in G_1 . Hence, $G_1 - s$ is strongly connected.

A valid BF-labeling of G_1 can be derived from the labeling of G by assigning the new edge $v_b \rightarrow v_k$ the same label that was on the old $v_{k-1} \rightarrow v_k$ edge of G and assigning the new $v_k \rightarrow v_b$ edge the same label as the lowest-labeled $x \rightarrow v_b$ edge of G such that $x \in L$.

By Claim 7.17, the set L'' contains no contact points. Each vertex $x \in G_1 - s$ has two internally vertex-disjoint $s \xrightarrow{+} x$ paths, P_1 and P_2 , to it in G . If neither of these paths wanders into L'' , then x has the same two internally vertex-disjoint $s \xrightarrow{+} x$ paths to it in G_1 . If one of them, say P_1 , does wander into L'' , then it must use both v_b and v_k and it can be patched to form a path P'_1 in G_1 in the same manner that we patched P_0 to form P'_0 . Since P'_1 uses a subset of the vertices that P_1 used, P'_1 and P_2 are internally vertex-disjoint. Hence, by Lemma 5.7 the dominator tree of G_1 is flat. $\square$

CLAIM 7.20. *The graph G_2 is BF-porous.* The original two vertex-disjoint $s \xrightarrow{+} x$ paths to any vertex x must be via v_b and v_k by Claims 7.16 and 7.17. The prefixes of these two paths can be replaced with $s \rightarrow v_b$ and $s \rightarrow v_k$ to yield vertex-disjoint $s \xrightarrow{+} x$ paths in G_2 .

By strong connectivity of $G - s$ any vertex of $L \cup B$ has a path to R in G . By Claims 7.14 and 7.18 that path must pass through either v_b or v_k . Thus any vertex of $L \cup B$ has a path in G_2 to either v_b or v_k . Since there is a path $v_b \xrightarrow{+} v_k$ that uses only vertices in B , then every vertex in $L' \setminus \{s\}$ has a path to v_k in G_2 . Since there either is a $v_k \xrightarrow{+} v_b$ path in $L \cup B$ or we have added the edge $v_k \rightarrow v_b$ to the graph, it follows that there is path in G_2 from v_k to every vertex in B . By Lemma 6.3, there is a path from v_k to every vertex in L . Since every vertex in $G_2 - s$ has a path to and from v_k , G_2 is strongly connected with an external source. By Lemma 5.7 the dominator tree of G_2 is flat.

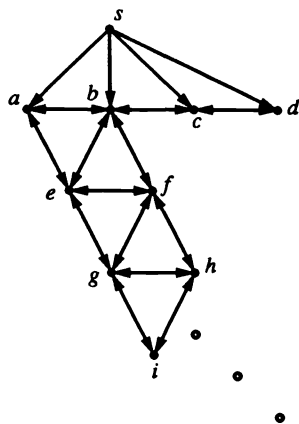


FIG. 10. A class of graphs with $m = 4$ matching the bounds of Theorem 7.1.

By giving the new edges $s \rightarrow v_k$, $s \rightarrow v_b$ and (possibly) $v_k \rightarrow v_b$ the labels -3 , -2 and -1 , respectively, keeping the old labels from G on the other edges, and then adding three to each label, we obtain a labeling that certifies that G_2 is BF-orderable. $\square$

In either case, let $n_2 = |L'|$; thus $n - 2 \geq n_2 \geq 3$, and hence we can use induction on n . $|E| = |E(G_2)| + |E(G_1)| - 4$. Thus G has

$$\begin{aligned} |E| &\leq f(n_2 + 1, 2) + f(n - n_2 + 2, m) - 4 \\ &= 4n_2 + 4 - 6 - 2 + 4n - 4n_2 + 8 - 6 - m - 4 \\ &= 4n - 6 - m \\ &= f(n, m) \end{aligned}$$

edges. $\square$

The result of Theorem 7.1 is tight. This can be seen from the example in Fig. 10 and its obvious generalizations for other n and m . A BF-labeling for the example is given in Table 1.

COROLLARY 7.21. *Let H'_j be BF-orderable and have n' vertices and m' contact points. Then, H'_j has $|E(H'_j)| \leq 4n' - 6 - m'$ edges.* $\square$

Combining Corollary 7.21 with Lemmas 5.4 and 5.6, we obtain Corollary 7.22.

TABLE 1

x	y	$x \rightarrow y$	x	y	$x \rightarrow y$	x	y	$x \rightarrow y$
s	a	1	e	b	11	g	i	21
s	b	2	b	f	12	i	h	22
s	c	3	f	e	13	h	f	23
s	d	4	e	f	14	f	b	24
d	c	5	e	g	15	b	c	25
c	b	6	f	g	16	c	d	26
b	a	7	g	f	17	h	i	27
a	b	8	f	h	18	i	g	28
a	e	9	g	h	19	g	e	29
b	e	10	h	g	20	e	a	30

COROLLARY 7.22. *Let H_j be BF-orderable and have n vertices and m contact points. Then, H_j has $|E(H_j)| \leq 2|E(H'_j)| + (n - n') \leq 8n' - 12 - 2m' + (n - n') \leq 8n - 12 - 2m$ edges. $\square$*

8. Implementation. We implemented our algorithm in the programming language C [Ker78] essentially as it is outlined in §§ 3 and 4 with one exception. We used a very simple $O(|V||E|)$ time dominators algorithm due to Purdom and Moore [Pur72] instead of the more complicated, but faster algorithms due to Harel, Lengauer, and Tarjan.

In implementing step (3) where we construct G' , there is one nontrivial part of the algorithm, which should be discussed briefly for the sake of completeness. This is the part of the construction of G' where we replace back edges. We implemented the replacement of back edges with the following $O(|E|)$ time algorithm.

(1) For every vertex compute its level in the dominator tree with a depth first traversal.

(2) For every vertex v that has descendants in the tree, and therefore causes us to create a vertex v' in G' , remember the corresponding vertex v' .

(3) Do a depth first traversal of the tree keeping the current path in an array, so that the vertex at level i on the current path can be found in constant time by looking at the i th entry.

(4) At each new vertex in the traversal look at all its outgoing edges in G . Decide in constant time whether an edge is a back edge by comparing the tree levels of the two endpoints. For each edge $u \rightarrow v$ that is a back edge, find the level l of v . Let w be the vertex at level l in the current tree path. Find w' . If the edge $w' \rightarrow v$ does not exist insert it. Add $u \rightarrow v$ to the list of back edges that $w' \rightarrow v$ replaces.

We use additional data structures in Step (4) to find the edge $w' \rightarrow v$ or decide that it does not exist in constant time per edge. First we add a special field called **back_edge_replacement** associated with each vertex that belongs to both G' and G . Initially each such field is null (i.e., no replacement exists). When we create the edge $w' \rightarrow v$ we make the **back_edge_replacement** field of v point to this edge. This enables us to check existence in constant time. The key observation that makes this scheme work is that as long as we are exploring the subtree rooted at w , any back edge to v , must be replaced by the same edge, $w' \rightarrow v$. When we complete the depth first search of that subtree we reset the **back_edge_replacement** field to **nil**. To do resetting in constant time per edge we keep a separate list of which back edge replacements were created with destinations at level l , for each l . Then, when we backtrack over a tree vertex at level l we scan down that list resetting the **back_edge_replacement** field of each destination vertex and destroying the list.

9. Verifying an ordering. In this section we consider the following variant of the recognition problem. Given a directed graph with distinguished source vertex s and an ordering of the edges, decide whether the ordering is a valid BF-ordering.

It is not clear if the $O(|V|^2)$ time recognition algorithm can be extended to solve this new problem in $O(|V|^2)$ time. However, the following modification of the Mehlhorn-Schmidt recognition algorithm, can be further modified to give a verification algorithm.

(1) Compute the dominators of $G - v$ for each $v \in V$.

(2) Use the dominator relationship to build the Mehlhorn-Schmidt auxiliary graph (defined in § 2.2).

This can be done as follows: For each $v \in V$: For each edge $u \rightarrow v \in E$: For each edge $v \rightarrow w \in E$: If u is reachable from s in $G - v$ and if w does not dominate u in $G - v$ then there is a path in G from s that ends $u \rightarrow v \rightarrow w$.

Since

$$\sum_{v \in V} d_{\text{in}}(v) \cdot d_{\text{out}}(v) \leq |V| \cdot \sum_{v \in V} d_{\text{in}}(v) = |V| \cdot |E|,$$

this step requires $O(|V||E|)$ time.

(3) Perform a topological numbering of the auxiliary graph. The auxiliary graph has at most $O(|V||E|)$ edges; hence this step takes $O(|V||E|)$ time.

The correctness of our method for computing the auxiliary graph follows from Lemma 9.1.

LEMMA 9.1. *Given a connected graph with $u \rightarrow v \in E$ and $v \rightarrow w \in E$, there is a path from s ending in $u \rightarrow v \rightarrow w$ if and only if there is a path from s to u in $G - v$, and w does not dominate u in $G - v$.*

Proof. If there is a path from s of the form $P = P'u \rightarrow v \rightarrow w$ then the subpath P' is a path from s ending in u that does not go through v or w . Hence, there is a path from s to u in $G - v$, and w does not dominate u in $G - v$.

Assume that there is a path from s to u in $G - v$, and that w does not dominate u in $G - v$. Then there is a path P' in $G - v$ from s to u that does not pass through w . The walk $P'u \rightarrow v \rightarrow w$ is a path in G . $\square$

Any algorithm that builds the auxiliary graph can be modified into a verification algorithm as follows. Build the auxiliary graph and then check if the labeling given for the input graph yields a valid topological ordering of the vertices of the auxiliary graph.

The running time of our algorithm to build the auxiliary graph depends on the cost of computing the dominators in Step (1), which can be done in time $O(|V||E|)$. Thus, this is the overall time for the verification algorithm.

It may be interesting to note that the $|V|$ dominators computations that we do in Step 1 are related to one another. It is natural to ask if there is an algorithm that can do all these computations in time $o(|V||E|)$. However, this would still leave a gap between the complexity of computing an ordering and verifying one.

Acknowledgments. The authors thank Donald Knuth for many useful discussions, pointers, and suggestions, Maria Klawe for pointing out errors in a previous draft, Robert Tarjan for referring us to [Tar81], and an anonymous referee for suggestions regarding the exposition.

REFERENCES

- [Aho79] A. V. AHO AND J. D. ULLMAN, *Principles of Compiler Design*, Addison-Wesley, Reading, MA, 1979.
- [Bel58] R. BELLMAN, *On a routing problem*, Quart. Appl. Math., 16 (1958), pp. 87–90.
- [Dij59] E. W. DIJKSTRA, *A note on two problems in connection with graphs*, Numer. Math., 1 (1959), pp. 269–271.
- [Dri87] J. R. DRISCOLL, H. N. GABOW, R. SHRAIRMAN, AND R. E. TARJAN, *Relaxed heaps: an alternative to Fibonacci heaps*, Technical Report CS-TR-109-87, Princeton University, Princeton, NJ, July 1987, pp. 1–13.
- [Fre84] M. L. FREDMAN AND R. E. TARJAN, *Fibonacci heaps and their uses in improved network optimization algorithms*, in Proc. 25th Annual Symposium on Foundations of Computer Science, 1984, pp. 338–346; J. Assoc. Comput. Mach., 34 (1987), pp. 596–615.
- [Har85] D. HAREL, *A linear time algorithm for finding dominators in flow graphs and related problems*, in Proc. 17th Annual Symposium on Theory of Computing, 1985, pp. 185–194.
- [Joh73] D. B. JOHNSON, *A note on Dijkstra's shortest path algorithm*, J. Assoc. Comput. Mach., 20 (1973), pp. 385–388.

- [Ker78] B. W. KERNIGHAN AND D. M. RITCHIE, *The C Programming Language*, Prentice-Hall, Englewood Cliffs, NJ, 1978.
- [Knu73] D. E. KNUTH, *The Art of Computer Programming, Volume 1: Fundamental Algorithms*, 2nd ed., Addison-Wesley, Reading, MA, 1973.
- [Law76] E. L. LAWLER, *Combinatorial Optimization: Networks and Matroids*, Holt, Reinhart and Winston, New York, 1976.
- [Lei83] C. E. LEISERSON AND R. Y. PINTER, *Optimal placement for river routing*, SIAM J. Comput., 12 (1983), pp. 447–462.
- [Len86] T. LENGAUER AND K. MEHLHORN, *VLSI complexity theory, efficient VLSI algorithms, and the HILL design system*, in *Algorithmics for VLSI*, C. Trullemans, ed., Academic Press, New York. pp. 33–89. (Cited in [Meh86].)
- [Len79] T. LENGAUER AND R. E. TARJAN, *A fast algorithm for finding dominators in a flowgraph*, ACM Trans. Programming Languages and Systems, 1 (1979), pp. 121–141.
- [Low69] E. S. LOWRY AND C. W. MEDLOCK, *Object code optimization*, Commun. ACM, 12 (1969), pp. 13–22.
- [Meh86] K. MEHLHORN AND B. H. SCHMIDT, *On BF-orderable graphs*, Discrete Appl. Math., 15 (1986), pp. 315–327.
- [Pro59] R. T. PROSSER, *Applications of Boolean matrices to the analysis of flow diagrams*, in Proc. AFIPS 1959 Eastern Joint Conference, Spartan Books, Baltimore, MD, pp. 133–138. (Cited in [Aho79].)
- [Pur72] P. W. PURDOM AND E. F. MOORE, *Algorithm 430: immediate predominators in a directed graph*, Commun. ACM, 15 (1972), pp. 777–778.
- [Shi81] D. R. SHIER AND C. WITZGALL, *Properties of labeling methods for determining shortest path trees*, J. Res. Nat. Bur. Standards, 86 (1981), pp. 317–330.
- [Tar72] R. TARJAN, *Depth first search and linear graph algorithms*, SIAM J. Comput., 1 (1972), pp. 146–160.
- [Tar74] ———, *Finding dominators in directed graphs*, SIAM J. Comput., 3 (1974), pp. 62–89.
- [Tar81] ———, *Fast algorithms for solving path problems*, J. Assoc. Comput. Mach., 28 (1981), pp. 594–614.

ON RESTRICTED TWO-FACTORS*

PAVOL HELL†‡, DAVID KIRKPATRICK‡¶, JAN KRATOCHVÍL§, AND IGOR KRÍŽ§

Abstract. A two-factor of G consists of disjoint cycles that cover $V(G)$. The authors consider the existence problem for two-factors in which the cycles are restricted to having lengths from a prescribed (possibly infinite) set of integers. Theorems are presented which derive the existence of such restricted two-factors in G from their existence in $G - u$ and $G - v$. The possibility of such theorems is then related to the complexity of the corresponding existence problem. In particular, the only four cases in which polynomial algorithms can be expected (in the sense that all other cases are shown to be NP-hard) are identified.

Key words. cycles, two-factors, triangle-free, NP-complete, polynomial algorithm

AMS(MOS) subject classifications. 05C38, 05B40, 05C99

1. Introduction. A *two-factor* of a graph G is a subgraph F of G in which $V(F) = V(G)$ and every vertex has degree two. Evidently, a two-factor of G consists of disjoint cycles that cover $V(G)$. The existence question for two-factors has been successfully studied; there is an elegant criterion for deciding if a graph G admits a two-factor, [1], [17], as well as a polynomial algorithm to find such a two-factor (or determine that none exists) [6], [7].

We consider the existence question under the additional restriction that the lengths of all the cycles comprising the two-factor belong to a given set of integers L . Specifically, let $L \subseteq \{3, 4, \dots\}$ be a nonempty set, and let $L^- = \{3, 4, \dots\} - L$. An *L -restricted two-factor* of a graph G is a two-factor F of G with the property that each component of F is a cycle the length of which belongs to the set L . In the case $L^- = \emptyset$ all cycle lengths are permitted, and we refer to it as the case of *unrestricted two-factors*; this is the well-studied case as explained above. The case $L^- = \{3\}$ has also been studied [3] and is usually referred to as the case of *triangle-free two-factors*. Another important case we shall encounter is that of *square-free two-factors*, i.e., the case when $L^- = \{4\}$.

For unrestricted two-factors [8] reports the following fact (cf. also [13]): If $uv \in E(G)$ and if $G - u$ as well as $G - v$ have two-factors, then G also has a two-factor. (This follows from Tutte's f -factor theorem, cf. [8, 13]; a simpler direct proof is given in the next section.) We show that similar results also hold for L -restricted two-factors when $L \subseteq \{3, 4\}$, but for no other L .

Vornberger [16] showed that recognizing graphs that have an odd-restricted two-factor ($L = \{3, 5, 7, \dots\}$) is NP-complete. Other L -restricted two-factor problems were shown to be NP-complete in Cornuejols and Pulleyblank's paper [4] (some of these proofs being attributed to Papadimitriou). On the other hand, [3] reports a polynomial algorithm for finding a triangle-free two-factor (or for showing that none exists). As an application of our constructions we show that recognizing graphs that have an L -restricted two-factor is NP-hard unless $L^- \subseteq \{3, 4\}$, i.e., except for the four particular cases $L^- = \emptyset$, $L^- = \{3\}$, $L^- = \{4\}$, and $L^- = \{3, 4\}$. This subsumes all the previous NP-com-

* Received by the editors July 14, 1987; accepted for publication (in revised form) May 5, 1988.

† School of Computing Science, Simon Fraser University, Burnaby, British Columbia, Canada V5A 1S6.

‡ Department of Computer Science, University of British Columbia, Vancouver, British Columbia, Canada V6T 1W5.

§ Department of Applied Mathematics, Charles University, Prague, Czechoslovakia.

¶ The work of these authors was supported in part by the Natural Sciences and Engineering Research Council of Canada under grants A5075 and A3583.

pleteness results, and in view of the above-mentioned polynomial algorithms for the first two cases ($L^- = \emptyset$, $L^- = \{3\}$), it suggests that the remaining two cases ($L^- = \{4\}$, $L^- = \{3, 4\}$) may also admit such algorithms. As further evidence we mention in the last section some augmenting path theorems that can be proved for these four cases.

2. Positive results. Our objective in this section is to extend the result cited in the Introduction, restated as Theorem 1 below. We begin by giving a simple proof of it, which will be helpful in understanding the extensions. The original proof in [8], as well as that in [13], depends on Tutte's f -factor theorem; the proof given here is elementary.

In the following text we often refer to set operations and terminology (such as union, intersection, symmetric difference, or membership) applied to subgraphs; it is to be understood that they apply to the edge-sets of those subgraphs.

THEOREM 1. *If $uv \in E(G)$ and if both $G - u$ and $G - v$ have two-factors, then G also has a two-factor.*

Proof. Let F_u and F_v be two-factors of $G - u$ and $G - v$, respectively, and let H be their symmetric difference, $H = F_u \oplus F_v$. In H each vertex other than u or v is incident with an equal number of edges of F_u and of F_v (0, 1, or 2 of each). Let P be a maximal trail in H of the form $u = x_0, x_1, \dots, x_k$ with all $x_{2i}x_{2i+1} \in F_v$ and all $x_{2i-1}x_{2i} \in F_u$ ($i = 0, 1, \dots$). (Recall that a trail may revisit a vertex but not an edge.) By the maximality of P and the above observation on incidencies in F_u and F_v , it follows that $x_k = u$ or $x_k = v$. If $x_k = u$ then $F_u \oplus P$ is a two-factor of G ; if $x_k = v$ then $(F_u \oplus P) \cup \{uv\}$ is such a two-factor. $\square$

In subsequent proofs we shall also be using trails of the above type. If H is any graph containing $F_u \oplus F_v$, a trail $P: u = x_0, x_1, \dots, x_k$ in H , with all $x_{2i}x_{2i+1} \in F_v$ and all $x_{2i-1}x_{2i} \in F_u$ ($i = 0, 1, \dots$), shall be called an *alternating u -trail* in H . If P is such a trail, we shall denote by P_i ($i = 0, 1, \dots$) the alternating u -trail $u = x_0, x_1, \dots, x_i$.

THEOREM 2. *If $uv \in E(G)$ and if both $G - u$ and $G - v$ have triangle-free two-factors, then G also has a triangle-free two-factor.*

First we prove a slightly weaker version of Theorem 2. Let G^- denote the graph obtained from G by deleting the edge uv , and G^+ the graph obtained from G by subdividing the edge uv with one new vertex zero, i.e., by replacing the edge uv with the path $u0, 0v$ (for $0 \notin V(G)$). Thus G^- and G^+ are defined relative to a special edge uv ; however, uv will always be clear from the context.

THEOREM 3. *If $uv \in E(G)$ and if both $G - u$ and $G - v$ have triangle-free two-factors, then G^- or G^+ also has a triangle-free two-factor.*

Proof. Again let F_u and F_v be triangle-free two-factors of $G - u$ and $G - v$, respectively, and let $H = F_u \oplus F_v$. The idea here is very similar to the above proof. We shall again be considering alternating u -trails in H ; however, we must be careful to choose P in such a way that the new two-factor does not contain any triangles. Therefore let P be a maximal alternating u -trail $u = x_0, x_1, \dots, x_k$ in H satisfying, for each edge $x_{2i}x_{2i+1}$ ($i = 0, 1, \dots$), all of the following four conditions:

- (i) There is no vertex y with $yx_{2i} \in F_u - (F_v \cup P_{2i})$ and $yx_{2i+1} \in F_u - P_{2i}$;
- (ii) There is no vertex y with $yx_{2i} \in F_u - (F_v \cup P_{2i})$ and $yx_{2i+1} \in F_v \cap P_{2i}$;
- (iii) There is no vertex $y \neq x_{2i+2}$ with $yx_{2i} \in F_v \cap P_{2i}$ and $yx_{2i+1} \in F_u - (F_v \cup P_{2i})$;
- (iv) There is no vertex $y \neq x_{2i+2}$ with $yx_{2i} \in F_u \cap F_v$ and $yx_{2i+1} \in F_u - (F_v \cup P_{2i})$.

In Fig. 1, $\in$ and $\notin$ denote, respectively, the membership and nonmembership in P_{2i} (except for the obvious relation $x_{2i}x_{2i+1} \notin P_{2i}$, which we did not show in the figures). The edges of $F_u - F_v$ are wavy, and those of $F_v - F_u$ are straight. Edges of $F_u \cap F_v$ are both wavy and straight; edges of F_u (which could also belong to F_v or not) are wavy with an interrupted straight line.

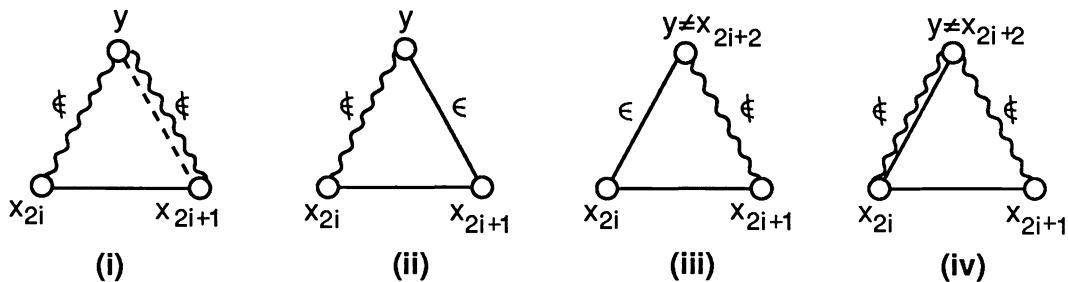


FIG. 1. The four forbidden configurations.

CLAIM 1. The trail P ends at $x_k = u$ or $x_k = v$.

Suppose $x_k \neq u, v$. Then x_k has an equal number (zero, one, or two) of incident edges of H in F_u and F_v ; thus P could not be maximal unless $k = 2i$ and each choice of $x_{2i}x_{2i+1} \in F_v - (F_u \cup P)$ violates one of (i)–(iv). Specifically, if $k = 2i - 1$ and P satisfies (i)–(iv), then letting $x_{2i-1}x_{2i}$ be any edge of $F_u - (F_v \cup P)$ extends P so that it still satisfies (i)–(iv), contrary to the maximality of P . Now, if $k = 2i$ and a particular choice of $x_{2i}x_{2i+1} \in F_v - (F_u \cup P)$ violates (i) or (ii), then there is a vertex y as described in the conditions and depicted in the figures. On the other hand, if some $x_{2i}x_{2i+1} \in F_v - (F_u \cup P)$ violates (iii) or (iv), then that means a violation occurs regardless of what choice of x_{2i+2} is made (in other words, in order to satisfy (iii), (iv), two distinct vertices would need to be x_{2i+2}). If there are two edges incident with x_{2i} in $F_v - (F_u \cup P)$, and if taking either as $x_{2i}x_{2i+1}$ violates one of (i)–(iv), then it could be because both violate one of (i), (ii), or because both violate one of (iii), (iv), or because one violates one of (i), (ii) and the other one of (iii), (iv). We shall treat the latter two cases together; taking into account also the possibility of there being only one edge incident with x_{2i} in $F_v - (F_u \cup P)$, we distinguish two situations:

(A) For each choice of $x_{2i}x_{2i+1}$ some vertex y violates (i) or (ii).

(B) For some choice of $x_{2i}x_{2i+1}$ two distinct vertices $y, y^{\sim}$ are required to be x_{2i+2} by (iii) or (iv).

Clearly if neither (A) nor (B) occurs, then some choice of $x_{2i}x_{2i+1}$ leads to no violation of (i)–(ii) and at most one vertex y is required to be x_{2i+2} by (iii)–(iv). Therefore, letting x_{2i+2} equal such a vertex y if it exists, or terminating at x_{2i+1} if it does not exist, then the maximality of $P: u = x_0, x_1, \dots, x_{2i}$ is contradicted.

We now show that both (A) and (B) are impossible. In case (A) we have $x_{2i-1}x_{2i} \in F_u \cap P_{2i}$ while any violation of (i) or (ii) involves a vertex y with $yx_{2i} \in F_u - (F_v \cup P_{2i})$. Thus x_{2i} is incident in H with two edges of F_u and two edges of F_v . If the two edges of F_v are $x_{2i}z$ and $x_{2i}z^{\sim}$ then z or $z^{\sim}$ are the choices for x_{2i+1} . Suppose $x_{2i+1} = z$ violates (i) or (ii) because of some y , and $x_{2i+1} = z^{\sim}$ violates (i) or (ii) because of some $y^{\sim}$. Clearly $y = y^{\sim}$ or else x_{2i} would be incident with three edges of F_u . For the same reason both $x_{2i+1} = z$ and $x_{2i+1} = z^{\sim}$ could not violate (i); thus we may assume that the choice $x_{2i+1} = z^{\sim}$ violates (ii). Then $yz^{\sim} \in F_v \cap P_{2i}$ yet no edge of F_u incident to y belongs to P_{2i} ; this is impossible as P_{2i} begins at $u \neq y$ and ends at $x_{2i} \neq y$. In case (B) we have $x_{2i}x_{2i+1} \in F_v$; if $y \neq y^{\sim}$ both violate (iii), (iv) then $x_{2i}y$ and $x_{2i}y^{\sim}$ are two more edges of F_v incident to x_{2i} , a contradiction. $\square$

CLAIM 2. If $x_k = u$, then $F_u \oplus P$ is a triangle-free two-factor of G^- . If $x_k = v$, then $(F_u \oplus P) \cup \{u0, 0v\}$ is a triangle-free two-factor of G^+ .

If $F_u \oplus P$ contains a triangle abc , then in G we have

- (a) $ab \in F_v \cap P$ and $ac, bc \in F_u - P$, or
- (b) $ab \in F_u - P$ and $ac, bc \in F_v \cap P$.

This follows from the definition of $F_u \oplus P$ and the fact that abc is not a triangle of F_u or of F_v . If $(F_u \oplus P) \cup \{u0, 0v\}$ contains a triangle abc , then abc does not contain the edges $u0, 0v$, and hence we also have (a) or (b) in G . Now we show that both (a) and (b) are impossible in G : In case (a) if, say, $a = x_{2i}$ and $b = x_{2i+1}$, then P violates (i) unless $ac \in F_u \cap F_v$ in which case $bc \in P$ according to (iv). In case (b) we may assume without loss of generality that bc precedes ac on P , i.e., that bc is an edge $x_j x_{j+1}$ and ac an edge $x_{j'} x_{j'+1}$ with $j < j'$. If $a = x_{2i}$ and $c = x_{2i+1}$, then P violates (ii), with $y = b$. If $c = x_{2i}$ and $a = x_{2i+1}$, then by (iii), $b = x_{2i+2}$, a contradiction to $ab \notin P$. $\square$

To prove Theorem 2 from Theorem 3, observe that G^- is a subgraph of G . Thus it only remains to consider the case when G^+ contains a triangle-free two-factor but G does not. In terms of the above proof, that means $(F_u \oplus P) \cup \{uv\}$ contains a triangle, i.e., that the path P constructed in that proof ends at $x_k = v$ and that $x_1 v \in F_u - P$. In this case G has an obvious triangle-free two-factor obtained from F_u by replacing $x_1 v$ with the path $x_1 u, uv$. $\square$

Note that Theorems 1 and 2 do not have a strict analogue for square-free two-factors, as can be seen by considering the graph K_4 with respect to any edge uv . (We thank Pierre Fraisse for this observation.) Nevertheless, Theorem 3 does generalize as follows.

THEOREM 4. *If $uv \in E(G)$ and if both $G - u$ and $G - v$ have square-free two-factors, then G^- or G^{++} also has a square-free two-factor.*

Here G^{++} is the graph obtained from G by subdividing the edge uv with two new vertices 0 and 1, i.e., by replacing uv with the path $u0, 01, 1v$ (for $0, 1 \notin V(G)$).

Proof. Again we assume that F_u and F_v are square-free two-factors of $G - u$ and $G - v$, respectively. However, instead of defining H as the symmetric difference, we let $H = F_u \cup F_v$; this will have the effect of allowing the edges of $F_u \cap F_v$ to be used in alternating trails. Specifically, let P be a maximal alternating u -trail in H , $P: u = x_0, x_1, \dots, x_k$, with each edge $x_{2i} x_{2i+1} \in F_v - F_u$ ($i = 0, 1, \dots$) and satisfying all of the following eight conditions:

- (v) There is no edge $yy' \in F_u - P_{2i}$ with $yx_{2i} \in F_u - (F_v \cup P_{2i})$ and $y'x_{2i+1} \in (F_v - F_u) \cap P_{2i}$;
- (vi) There is no edge $yy' \in (F_u \cap F_v) - P_{2i}$ with $yx_{2i} \in F_u - (F_v \cup P_{2i})$ and $y'x_{2i+1} \in F_v - (F_u \cup P_{2i})$;
- (vii) There is no edge $yy' \in (F_v - F_u) \cap P_{2i}$ with $yx_{2i} \in F_u - (F_v \cup P_{2i})$ and $y'x_{2i+1} \in F_v - F_u$;
- (viii) There is no edge $yy' \in F_u - P_{2i}$ with $y' \neq x_{2i+2}$ and $yx_{2i} \in F_u - P_{2i}$, $y'x_{2i+1} \in F_u - P_{2i}$;
- (ix) There is no edge $yy' \in (F_u \cap F_v) - P_{2i}$ with $y' \neq x_{2i+2}$ and $yx_{2i} \in F_v - (F_u \cup P_{2i})$, $y'x_{2i+1} \in F_u - (F_v \cup P_{2i})$;
- (x) There is no edge $yy' \in F_u - P_{2i}$ with $y' \neq x_{2i+2}$ and $yx_{2i} \in (F_v - F_u) \cap P_{2i}$, $y'x_{2i+1} \in F_u - (F_v \cup P_{2i})$;
- (xi) There is no edge $yy' \in (F_v - F_u) \cap P_{2i}$ with $y' \neq x_{2i+2}$ and $yx_{2i} \in F_u - P_{2i}$, $y'x_{2i+1} \in F_u - P_{2i}$;
- (xii) There is no edge $yy' \in (F_v - F_u) \cap P_{2i}$ with $y' \neq x_{2i+2}$ and $yx_{2i} \in F_v - F_u$, $y'x_{2i+1} \in F_u - (F_v \cup P_{2i})$.

In conditions (v)–(xii) it is to be assumed that yy' is disjoint from $x_{2i} x_{2i+1}$ as shown in Fig. 2. (We use the same conventions as in Fig. 1.)

So far we have encountered one basic difference from the previous proofs (in addition to an evident increase in complexity): the trail P can use the edges of $F_u \cap F_v$, although only in positions $x_{2i-1} x_{2i}$ ($i = 0, 1, \dots$). The second basic difference is the following: recall that the situation (B) from Claim 1 was shown impossible there. The analogous situation here can actually occur (see Fig. 3).

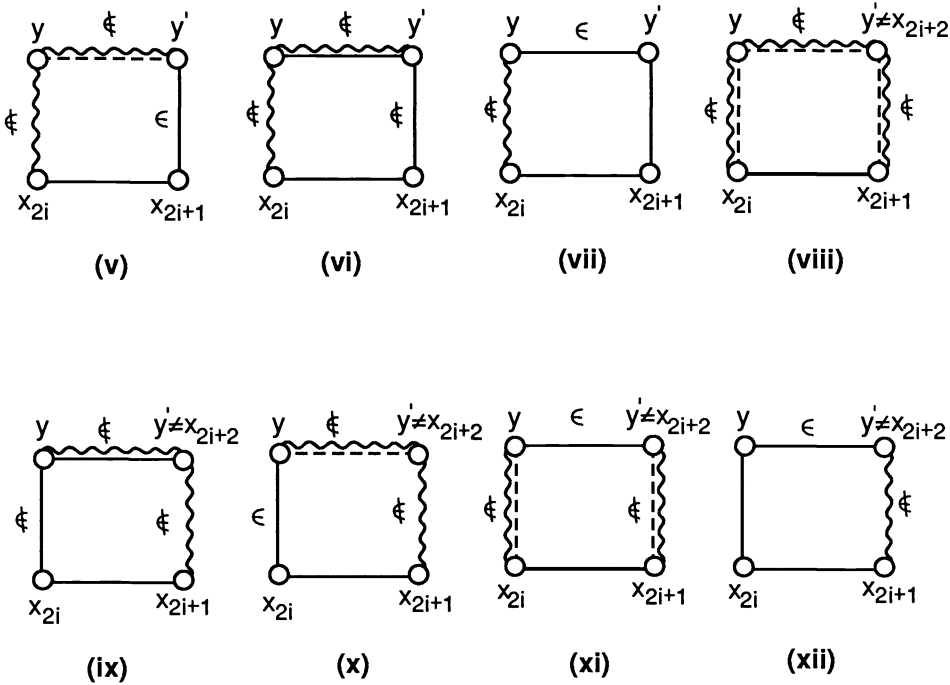
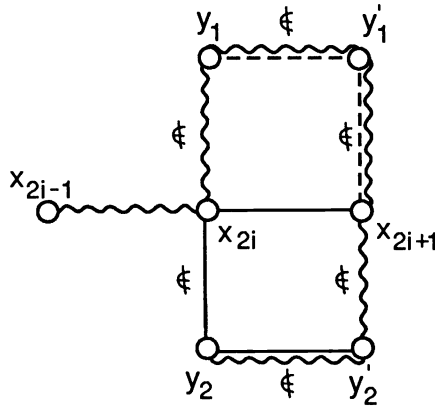


FIG. 2. The eight forbidden configurations.

FIG. 3. A choice of $x_{2i}x_{2i+1}$ that would require $x_{2i+2} = y'_1$ (by (viii)) and $x_{2i+2} = y'_2$ (by (ix)).

Nevertheless, the following claim remains true.

CLAIM 1. The trail P ends at $x_k = u$ or $x_k = v$.

As before, if $x_k \neq u, v$ then $k = 2i$, and for each choice of $x_{2i}x_{2i+1} \in F_v - (F_u \cup P)$ one of the following situations must occur:

(A) Some edge yy' violates (v), (vi), or (vii), or

(B) Some edges $y_1y'_1$, $y_2y'_2$ force two different choices $x_{2i+2} = y'_1$ and $x_{2i+2} = y'_2$ by (viii)–(xii).

We first prove that it is impossible that for all choices (there are at most two) of $x_{2i}x_{2i+1} \in F_v - (F_u \cup P)$ situation (A) occurs: assume that for $x_{2i}x_{2i+1}$ (v), (vi), or (vii)

is violated by yy' . Since $x_{2i-1}x_{2i} \in F_u \cap P_{2i}$, we have $y \neq x_{2i-1}$, and hence there is another choice $x_{2i}x_{2i+1} \in F_v - (F_u \cup P)$; assume that for $x_{2i}x_{2i+1}$ (v)–(vii) is violated by $y \sim y'$. Clearly $y \sim = y$, or else x_{2i} is incident with three edges of F_u . If $y \sim' \neq y'$, then for the same reason one of yy' and $yy \sim'$ is not in F_u ; hence, say, $x_{2i}x_{2i+1}$ violates (vii) and therefore $x_{2i}x_{2i+1}$ violates (v) or (vi). (Otherwise there is an unbalanced number of edges of $F_u \cap P$ and $F_v \cap P$ at $y = y \sim$.) This is not possible, as $y \neq u$, x_k would be incident with exactly one edge of P . If, on the other hand, $y' = y \sim'$ then $yy' \in F_u - F_v$ or else y' is incident with three edges of F_v . Thus both $y'x_{2i}$ and $y'x_{2i}$ are in $F_v \cap P_{2i}$ while $yy' \in P_{2i}$; this is impossible for $y' \neq u, x_k$.

As shown in Fig. 3, some choice of $x_{2i}x_{2i+1}$ may lead to (B). However, there is only a limited number of configurations that can cause this to happen: Assume that $x_{2i}x_{2i+1}$ leads to (B), i.e., that conditions (viii)–(xii) require two different choices $x_{2i+1} = y'_1$ and $x_{2i+1} = y'_2$ for some edges $y_1y'_1, y_2y'_2$. We first prove that $y_1 \neq y_2$ (in addition to $y'_1 \neq y'_2$). If $y_1 = y_2$ then

- $x_{2i}y_1 \in F_u - P_{2i}$, and we obtain a contradiction at y_1 (three edges of F_u , or an unbalanced number of edges of $F_u \cap P_{2i}$ and $F_v \cap P_{2i}$); or

- $x_{2i}y_1 \in F_v - P_{2i}$, and we have a similar contradiction (three edges of F_v incident to y_1); or

- $x_{2i}y_1 \in F_v \cap P_{2i}$, and we either note a similar contradiction again (three edges of F_v incident to y_1 , or an unbalanced number of edges of $F_u \cap P_{2i}$ and $F_v \cap P_{2i}$ at y_1), or, if both $y_1y'_1$ and $y_2y'_2$ contradict condition (x), we obtain the four-cycle $y_1y'_1x_{2i+1}y'_2y_1$, contradicting the fact that F_u is square-free.

Therefore $y_1 \neq y_2$, and since $x_{2i-1}x_{2i} \in F_u$, $x_{2i}x_{2i+1} \in F_v$, we may assume without loss of generality that $x_{2i}y_1 \in F_u$ and $x_{2i}y_2 \in F_v$ (as in Fig. 3). In fact, considering the position of x_{2i} on the path P_{2i-1} , and the fact that none of the edges $x_{2i}y$ in (viii)–(xii) belong to $F_u \cap P_{2i}$, we conclude that $x_{2i}y_1 \in F_u - P_{2i}$ and $x_{2i}y_2 \in F_v - P_{2i}$. This leaves the four possibilities obtained from combining each of (viii), (xi) with each of (ix), (xii); in addition to the case illustrated in Fig. 3, we list them in Fig. 4.

We have just proved the following subclaim.

SUBCLAIM. If $x_{2i}x_{2i+1}$ leads to (B) then H contains one of the four situations depicted in Figs. 3 and 4.

Note that in Figs. 3 and 4 we have marked the edge $x_{2i}y_2$ as $\notin P_{2i}$ even though (xii) does not require it; this is so because $x_{2i}y_2 \in P_{2i}$ would mean $x_{2i}y_2 \in P_{2i-1}$, contradicting the fact that x_{2i} is not incident with any edge of $F_u \cap P_{2i-1}$. (Recall that we are still assuming that $x_{2i} = x_k \neq u$, so that P_{2i-1} cannot begin at x_{2i} .)

In all four cases there is an obvious alternate choice, $x_{2i+1} = y_2$. We will be done if we can show that $x_{2i}x_{2i+1}$ does not lead to (A) or (B).

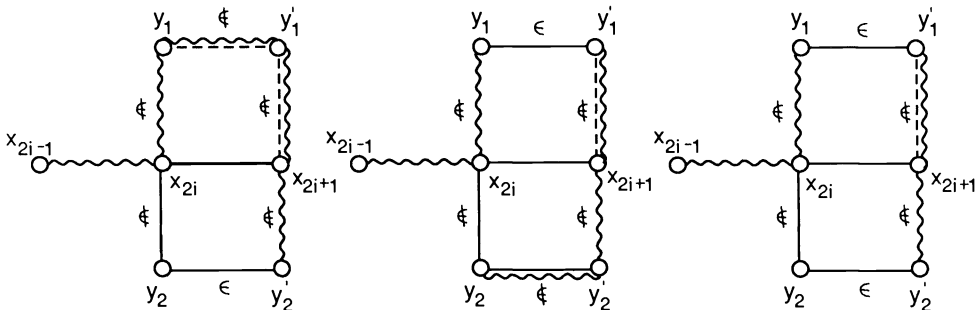


FIG. 4. The other configurations causing (B).

Case 1. $x_{2i}x_{2i+1}$ leads to (B).

According to the above subclaim, there is a configuration similar to one of the situations depicted in Figs. 3 and 4, on vertices $x_{2i-1}, x_{2i}, x_{2i+1}, y_1, y_1', y_2, y_2'$ (see Fig. 5).

By considering the degree of x_{2i} in F_u and F_v we conclude that $y_1 = y_1'$ and $y_2 = y_2'$. Note that it is not possible that both $y_1 y_1' \in F_u$ and $y_1 y_1' \in F_v$ because in such a case we would necessarily have $y_1' = y_1$ and $y_1' x_{2i+1} \in F_u, y_1' x_{2i+1} \in F_v$, yielding three edges of F_u at y_1' . Otherwise, say, $y_1 y_1' \in F_v \cap P$, as depicted in the example in Fig. 5, and, regardless of whether $y_1 y_1' \in F_u - P$ or $y_1 y_1' \in F_v \cap P$, we obtain a contradiction at $y_1 = y_1'$ arising from an unbalanced number of edges of $F_u \cap P$ and $F_v \cap P$. Thus Case 1 does not occur.

Case 2. $x_{2i}x_{2i+1}$ leads to (A).

Suppose x_{2i}, x_{2i+1}, y, y' violate (v), (vi), or (vii). Then $y = y_1$ and $y' = y_2'$. If $y y' \in F_u - P$ (as in (v) and (vi)) then we have a contradiction at y_1 because of the degree in F_u or an imbalance between $F_u \cap P$ and $F_v \cap P$. If $y y' \in F_v \cap P$ (as in (vii)) we also have a contradiction because of an imbalance between $F_u \cap P$ and $F_v \cap P$ at y_1 . Thus Case 2 does not occur. This completes the proof of Claim 1. $\square$

CLAIM 2. If $x_k = u$, then $F_u \oplus P$ is a square-free two-factor of G^- . If $x_k = v$, then $(F_u \oplus P) \cup \{u0, 01, 1v\}$ is a square-free two-factor of G^{++} .

If $F_u \oplus P$ or $(F_u \oplus P) \cup \{u0, 01, 1v\}$ contains a four cycle $abcd$, then we may assume that in G we have the following:

- (a) $ab \in (F_v - F_u) \cap P, bc, cd, da \in F_u - P$; or
- (b) $ab, cd \in (F_v - F_u) \cap P, bc, da \in F_u - P$; or
- (c) $ab, bc \in (F_v - F_u) \cap P, cd, da \in F_u - P$; or
- (d) $ab, bc, cd \in (F_v - F_u) \cap P, da \in F_u - P$.

We now prove that each of these four situations is impossible.

(a) Without loss of generality $a = x_{2i}, b = x_{2i+1}$ for some i ; then $y = d, y' = c$ contradicts (viii) because $bc \notin P$ implies that $y' \neq x_{2i+2}$.

(b) Similarly, we may assume without loss of generality that cd precedes ab on P and that $a = x_{2i}, b = x_{2i+1}$ for some i (thus, $cd \in P_{2i}$); then, as in (a), $y = d, y' = c$ contradicts (xi).

(c) We may assume that, say, bc precedes ab on P ; but we must consider both possibilities, that $a = x_{2i}, b = x_{2i+1}$, or that $a = x_{2i+1}, b = x_{2i}$ (for some i). In the first

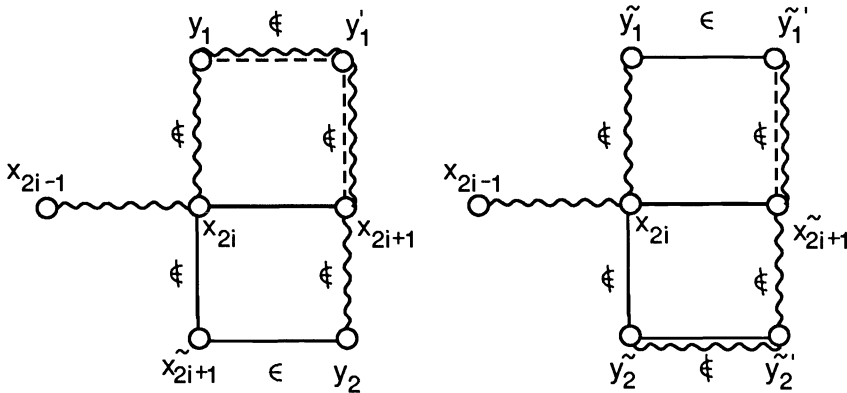


FIG. 5. Both $x_{2i}x_{2i+1}$ and $x_{2i}x_{2i+1}$ lead to (B). (This is only an example; there are 16 possible combinations to illustrate.)

case ($a = x_{2i}, b = x_{2i+1}$), $y = d, y' = c$ contradicts (v), unless $da \in F_u \cap F_v$ (see Fig. 6). In this situation (when $da \in F_u \cap F_v$) we must have $cd \notin F_v$, because F_v is square-free. Now $bc = x_{2j}x_{2j+1}$ (for some $j < i$): either $c = x_{2j}, b = x_{2j+1}$ and $y = d, y' = a$ contradicts (vi), or $b = x_{2j}, c = x_{2j+1}$ and $y = a, y' = d$ contradicts (ix). (For both of these contradictions we must remember to consider memberships in P_{2j} and not P_{2i} .) In the *second case* ($b = x_{2i}, a = x_{2i+1}$), $y = c, y' = d$ contradicts (x), unless $da \in F_u \cap F_v$, which is the situation dealt with above. Thus (c) is also impossible.

(d) We first prove that there is no four-cycle $prst$ in G such that

$$pr \in F_u - (F_v \cup P_{2i}), \quad rs, tp \in (F_v - F_u) \cap P_{2i}, \quad st \in F_v - (F_u \cup P_{2i})$$

for any i . Indeed, if such a four-cycle $prst$ were to exist in G , then we may assume that sr precedes pt on P and that $pt = x_{2j}x_{2j+1}$ for some $j < i$. If $p = x_{2j}$ and $t = x_{2j+1}$, then $y = r, y' = s$ contradict (vii); if $t = x_{2j}$ and $p = x_{2j+1}$ then $y = s, y' = r$ contradict (xii). Thus such a four-cycle $prst$ cannot exist. It now follows that if (d) were to occur in G , then bc cannot come after both ab and cd on P . (If $bc = x_{2i}x_{2i+1}$ and $ab, cd \in P_{2i}$ then $abcd = rstp$ yields a contradiction.) Hence we may assume without loss of generality that $ab = x_{2i}x_{2i+1}$ and $bc, cd \in P_{2i}$: if $a = x_{2i}$ and $b = x_{2i+1}$ then $y = d, y' = c$ contradict (vii); if $a = x_{2i+1}$ and $b = x_{2i}$ then $y = c, y' = d$ contradict (xii). This proves that (d) is impossible; Claim 2 and Theorem 4 follow. $\square$

Let G be a graph and $uv \in E(G)$. We denote by $\mathcal{G}^*$ the set of all graphs obtained from G by subdividing the edge uv by any (finite) number of vertices. In the remainder of the paper we shall often refer to elements of $\mathcal{G}^*$. For simplicity we shall say *some* (any, all) G^* to mean *some* (any, all) $G^* \in \mathcal{G}^*$.

THEOREM 5. Let $L^- \subseteq \{3, 4\}$. If $uv \in G$ and if both $G - u$ and $G - v$ have L -restricted two-factors, then G^- or some G^* also has an L -restricted two-factor.

Proof. For the purposes of this proof, G^* may be chosen as follows: if L^- is nonempty and finite, the edge uv is subdivided with $\max(L^-) - 2$ (or more) new vertices; otherwise $G^* = G$.

When $L^- = \emptyset$, $G^* = G$ and Theorem 1 gives the desired conclusion. When $L^- = \{3\}$, $G^* = G^+$ and we apply Theorem 3. When $L^- = \{4\}$, $G^* = G^{++}$ and we conclude by Theorem 4. Finally, when $L^- = \{3, 4\}$, we take L -restricted two-factors F_u of $G - u$ and F_v of $G - v$, let $H = F_u \cup F_v$, and define $P: u = x_0, \dots, x_k$ to be a maximal alternating u -trail in H with each edge $x_{2i}x_{2i+1} \in F_v - F_u$ ($i = 0, 1, \dots$) and satisfying all 12 constraints (i)–(xii).

CLAIM 1. The trail ends at $x_k = u$ or $x_k = v$.

Otherwise, $k = 2i$ and for each choice of $x_{2i}x_{2i+1}$, we have the following:

(A) Some vertex y violates (i)–(ii), or some edge yy' violates (v)–(vii); or

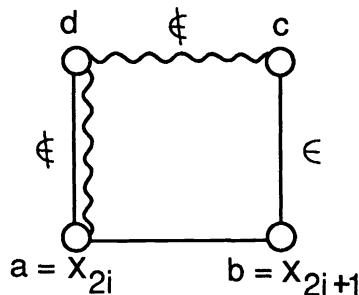


FIG. 6. A possible way to create (c). (The membership sign $\in$ refers to membership in P_{2i} .)

(B) Some two vertices $y'_1 \neq y'_2$ are required to be x_{2i+2} by (iii)–(iv) or (viii)–(xii).

It again turns out that not every choice of $x_{2i}x_{2i+1}$ can lead to (A) or (B). Most of the work in verifying this is given in the preceding proof; it only needs to be checked that adding the rules (i)–(iv) does not change the situation. These verifications are similar and shall be omitted here. $\square$

CLAIM 2. If $x_k = u$, then $F_u \oplus P$ is an L -restricted two-factor of G^- . If $x_k = v$, then $(F_u \oplus P) \cup \{u0, 01, 1v\}$ is an L -restricted two-factor of G^* .

Since P satisfies both (i)–(iv) and (v)–(xii), Claim 2 follows from the corresponding claims in the proofs of Theorems 3 and 4. $\square$

3. Negative results. We begin by showing that Theorem 5 is the best possible in the following sense.

THEOREM 6. For any nonempty set $L \subseteq \{3, 4, \dots\}$ with $L^- \not\subseteq \{3, 4\}$ there exists a graph G with an edge uv such that both $G - u$ and $G - v$ have L -restricted two-factors, but neither G^- nor any G^* has an L -restricted two-factor.

Proof. We define, for all $\lambda \geq 3$ and all $p \geq 0, q \geq 0$, the graphs D_λ and $P_{p,q}$ (see Fig. 7).

Let L be finite and let $\lambda = \max L$. Then $G = D_\lambda$ satisfies the conditions in Theorem 6. Indeed, both $D_\lambda - u$ and $D_\lambda - v$ are just λ -cycles, while any two-factor of D_λ^- or of any D_λ^* is a cycle with more than λ vertices.

Let there exist nonnegative integers p and q such that $k = p + q + 5 \in L^-$ and $n = p + 2q + 9 \in L$. Then $G = P_{p,q}$ satisfies the conditions in Theorem 6. In fact, $(P_{p,q} - u) - \{ab, a'b'\}$ is a cycle of length n , and $(P_{p,q} - v) - \{tb', t'b\}$ is also a cycle of length n . On the other hand, any two-factor F of G^- or of any G^* must contain the paths $t \cdots ra$, and $t' \cdots r'a'$. In fact, F must also contain the path $b \cdots b'$; this is obvious when $p \neq 0$, and for $p = 0$ follows from the fact there is otherwise the forced cycle $t \cdots rabt' \cdots r'a'b't$, which misses u and v . Moreover, F must also contain at least one of the edges $tb', t'b$; otherwise F would miss v . If F contains both tb' and $t'b$, then F must miss u . So, without loss of generality, $tb' \in F$ and $t'b \notin F$. Then the cycle $t \cdots rab \cdots b't$ in F is of the forbidden length k . Since this cycle does not contain the special edge uv , the argument applies both to G^- and to any G^* , and hence they cannot have an L -restricted two-factor.

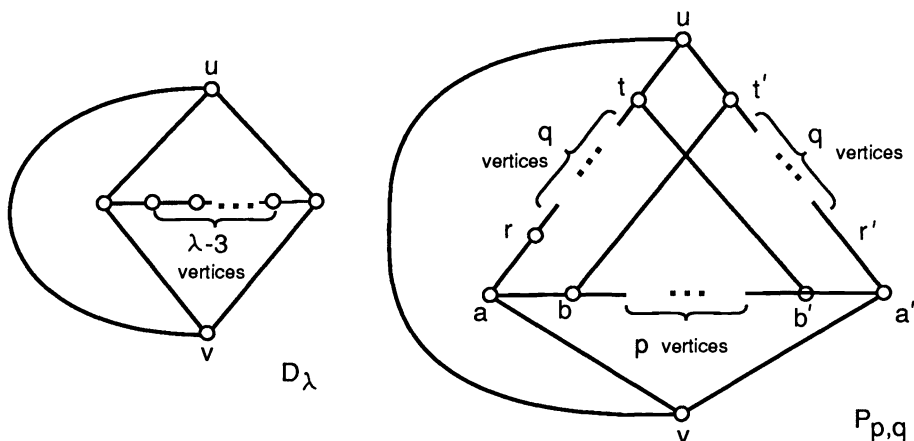


FIG. 7. The graphs D_λ and $P_{p,q}$.

When L^- is finite, $L^- \not\subseteq \{3, 4\}$, then letting $k = \max L^-$ and $n = k + 4$ implies that $k = p + q + 5$ and $n = p + 2q + 9$ for $q = 0$ and $p = k - 5$. Thus $G = P_{p,q}$ satisfies Theorem 6.

In the remaining case when both L and L^- are infinite, there must also always exist nonnegative integers p and q such that $p + q + 5 \in L^-$ and $p + 2q + 9 \in L$. Otherwise $k \in L^-$, $k \geq 5$, implies $k + 4 \in L^-$, $k + 5 \in L^-$, $\dots$, $2k - 1 \in L^-$. Therefore either some $k \in L^-$ has $k \geq 9$ and $\max L \leq k + 3$, or $\max L^- \leq 8$. $\square$

The fact that a graph G with a special edge uv satisfies Theorem 6 is closely related to a somewhat stronger property of G , which will turn out to be crucial to prove the apparent intractability of restricted two-factor problems. In addition to the notation $G^- = G - uv$, we also introduce here the notation $G^{--} = G - \{u, v\}$. We say that G is an L -clamp (with respect to $uv \in E(G)$), if $G - u$ and $G - v$ have L -restricted two-factors but G^- , G^{--} , and all G^* do not. It is easy to see that when L is finite and $\lambda = \max L$, the graph $G = D_\lambda$ is actually an L -clamp. Interestingly, it is also true that each graph $G = P_{p,q}$ with $p \geq 1$ is an L -clamp, provided $p + q + 5 \in L^-$ and $p + 2q + 9 \in L$. However, this fails when $p = 0$, because of the cycle $t \cdots rabt' \cdots r'a'b't$, which is an L -restricted two-factor of G^{--} . Nevertheless, the following general construction (Lemma 7) shows that the existence of a graph G satisfying Theorem 6 (i.e., a counterexample to Theorem 5) implies the existence of a clamp.

The graph H is said to be a *modular extension* of G^- if H contains G^- as an induced subgraph and no vertex of G^{--} is adjacent to a vertex of $H - G^-$. The condition that no G^* has an L -restricted two-factor is equivalent to the assertion that every L -restricted two-factor F of any modular extension H of G^- has each cycle entirely contained in G^- or in $H - G^-$. We say that G^- is L -coherent if neither G^- nor any G^* has an L -restricted two-factor, i.e., if in every L -restricted two-factor F of any modular extension H of G^- , the union of the cycles of F that are included in G^- misses at least one of the vertices u, v .

LEMMA 7. For any $L \subseteq \{3, 4, \dots\}$ the following three statements are equivalent:

- (1) If $G - u$ and $G - v$ have L -restricted two-factors, then G^- or some G^* has an L -restricted two-factor.
- (2) If $G - u$ and $G - v$ have L -restricted two-factors, then G^- or G^{--} or some G^* has an L -restricted two-factor.
- (3) If $G - u$ and $G - v$ have L -restricted two-factors, then (both G^- and G^{--}) or (some G^*) has an L -restricted two-factor.

In other words, for any L either all (1)–(3) hold or none holds.

Proof. If L is finite then the graph D_λ (with $\lambda = \max L$) shows that all three statements are false; so we may assume that L is infinite.

Clearly (3) implies (1) and (1) implies (2). To prove that (2) implies (1), suppose that G is a counterexample to (1), i.e., a graph G with edge uv such that $G - u$ and $G - v$ have L -restricted two-factors, but G^- and all G^* do not. Thus G^- is L -coherent. Construct a graph H as illustrated in Fig. 8.

Since L is infinite, we may choose the value of $m \geq 1$ so that $m + 5 \in L$. We now claim that H is a counterexample to (2). Indeed, $H - u$ has an obvious L -restricted two-factor consisting of L -restricted two-factors of $G_1^- - v_1$ and $G_2^- - v_2$ together with the lower $(m + 5)$ -cycle of H^- containing the unnamed central vertices x, y , and v_1, v_2 , and v . Symmetrically, there is an L -restricted two-factor in $H - v$. Let F be any L -restricted two-factor of any H^* or of H^- or H^{--} . Since all of these graphs are modular extensions of G_1^- (and of G_2^-), the cycles of F included in G_1^- (respectively, in G_2^-) must miss at least one of u_1 or v_1 (respectively, u_2 or v_2). But, since the path $x \cdots y$ must belong to F , exactly one of the edges xu_1, xv_2 (exactly one of the edges yu_2, yv_1 , respec-

tively) belongs to F , and hence only one of u_1 and v_2 (respectively, only one of u_2 and v_1) can be missed by the cycles of F included in G_1^- (respectively, in G_2^-). Thus F contains either the cycle $xu_1uu_2y \cdots x$ or the cycle $xv_2vv_1y \cdots x$, missing either v or u , a contradiction.

To prove that (1) implies (3) we assume that $G - u$ and $G - v$ have L -restricted two-factors but each G^* does not. Then we construct H again as in Fig. 8. Since both $H - u$ and $H - v$ have L -restricted two-factors (cf. above), (1) implies that H^- or some H^* also has an L -restricted two-factor. It is not hard to see that the former case is impossible, and that in the latter case both G^- and G^{--} must have L -restricted two-factors. $\square$

The equivalence of (1) and (2) is crucial to our constructions. We remark on (3) only because it seems to come out of the same construction. In fact, exchanging the names u_2 and v_1 in H^- (cf. Fig. 8), and identifying as before, we obtain a new graph H' . Arguing in H' in the same vein as we did in H we conclude that (1) also implies the following:

(4) If both G^- and G^{--} have L -restricted two-factors, then some G^* has an L -restricted two-factor, or both $G - u$ and $G - v$ have L -restricted two-factors.

This is worth restating for the case $L^- \subseteq \{3, 4\}$, when we know (1) holds. Note that this includes the case of unrestricted two-factors.

COROLLARY 8. *If $L^- \subseteq \{3, 4\}$, and if no G^* has an L -restricted two-factor, then $G - u$ and $G - v$ have L -restricted two-factors if and only if G^- and G^{--} have L -restricted two-factors.* $\square$

We have now shown that if $L^- \not\subseteq \{3, 4\}$, then there is an L -clamp; we shall denote one such L -clamp by G_L . We can now show our last construction (see Fig. 9).

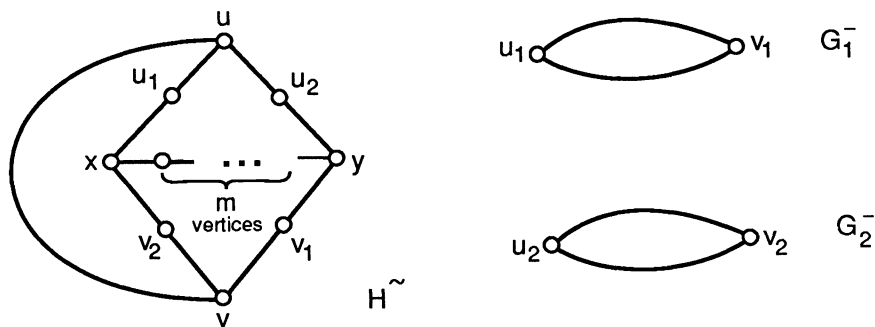


FIG. 8. The graph H is obtained from H^- and G_1^- , G_2^- by identifying all pairs of vertices with equal labels.

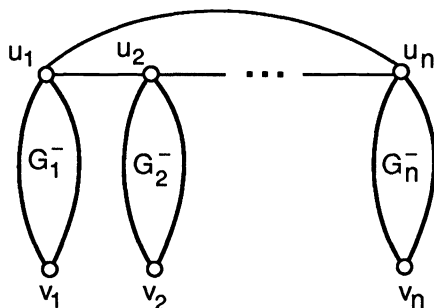


FIG. 9. The graph H_L .

The graph H_L consists of n copies of G_L^- , labeled G_i^- ($i = 1, 2, \dots, n$), with the vertices u_i joined in a cycle of length n . The value of n is chosen so that $n \in L$. Note that H_L is a modular extension of each G_i^- .

LEMMA 9. *Let $L^- \not\subseteq \{3, 4\}$, and let H_L be an induced subgraph of some graph H in which no vertex of $H_L - \{v_1, v_2, \dots, v_n\}$ is adjacent to any vertex of $H - H_L$. If F is an L -restricted two-factor of H then every cycle C of F containing a vertex of $H_L - \{v_1, v_2, \dots, v_n\}$ lies entirely inside H_L . Moreover, if one v_i lies in a cycle contained in H_L then all do.*

Proof. Since each G_i^- is L -coherent and H is a modular extension of G_i^- , the union of those cycles of F that are included in G_i^- contains at most one of the vertices u_i and v_i . Since G_i^- does not have an L -restricted two-factor, the union of these cycles must miss precisely one of u_i and v_i . Suppose v_i belongs to a cycle C of F that is included in H_L . Then by the coherence of G_i^- , C is included in G_i^- . Hence u_i is missed by all the cycles of F that are included in G_i^- , so the cycle $u_1 u_2 \dots u_n u_1$ must belong to F ; i.e., each u_j must be missed by the cycles of F belonging to G_j^- . Therefore all v_j must belong to cycles of F contained in H_L . $\square$

We also note that the graph H_L has an L -restricted two-factor: it is enough to combine the cycle $u_1 u_2 \dots u_n u_1$ with L -restricted two-factors of all $G_i^- - u_i$. Moreover, the graph $H_L - \{v_1, v_2, \dots, v_n\}$ also has an L -restricted two-factor consisting of the corresponding factors of all $G_i^- - v_i$. These properties, together with Lemma 9, will be sufficient to ensure that, when $L^- \not\subseteq \{3, 4\}$, the L -restricted two-factor problem is NP-hard.

Since L is a fixed set of lengths, determining if, say, the graph $G = C_n$ has an L -restricted two-factor amounts to testing the membership of n in L . Thus we cannot in general conclude that these problems are in NP, unless the description of L is easily given, e.g., for finite sets L . (In fact, some of these problems are undecidable [10]; cf. [12].) This is the reason we only aim to prove the problems are NP-hard. Notice that these results do translate to NP-completeness results for all the cases when testing membership in L is in NP (e.g., finite L or finite L^-). (An alternative approach, cf. [5], would be to consider the permitted cycles, viewed as subgraphs, part of the input; then all these problems are indeed in NP.)

THEOREM 10. *When $L^- \not\subseteq \{3, 4\}$ the L -restricted two-factor problem is NP-hard.*

Proof. Let $n \in L$. Since $n \geq 3$, the n -dimensional matching problem is NP-hard. In that problem we are given a disjoint union of n sets, $V = V_1 \cup V_2 \cup \dots \cup V_n$ and a collection E of subsets $e \subseteq V$ with the property that each $e \cap V_i$ has exactly one element; we are asked whether there is a subcollection $E' \subseteq E$ of the e 's that partition V . Given an instance of the n -dimensional matching problem we take a copy H_e of H_L for every $e \in E$, identifying vertex v_i in H_e with the unique element of $e \cap V_i$. It follows from the properties of H_L explained above that the resulting graph has an L -restricted two-factor if and only if the original problem has an n -dimensional matching (cf. [14], [15], or [9, p. 68], for similar arguments). $\square$

4. Conclusions and future directions. When $L^- \subseteq \{3, 4\}$ we have shown that L -restricted two-factors of $G - u$ and $G - v$ may be used to find an L -restricted two-factor of some G^* or of G^- (and G^{--}). In fact, the proofs we have given can be interpreted as linear-time algorithms for finding such factors. (The *existence* of such factors can be proved even for infinite graphs.) When $L^- \not\subseteq \{3, 4\}$ we have constructed examples G having L -restricted two-factors in $G - u$ and $G - v$, but none in any G^* or G^- (and G^{--}). These were then used to show that the corresponding L -restricted two-factor problems are NP-hard. Thus we can show the apparent intractability of all L -restricted two-factor problems, except for the following four cases:

- (1) $L^- = \emptyset$;
- (2) $L^- = \{3\}$;
- (3) $L^- = \{4\}$;
- (4) $L^- = \{3, 4\}$.

It is well known that the unrestricted two-factor problem (1) is in P , [7]. Recently, Cornuejols, Hartvigsen, and Pulleyblank [3] have shown that the triangle-free two-factor problem (2) is also in P . We take our positive results to be an indication that the remaining two L -restricted two-factor problems (3) and (4) should also admit polynomial algorithms. In a future paper we shall offer further evidence of this by proving an augmenting path theorem for each of the cases (1)–(4). In fact, these alternating “paths” are trails, very much like our alternating u -trails, satisfying the appropriate subset of (i)–(xii), except that the straight edges (here representing edges of a particular two-factor F_v of $G - v$) can be any edges of $G - F_u$. At this time we do not know how to find these augmenting trails in polynomial time.

REFERENCES

- [1] H. B. BELCK, *Reguläre Factoren von Graphen*, J. Reine Angew. Math., 188 (1950), pp. 228–252.
- [2] C. BERGE, *Sur le couplage maximum d'un graphe*, C.R. Acad. Sci. Paris Ser. 1. Math., 247 (1958), pp. 258–259.
- [3] G. CORNUEJOLS, D. HARTVIGSEN, AND W. PULLEYBLANK, *Triangle-free two-factors*, personal communication, 1986.
- [4] G. CORNUEJOLS AND W. PULLEYBLANK, *A matching problem with side conditions*, Discrete Math., 29 (1980), pp. 135–159.
- [5] G. CORNUEJOLS, D. HARTVIGSEN, AND W. PULLEYBLANK, *Packing subgraphs in a graph*, Oper. Res. Lett., 4 (1982), pp. 139–143.
- [6] J. EDMONDS, *Paths, trees, and flowers*, Canad. J. Math., 17 (1965), pp. 449–467.
- [7] J. EDMONDS AND E. L. JOHNSON, *Matching: a well-solved class of integer linear programs*, in Combinatorial Structures and Their Applications, R. Guy, H. Hanani, N. Sauer, and J. Schönheim, eds., Gordon and Breach, New York, 1970, pp. 89–92.
- [8] P. FRAISSE, P. HELL, AND D. KIRKPATRICK, *A note on f -factors in directed and undirected graphs*, Graphs and Combinatorics, 2 (1986), pp. 61–66.
- [9] M. GAREY AND D. S. JOHNSON, *Computers and Intractability*, W. H. Freeman, San Francisco, 1979.
- [10] J. E. HOPCROFT AND J. D. ULLMAN, *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley, Reading, MA, 1979.
- [11] P. HELL AND D. KIRKPATRICK, *Packing by cliques and by finite families of graphs*, Discrete Math., 49 (1984), pp. 118–133.
- [12] ———, *Packings by complete bipartite graphs*, SIAM J. Algebraic Discrete Methods, 7 (1986), pp. 199–209.
- [13] P. KATERINIS, *Some conditions for the existence of f -factors*, J. Graph Theory, 9 (1985), pp. 513–521.
- [14] D. KIRKPATRICK AND P. HELL, *On the completeness of a generalized matching problem*, in Proc. 10th Annual ACM Symposium on Theory of Computing, San Diego, CA, 1978, pp. 240–245.
- [15] ———, *On the complexity of general graph factor problems*, SIAM J. Comput., 12 (1983), pp. 601–609.
- [16] O. VORNBERGER, *Komplexität von Wegeproblemen in Graphen*, Reihe Theoretische Informatik, Universität Paderborn, 5 (1979).
- [17] W. TUTTE, *The factors of graphs*, Canad. J. Math., 4 (1952), pp. 314–328.

SILVERMAN GAMES ON DISJOINT DISCRETE SETS*

G. A. HEUER† AND W. DOW RIEDER‡

Abstract. Previous work by Heuer on Silverman games on odd versus even positive integers with threshold $T > 1$ and penalty $\nu > 0$, is extended to games on more general strategy sets S_1 and S_2 of positive numbers. Much of this paper is devoted to games where S_1 and S_2 are disjoint and discrete. Results include complete classification and solutions for $T \leq T^*$ (where T^* depends on S_1 and S_2), complete classification and solutions for $T > T^*$ and $\nu \geq 1$, and partial classification with solutions for $T > T^*$ and $\nu < 1$.

Key words. Silverman games, two-person zero sum games

AMS(MOS) subject classification. 90D05

1. Introduction. A Silverman game is a two-person zero sum game defined in terms of two sets of positive numbers, S_1 and S_2 , and two parameters, the threshold $T > 1$ and the penalty $\nu > 0$. Players I and II choose numbers from S_1 and S_2 , respectively. The higher number wins one, unless it is at least T times as large as the other, in which case it loses ν . If the numbers are equal the payoff is zero.

Two prototypes are attributed to David Silverman. In one, $S_1 = S_2 = [1, \infty)$ and $\nu = 1$; in the other $S_1 = S_2 = \mathbb{N}$, the set of positive integers, $T = 3$ and $\nu = 1$. In the second type the unique optimal strategy for each player is to choose each of one, two, and five with probability $\frac{1}{3}$. The symmetric cases ($S_1 = S_2 = S$) have been investigated in some generality by R. J. Evans. In the presence of this symmetry, the game value, if it exists, must be zero, and the two players have the same optimal strategies. For the case where S is an arbitrary discrete set (no cluster points) of positive numbers and ν exceeds a certain constant $\nu_0 < 1$, Evans [1] has exhibited an optimal strategy and has shown its uniqueness. In [2] Evans has examined the case where S is an open interval (a, b) with $a > 0$, and has found that an optimal strategy exists if and only if for some integer $n \geq 2$, $\nu = \sec(\pi/(n+1)) - 1$ and $T^{n-1} < b/a \leq \infty$. Again the optimal strategy is given explicitly. A somewhat similar game has been discussed by Mendelsohn [5].

An analogous game where $S = [a, \infty)$, $a > 0$, has been analyzed by Heuer [4]. Here the payoff, instead of being a step function of the ratio of the two numbers, increases linearly from zero as the ratio increases from one, until the payoff reaches one, after which it decreases linearly with increasing ratio. Again there are two parameters, and for most of their values solutions (all of finite type) have been obtained.

Nonsymmetric Silverman games were first examined by Heuer [3], where S_1 and S_2 were taken to be the odd and even positive integers, respectively. Solutions of finite type, usually not unique, were constructed for all values of T and ν . In this paper we extend that work to much more general disjoint sets S_1 and S_2 . For the case where $S_1 \cup S_2$ has a least element and where for each x in S_i ($i = 1, 2$) there is a least $y > x$ in S_{3-i} , we give a complete classification with solutions when $\nu \geq 1$, and partial results for $\nu < 1$. Of course this includes the situation where S_1 and S_2 are discrete. Some of the theorems do not require S_1 and S_2 to be disjoint.

* Received by the editors June 3, 1987; accepted for publication (in revised form) May 5, 1988. This work was supported in part by a grant from the Bush Foundation, St. Paul, Minnesota.

† Department of Mathematics and Computer Science, Concordia College, Moorhead, Minnesota 56560.

‡ Concordia College, Moorhead, Minnesota 56560. Present address, Mathematics Department, North Dakota State University, Fargo, North Dakota 58105.

2. Preview of principal results. We assume throughout the paper that S_1 and S_2 are subsets of $(0, \infty)$ with $0 < c_1 = \inf S_1 \neq d_1 = \inf S_2$. Our convention will be that $c_1 < d_1$. This obviously entails no loss of generality. The parameters T and ν always satisfy $T > 1$ and $\nu > 0$. The payoff function A for a Silverman game on $S_1 \times S_2$ is defined by

$$(2.0.1) \quad A(x, y) = \begin{cases} \nu & \text{if } x \leq y/T, \\ -1 & \text{if } y/T < x < y, \\ 0 & \text{if } x = y, \\ 1 & \text{if } y < x < Ty, \\ -\nu & \text{if } Ty \leq x. \end{cases}$$

Although the title of this paper refers to discrete sets S_1 and S_2 , the results are somewhat more generally applicable. If $c_1 \in S_1$ and for each element of S_i ($i = 1, 2$) there is a least element of S_{3-i} exceeding it, the game may be reduced by dominance to one where the S_i are not only discrete but are interlacing sequences, as was the case with odd versus even integers. We show this in § 5.

For $\nu \geq 1$ the games fall into eight classes. There are eight functions $H_1, \pm H_2, \pm H_3, \pm H_4$, and H_5 such that every Silverman game with S_1 and S_2 as described in the preceding paragraph has, for given T , game value $V = H_i(\nu)$, $1 \leq i \leq 5$, or $V = -H_i(\nu)$, $2 \leq i \leq 4$; $1 \leq \nu < \infty$. For a given S_1 and S_2 the class and game value are left-continuous step functions of T . More precisely, the T -axis, $T > 1$, is partitioned into a union of left open, right closed intervals $(a_i, b_i]$ such that for $a_i < T \leq b_i$ the game value is independent of T and is given by only one of the eight functions of ν . Actually the functions H_4 and H_5 involve an additional parameter k , which also remains fixed for $a_i < T \leq b_i$. For each interval $(a_i, b_i]$ there are finite probability vectors $P = (p_1, \dots, p_{n(i)})$ and $Q = (q_1, \dots, q_{n(i)})$, independent of T , and pure strategies $s_1, \dots, s_{n(i)}$ in S_1 and $t_1, \dots, t_{n(i)}$ in S_2 (these are dependent on T) such that the mixed strategies assigning probability p_j to s_j and q_j to t_j for $j = 1, \dots, n(i)$ are optimal for all T in $(a_i, b_i]$. Our theorems show how to determine the class of a game, given S_1, S_2 , and T , and give solutions for all eight classes.

The situation for $0 < \nu < 1$ seems to be more complicated, as Evans has found to be true also in the symmetric case, and our results are incomplete. Five classes occur that are natural extensions of five of the above eight classes. For each of them we give conditions under which they do occur and give the solution, but this classification is incomplete.

Here is a brief description of each of the eight classes. In classes 1A and 2A, Player I has an optimal pure strategy, and the game value is $H_1(\nu) = \nu$ and $H_2(\nu) = \min \{1, \nu\}$, respectively. In class 2B, Player II has an optimal pure strategy and the value is $-H_2(\nu)$. Our theorems in this paper take care of all $\nu > 0$. In class 3A for $\nu \geq 1$, the game value is $H_3(\nu) = (\nu - 1)/(\nu + 3) = (1 - \alpha)/(1 + \alpha)$ where $\alpha = 2/(\nu + 1)$ and each player has a "two-point" optimal strategy, i.e., one which is a mix of two pure strategies. The case of the odd versus even integers [3] falls into this class when $T > 2$, and we give conditions under which the solution for $0 < \nu < 1$ is like that case. This seems to be the natural extension of class 3A to the range $0 < \nu < 1$. Class 3B is like 3A with the players' roles reversed, and the value is $-H_3(\nu)$. (S_1 still has $c_1 = \min (S_1 \cup S_2)$ but it is not involved in an optimal mixed strategy.)

In the remaining classes the additional integer $k \geq 1$ occurs as a parameter, and no extension to $0 < \nu < 1$ is suggested at this point. In class 4A. k the game value is $H_4(\nu) = \sum_{j=0}^{2k} (-\alpha)^j / \sum_{j=0}^{2k} \alpha^j$, $\nu \geq 1$, and each player has a $(2k + 1)$ -point optimal

strategy. Class 4B. k is the same, with roles reversed, and the value is $-H_4(\nu)$, $\nu \geq 1$. In class 5A. k , game value is $H_5(\nu) = (1 - \alpha + \alpha^2 - \cdots + \alpha^{2k-2} + \alpha^{2k-1} - \alpha^{2k}) / \sum_{j=0}^{2k} \alpha^j$ and again each player has a $(2k + 1)$ -point optimal strategy.

It will be useful to observe the equivalence of certain Silverman games. For this purpose here, and occasionally elsewhere, we denote a Silverman game by the ordered quadruple (S_1, S_2, T, ν) . By aS and S^p we mean, respectively, the sets $\{as: s \in S\}$ and $\{s^p: s \in S\}$.

THEOREM 2.1. *Given $S_1, S_2, T, \nu, a > 0$, and $p > 0$, the Silverman games (S_1, S_2, T, ν) , (aS_1, aS_2, T, ν) , and (S_1^p, S_2^p, T^p, ν) are equivalent.*

Proof. The mapping $(x, y) \mapsto (ax, ay)$ from $S_1 \times S_2$ to $aS_1 \times aS_2$ is a bijection, and we see at once from (2.0.1) that for given T and ν , $A(x, y) = A(ax, ay)$ for all (x, y) in $S_1 \times S_2$. Thus (S_1, S_2, T, ν) and (aS_1, aS_2, T, ν) are equivalent games.

The mapping $(x, y) \mapsto (x^p, y^p)$ from $S_1 \times S_2$ to $S_1^p \times S_2^p$ likewise is a bijection, and (now using $A(x, y, T, \nu)$ for $A(x, y)$) we have $A(x, y, T, \nu) = A(x^p, y^p, T^p, \nu)$ so (S_1, S_2, T, ν) and (S_1^p, S_2^p, T^p, ν) are equivalent. $\square$

3. Class 1A. Recall that our convention is $\inf S_1 < \inf S_2$, i.e., Player I has some numbers available in S_1 that are smaller than any numbers in S_2 available to Player II. If c is such a number and T is such that every element of S_2 is at least as large as cT , Player I always wins ν by choosing c . If $\nu \geq 1$, ν is the largest payoff that can occur under any circumstances, and obviously a strategy that wins ν is optimal. We shall see that such a strategy is optimal even when $\nu < 1$. The first two theorems give precise formulations. For these Class 1A cases (as well as for Classes 2A and 2B) we have no need to assume that either S_1 and S_2 are discrete or they are disjoint.

We shall always denote the payoff function by A . Thus $A(x, y)$ is the payoff to Player I when pure strategies x and y are employed by Players I and II, respectively. If P and Q are strategies for Players I and II, respectively, and at least one of them is not a pure strategy, then we denote the expected payoff by $E(P, Q)$. V always denotes game value.

THEOREM 3.1 (Class 1A, $\nu \geq 1$). *Let $T \leq d_1/c_1$ and $\nu \geq 1$. Assume that $S_1 \cap (0, d_1/T] \neq \emptyset$. Then the game value $V = \nu$, every mixed strategy defined by a probability distribution on $S_1 \cap (0, d_1/T]$ is optimal for Player I, and all strategies for Player II are optimal. If $\nu > 1$ the above are the only optimal strategies for Player I.*

Proof. Since $\nu \geq 1$, ν is the maximum payoff possible, and consequently $V \leq \nu$. If $c \in S_1$ and $c \leq d_1/T$ then Player I wins ν by choosing c . It follows that $V = \nu$ and every mixed strategy defined by a distribution on $S_1 \cap (0, d_1/T]$ is optimal. Whatever strategy Player II employs, the payoff is at most ν and therefore the strategy is optimal.

Now suppose $\nu > 1$ and $c > d_1/T$. Then there is a number d in S_2 with $d < cT$, so that when c is played against d the payoff is 1, -1 or $-\nu$, all of which are smaller than ν . If Player I assigns a positive probability to c (or to any interval (c, c') with $c > d_1/T$), the expected payoff against pure strategy d is smaller than ν , and Player I's strategy is not optimal. $\square$

THEOREM 3.2 (Class 1A), $\nu < 1$). *Let $T \leq d_1/c_1$ and $0 < \nu < 1$. Assume that $S_1 \cap (0, d_1/T] \neq \emptyset$ and that S_2 is unbounded. Then the game value $V = \nu$, and every probability distribution on $S_1 \cap (0, d_1/T]$ is optimal for Player I. Player II may fail to have an optimal strategy, but given $\varepsilon > 0$, the following strategy, which we denote by Q_ε , is ε -optimal for Player II: Choose m so that $1 - \nu < m\varepsilon$. Choose any $d^{(1)}$ in S_2 , and when $d^{(k)}$ is determined, choose $d^{(k+1)}$ in S_2 so that $d^{(k+1)} > d^{(k)}T$, $k = 1, \dots, m - 1$. Assign probability $1/m$ to each $d^{(k)}$, $1 \leq k \leq m$.*

If $S_1 \cap [d_1, d_1T) = \emptyset$ and $d_1 \in S_2$, then the only optimal strategies for Player I are those on $S_1 \cap (0, d_1/T]$, and pure strategy d_1 is optimal for Player II.

Proof. It is clear that Player I wins ν with any distribution on $S_1 \cap (0, d_1/T]$, and thus $V \geq \nu$. Let $\varepsilon > 0$ be given. If $c \leq d^{(1)}$, then $A(c, d^{(k)})$ is -1 , zero or ν for each k and therefore $E(c, Q_\varepsilon) \leq \nu$. If $c > d^{(1)}$, then let k be the largest integer such that $c > d^{(k)}$. Then for $1 \leq j \leq k-1$, $A(c, d^{(j)}) = -\nu$, and for $k+1 \leq j \leq m$, $A(c, d^{(j)}) = -1$ or ν , so $A(c, d^{(j)}) \leq \nu$ for all $j \neq k$. $A(c, d^{(k)}) \leq 1$, so $E(c, Q_\varepsilon) \leq [1 + \nu(m-1)]/m = \nu + (1-\nu)/m < \nu + \varepsilon$. Thus $V < \nu + \varepsilon$. It follows that $V = \nu$ and that the strategy Q_ε is ε -optimal.

Now suppose that $S_1 \cap [d_1, d_1 T) = \emptyset$ and $d_1 \in S_2$. If F is any distribution on S_1 with $F(d_1/T) < 1$, then against pure d_1 its expected payoff is

$$E(F, d_1) = \int_{(0, d_1/T]} \nu dF + \int_{(d_1/T, d_1)} (-1) dF + \int_{(d_1, d_1 T)} (1) dF + \int_{[d_1 T, \infty)} (-\nu) dF.$$

Since $(d_1, d_1 T) \cap S_1 = \emptyset$, $\int_{[d_1 T, \infty)} dF = 1 - \int_{(0, d_1 T)} dF$ and $A(d_1, d_1) = 0$, we have

$$\begin{aligned} E(F, d_1) &= 2\nu \int_{(0, d_1/T]} dF - \nu + (\nu - 1) \int_{(d_1/T, d_1)} dF \\ &\leq 2\nu F(d_1/T) - \nu < \nu, \end{aligned}$$

so F is not optimal. To see that pure d_1 is now optimal for Player II, observe that for $c < d_1$, $A(c, d_1)$ is -1 or ν , while for $c \geq d_1 T$, $A(c, d_1) = -\nu$, and thus $A(c, d_1) \leq \nu$ in all cases.

Finally, the following examples show that Player II may fail to have an optimal strategy.

Example 3.3. Let $S_1 = \{c^{3n-2}; n \in \mathbb{N}\}$ and let $S_2 = \{c^{3n}; n \in \mathbb{N}\}$, where $1 < c < T < c^2 = d_1/c_1$. Given any mixed strategy Q on S_2 , let c^{3k} be the smallest pure strategy involved in Q , and p its probability. Then $A(c^{3k+1}, c^{3k}) = 1$ and for all $n > k$, $A(c^{3k+1}, c^{3n}) = \nu$. Thus $E(c^{3k+1}, Q) = p + \nu(1-p) > \nu p + \nu(1-p) = \nu$, so Q is not optimal.

Example 3.4. Let $d > 1 + \sqrt{2}$, $S_1 = \{1\} \cup \{d^n + 1; n \in \mathbb{N}\}$, $S_2 = \{d^n; n \in \mathbb{N}\}$, and $T = d - 1$. Again we have $1 < T < d = d_1/c_1$, and the same argument applies.

4. Classes 2A and 2B. In class 1A we have $d_1 \geq c_1 T$, and c_1 is optimal for Player I. If $d_1 < c_1 T$, Player I may still have an optimal pure strategy $c > c_1$. Consider the example $S_1 = \{1, 3, 5, 17, 19, \dots\}$, $S_2 = \{2, 4, 16, 18, 20, \dots\}$ with $T = 3$. The pure strategy $c^* = 5$ has the property that for every d in S_2 , either $d < c^* < dT$ or $d \geq c^* T$. In such a game $A(c^*, d)$ is one or ν for every d in S_2 , so that $V \geq \min\{1, \nu\}$. In this section we show $V = \min\{1, \nu\}$ and every such c^* is optimal. These games where Player I has an optimal pure strategy $c^* > d_1/T$ constitute our class 2A. As in the preceding section, we make no assumptions yet about discreteness or disjointness of S_1 and S_2 .

THEOREM 4.1 (Class 2A, $\nu \geq 1$). *Suppose that $S_2 = S'_2 \cup S''_2$, where $\sup S'_2 < d_1 T$ and $\inf S''_2 > (\sup S'_2)T$ for some $T > d_1/c_1$. Let*

$$S^*_1 = \{c^* \in S_1; d < c^* < d_1 T \text{ for all } d \text{ in } S'_2 \text{ and } c^* T \leq \inf S''_2\}.$$

Assume that $S^*_1 \neq \emptyset$, and let $\nu \geq 1$.

Then for the Silverman game (S_1, S_2, T, ν) , $V = \min\{1, \nu\} = 1$. Every distribution on S^*_1 is optimal for Player I, and every distribution on $S^*_2 = \{d \in S_2; d < c_1 T\}$ is optimal for Player II.

Proof. It is immediate that if $c^* \in S^*_1$, then $A(c^*, d) = 1$ or ν for every d in S_2 . Thus $V \geq \min\{1, \nu\} = 1$. If $d \in S^*_2$, then $d < cT$ for all c in S_1 so $A(c, d) \leq 1$. It follows that $V = 1$ and that the strategies described in the theorem are optimal. $\square$

Player II may have optimal strategies other than those in S_2^* in the above theorem. In the example preceding the theorem, for instance, $S_2^* = \{2\}$, but the mixed strategy which assigns probability $(\nu - 1)/(\nu + 1)$ to two and $2/(\nu + 1)$ to four is also optimal.

THEOREM 4.2 (Class 2A, $\nu < 1$). *Let S_1 , S_2 , and T be as in Theorem 4.1, with S_2 unbounded above and let $\nu < 1$. Then $V = \min \{1, \nu\} = \nu$ and every distribution on S_1^* is optimal for Player I. Player II may fail to have an optimal strategy, but given $\varepsilon > 0$, the strategy Q_ε described in Theorem 3.2 is ε -optimal.*

Proof. If $c^* \in S_1^*$, then $A(c^*, d)$ is one or ν for every d in S_2 , so $V \geq \nu$. The proof that $E(c, Q_\varepsilon) < \nu + \varepsilon$ for every c in S_1 is exactly as in Theorem 3.2, and it follows that $V = \nu$, Q_ε is ε -optimal, and all distributions on S_1^* are optimal.

The following example shows that Player II may fail to have an optimal strategy.

Example 4.3. Let $1 < c < T < c^2$, $S_1 = \{c^2\} \cup \{c^{3n+1} : n \in \mathbb{N}\}$, and $S_2 = \{c^{3n} : n \in \mathbb{N}\}$. Then our hypotheses are satisfied with $S_2' = \{c^3\}$ and $S_1^* = \{c^4\}$. Given any strategy Q on S_2 , let c^{3k} be the smallest pure strategy to which it assigns positive probability. Then $E(c^{3k+1}, Q) > \nu$ exactly as in Example 3.3.

Class 2B occurs when Player II has one or more pure strategies d^* with $A(c, d^*) = -1$ or $-\nu$ for every c in S_1 . As an example, let $S_1 = \{1, 3, 21, 23, 25, \dots\}$ and $S_2 = \{2, 4, 22, 24, 26, \dots\}$, $T = 5$, and $d^* = 4$. The situation is similar to that in class 2A, but with Player II having the advantage. The theorems are very much mirror images of those above and may be proved by making obvious modifications in the proofs of Theorems 4.1 and 4.2.

THEOREM 4.4 (Class 2B, $\nu \geq 1$). *Suppose that $S_1 = S_1' \cup S_1''$, where $\sup S_1' < c_1^T$ and $\inf S_1'' > (\sup S_1')T$ for some $T > 1$, and assume that*

$$S_2^* := \{d^* \in S_2 : c < d^* < c_1 T \text{ for all } c \in S_1' \text{ and } d^* T \leq \inf S_1''\} \neq \emptyset.$$

Let $\nu \geq 1$.

Then $V = \max \{-1, -\nu\} = -1$. Every distribution on S_1' is optimal for Player I, and every distribution on S_2^* is optimal for Player II. If $\nu > 1$, there are no other optimal strategies for Player I.

THEOREM 4.5 (Class 2B, $\nu < 1$). *Let S_1 , S_2 , and T be as in Theorem 4.4 with S_1 unbounded above, and $\nu < 1$. Then $V = \max \{-1, -\nu\} = -\nu$ and every distribution on S_2^* is optimal for Player II. Player I may fail to have an optimal strategy, but if $\varepsilon > 0$ the following strategy P_ε is ε -optimal for Player I: Choose m such that $1 - \nu < m\varepsilon$. Choose any $c^{(1)}$ in S_1 , and for $1 \leq k \leq m - 1$ when $c^{(k)}$ has been chosen, choose $c^{(k+1)}$ in S_1 with $c^{(k+1)} > c^{(k)}T$. Assign each $c^{(k)}$ probability $1/m$.*

The following example shows that Player I may fail to have an optimal strategy.

Example 4.6. Let $1 < c < T < c^2$, $S_1 = \{c^{3n} : n \in \mathbb{N}\}$, and $S_2 = \{c^{3n+1} : n \in \mathbb{N}\}$. Then our hypotheses are satisfied with $S_1' = \{c^3\}$ and $S_2^* = \{c^4\}$. Given any strategy P on S_1 , let c^{3k} be the smallest element of S_1 assigned positive probability. Then, as in Example 3.3, we see that $E(P, c^{3k+1}) < -\nu$, so P is not optimal.

5. Reduction to interlacing sequences. We shall assume hereafter that $S_1 \cap S_2 = \emptyset$.

THEOREM 5.1. *Let $c_1 = \inf S_1$ and $d_1 = \inf S_2$ as usual. For $n \geq 1$, let $c_{n+1} = \inf (S_1 \cap (d_n, \infty))$ and $d_n = \inf (S_2 \cap (c_n, \infty))$. If every $c_i \in S_1$ and every $d_i \in S_2$, then the Silverman game on $S_1 \times S_2$ reduces by dominance to that on $S_1' \times S_2'$, where $S_1' = \{c_1, c_2, c_3, \dots\}$ and $S_2' = \{d_1, d_2, d_3, \dots\}$. (We note that these sequences interlace: $c_i < d_i < c_{i+1}$ for each i . Note also that the hypotheses are always satisfied when S_1 and S_2 are discrete sets.)*

Proof. Consider any elements c and c' of S_1 with $c < c'$ and no elements of S_2 in (c, c') . We show that strategy c dominates c' , i.e., $A(c, d) \geq A(c', d)$ for all d in S_2 .

First suppose $d > c'$. If $d < cT$ then $d < c'T$ also, and $A(c, d) = A(c', d) = 1$. If $cT \leq d$ then $A(c, d) = \nu$, while $A(c', d)$ is ν or -1 . Now suppose $d < c'$. If $c'/T < d$, then $A(c', d) = A(c, d) = 1$. If $d \leq c'/T$, then $A(c', d) = -\nu$, while $A(c, d) = -\nu$ or one. In all cases $A(c, d) \geq A(c', d)$.

If c is an element of S_1 with $c_i < c < c_{i+1}$ for some i , then by definition of c_{i+1} , $c < d_i$, so there are no elements of S_2 in (c_i, c) , and the preceding paragraph shows that c_i dominates c . A similar argument shows that every element of $S_2 \setminus S'_2$ is dominated by an element of S'_2 . $\square$

Example 5.2. Let $S_1 = \bigcup_{i \in \mathbb{N}} S_{i1}$, where S_{i1} is any subset of $[2i - 1, 2i)$ containing $2i - 1$. Let $S_2 = \bigcup_{i \in \mathbb{N}} S_{i2}$, where S_{i2} is any subset of $[2i, 2i + 1)$ containing $2i$. Then the Silverman game on $S_1 \times S_2$ reduces by dominance to that on odd versus even integers.

6. Class 3A. For the remainder of the paper we shall assume that S_1 and S_2 are unbounded interlacing sequences:

$$(6.0.1) \quad S_1 = \{c_1, c_2, c_3, \dots\}, \quad S_2 = \{d_1, d_2, d_3, \dots\},$$

where $0 < c_i < d_i < c_{i+1}$ for each i and $\lim_{n \rightarrow \infty} c_n = \infty$.

The following notation will be useful:

$$(6.0.2) \quad \text{Given } S_1 \text{ and } S_2 \text{ and positive integers } m, n,$$

let $J_{m,n}$ denote the interval $(c_m/d_n, d_m/c_n]$ and $T^* = \max \{d_1/c_1, c_2/d_1\}$.

For given S_1, S_2 , and $T > d_1/c_1$, define sequences $\{r_j\}$ and $\{s_i\}$ by

$$(6.0.3) \quad c_{r_j} < d_j T \leq c_{r_j+1},$$

$$(6.0.4) \quad d_{s_i} < c_i T \leq d_{s_i+1}.$$

This means that in column j of the payoff matrix the entry changes from 1 or -1 or ν to $-\nu$ at rows r_j and $r_j + 1$, and in row i , the payoff changes from -1 or 1 or $-\nu$ to ν at columns s_i and $s_i + 1$.

The mixed strategy P for Player I that assigns p_i to c_i for $i = 1, 2, \dots$ will be denoted by $P = (p_1, p_2, p_3, \dots)$, and similarly $Q = (q_1, q_2, q_3, \dots)$ denotes the mixed strategy for Player II, which for each i assigns probability q_i to d_i .

We shall make frequent use of the following lemma, which reduces the infinite series $E(P, d_s) = \sum_{i=1}^{\infty} A(c_i, d_s) p_i$ and $E(c_r, Q) = \sum_{i=1}^{\infty} A(c_r, d_i) q_i$ to finite sums.

LEMMA 6.1. For all mixed strategies P and Q and pure strategies c_r, d_s ,

$$(6.1.1) \quad E(P, d_s) = 2\nu \sum \{p_i: c_i \leq d_s/T\} + (\nu - 1) \sum \{p_i: d_s/T < c_i < d_s\} \\ + (\nu + 1) \sum \{p_i: d_s < c_i < d_s T\} - \nu, \quad \text{and}$$

$$(6.1.2) \quad E(c_r, Q) = -2\nu \sum \{q_i: d_i \leq c_r/T\} - (\nu - 1) \sum \{q_i: c_r/T < d_i < c_r\} \\ - (\nu + 1) \sum \{q_i: c_r < d_i < c_r T\} + \nu.$$

Proof. By definition of the payoff function A in (2.0.1),

$$E(P, d_s) = \nu \sum \{p_i: c_i \leq d_s/T\} - \sum \{p_i: d_s/T < c_i < d_s\} + \sum \{p_i: d_s < c_i < d_s T\} \\ - \nu \sum \{p_i: d_s T \leq c_i\} = \nu \sum_1 - \sum_2 + \sum_3 - \nu \sum_4,$$

say. But $\sum_{i=1}^{\infty} p_i = 1$, and substitution of $1 - \sum_1 - \sum_2 - \sum_3$ for $\sum_4$ yields (6.1.1). Equation (6.1.2) is obtained similarly. $\square$

As stated earlier, class 3A Silverman games have game value $V = H_3(\nu)$, where H_3 is the “bouncing ball” function first described in the odds versus evens game [3]. See Fig. 1. H_3 is a piecewise rational function that increases in degree and complexity from one piece to the next with decreasing ν . The simplest piece extends over the interval $1 \leq \nu < \infty$, where $H_3(\nu) = (\nu - 1)/(\nu + 3)$, a hyperbola with horizontal asymptote at height one. While it is possible to incorporate class 3A games into a single theorem, in doing so the complexity of the situation in the range $0 < \nu < 1$ obscures the simplicity of the result for the vast portion of the range, namely $\nu \geq 1$, and we prefer to state two theorems.

THEOREM 6.2 (Class 3A, $\nu \geq 1$). *Let S_1, S_2, T^* , and $J_{m,n}$ be as described in (6.0.1) and (6.0.2), and let $\nu \geq 1$. Assume we have the following:*

$$(6.2.1) \quad T > T^*;$$

$$(6.2.2) \quad \text{For some } m \geq 2, \quad T \in J_{m,1} \setminus J_{m+1,1};$$

$$(6.2.3) \quad d_m < c_2 T.$$

Then $V = (\nu - 1)/(\nu + 3)$. The strategies P and Q with $p_1 = q_k = 2/(\nu + 3)$ and $p_m = q_1 = (\nu + 1)/(\nu + 3)$, where $k = s_2$ is the integer such that $d_k < c_2 T \leq d_{k+1}$, are optimal.

Remarks. (i) If (6.2.1) fails, we are in class 1A ($T \leq d_1/c_1$) or 2B

$$(d_1/c_1 < T \leq c_2/d_1).$$

(ii) If (6.2.2) holds with $m = 2$ and (6.2.3) fails, then the game is in class 2A with a saddle point at (c_2, d_1) . The situation where (6.2.2) holds for some $m > 2$ but (6.2.3) fails is treated in § 8.

(iii) As noted, k in the theorem is the s_2 of (6.0.4). We easily check that m is the r_1 of (6.0.3), and that the conditions (6.2.1)–(6.2.3) are equivalent to the condition $s_1 < r_1 \leq s_2$.

(iv) In subsequent sections we shall use the variable $\alpha = 2/(\nu + 1)$. In these terms, $V = (1 - \alpha)/(1 + \alpha)$, $p_1 = \alpha/(1 + \alpha)$ and $p_m = 1/(1 + \alpha)$.

Proof of Theorem 6.2. We show first that $E(P, d_s) \geq (\nu - 1)/(\nu + 3)$ for all s . If $1 \leq s \leq m - 1$, then $d_s < c_m < d_1 T$ by (6.2.2), so p_m is in the third sum in (6.1.1). Since p_1 is in one of those sums, $E(P, d_s) \geq (\nu - 1)p_1 + (\nu + 1)p_m - \nu = (\nu - 1)/(\nu + 3)$. If $s \geq m$, then $d_s \geq d_m \geq c_1 T$ by (6.2.2), so p_1 is in the first sum in (6.1.1). Also, $c_m \leq c_s < d_s$, so p_m is in one of the first two sums and therefore $E(P, d_s) \geq (2\nu)2/(\nu + 3) + (\nu^2 - 1)/(\nu + 3) - \nu = (\nu - 1)/(\nu + 3)$.

We complete the proof by showing that $E(c_r, Q) \leq (\nu - 1)/(\nu + 3)$ for all r . First consider $r = 1$. By (6.2.1), $c_1 < d_1 < c_1 T$, so q_1 is in the third sum in (6.1.2), and therefore $E(c_1, Q) \leq -(\nu + 1)^2/(\nu + 3) + \nu = (\nu - 1)/(\nu + 3)$. Next, if $2 \leq r \leq k$, then $c_r < d_k < c_2 T$ by definition of k , so q_k is in the third sum in (6.1.2), and q_1 is in one of them; thus $E(c_r, Q) \leq -(\nu - 1)(\nu + 1)/(\nu + 3) - 2(\nu + 1)/(\nu + 3) + \nu = (\nu - 1)/(\nu + 3)$. Finally, suppose $r \geq k + 1$. Note that by definition of k and (6.2.3) we have $k \geq m$. Thus from (6.2.2) we see that $d_1 T \leq c_{m+1} \leq c_{k+1} \leq c_r$, so d_1 is in the first sum. Consequently, $E(c_r, Q) \leq -2\nu(\nu + 1)/(\nu + 3) + \nu \leq (\nu - 1)/(\nu + 3)$. $\square$

The case where S_1 and S_2 are, respectively, the odd and even positive integers, with $T > 1$ and $\nu \geq 1$, which was Theorem 4 of [3], may be seen to be a special case of Theorem 6.2 above. The interval $J_{m,1}$ is $(m - .5, 2m]$, $T^* = 2$, every $T > 2$ is in some $J_{m,1}$, and (6.2.3) holds. A large class of cases where S_2 is a translate of S_1 , including the odds versus evens case, is described in the following corollary.

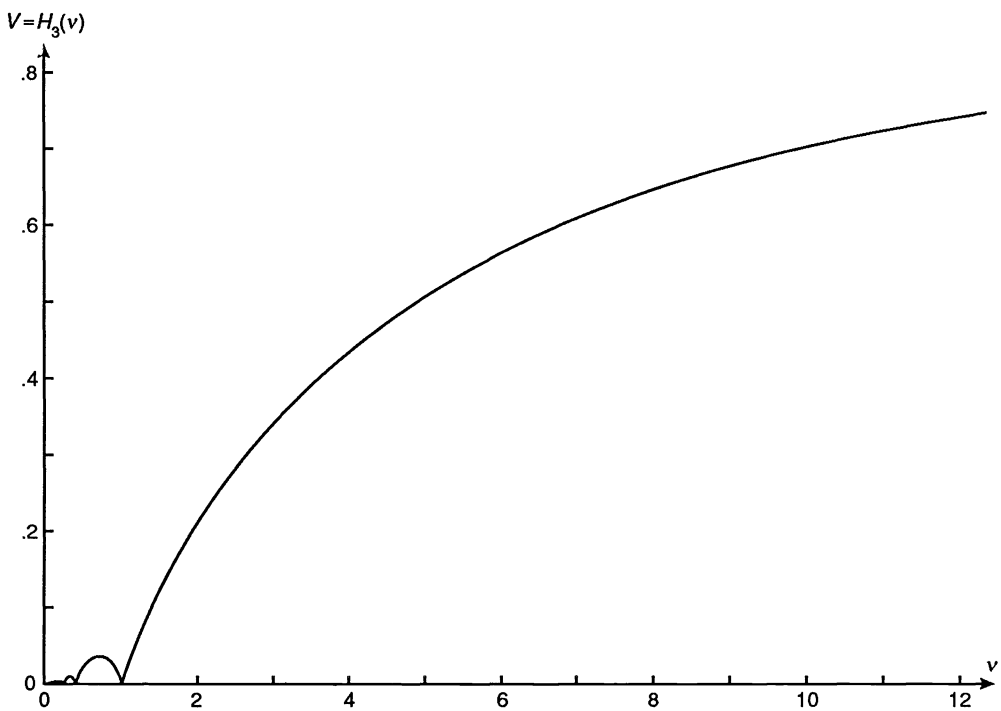


FIG. 1(a)

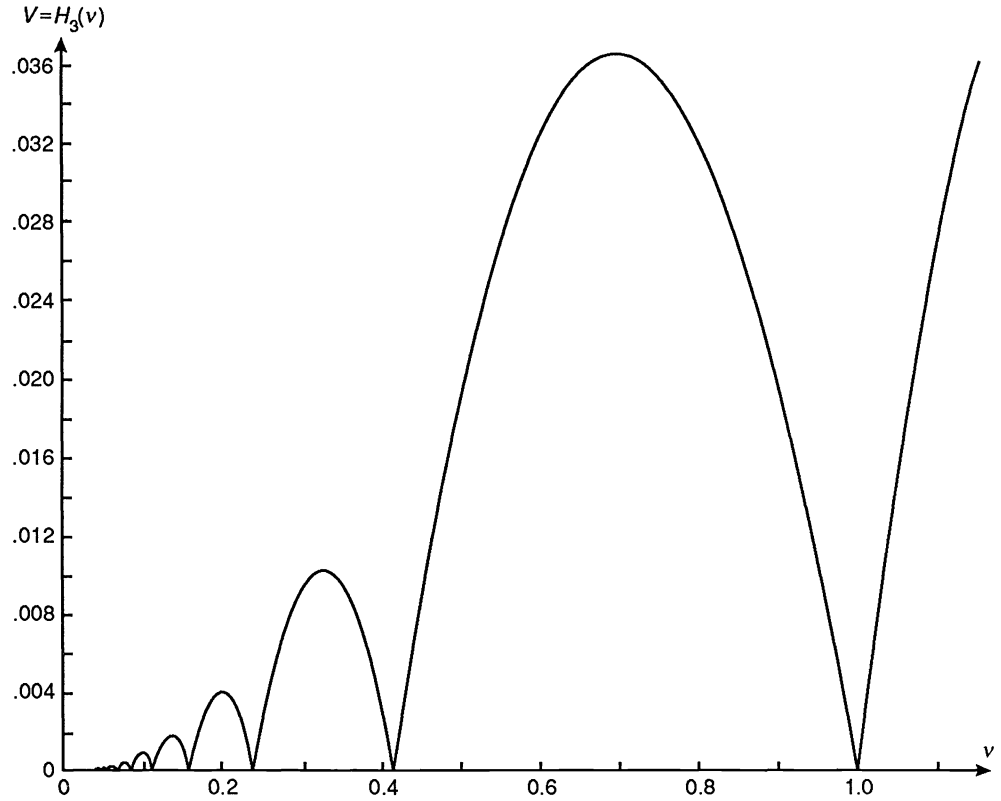


FIG. 1(b)

COROLLARY 6.3. Let $S_1 = \{d_r - a : r \in \mathbb{N}\}$ and $S_2 = \{d_r : r \in \mathbb{N}\}$, where $0 < a < d_1$, $d_2 \geq 2d_1$ and $d_{r+1} > d_r + a$ for each r . Let $T > T^*$, $\nu \geq 1$ and assume that $T \in J_{m,1}$ for some $m \geq 2$.

Then $V = (\nu - 1)/(\nu + 1)$ and the two-point strategies P and Q described in Theorem 6.2 are optimal.

Proof. The only condition in Theorem 6.2 not explicitly assumed here is (6.2.3). That $T \in J_{m,1}$ implies $(d_m - a)/d_1 < T$, so

$$\begin{aligned} d_m &< d_1 T + a = c_2 T - (c_2 - d_1) T + a \\ &= c_2 T - (d_2 - d_1 - a) T + a \\ &\leq c_2 T - (d_1 - a) T + a \\ &= c_2 T + a(1 - (d_1/a - 1) T) \\ &\leq c_2 T + a(1 - (1 - a/d_1) T) \end{aligned}$$

because $x + x^{-1} \geq 2$ for $x > 0$. Now $1 - a/d_1 = c_1/d_1 \geq 1/T^* > 1/T$, so $d_m < c_2 T$. $\square$

This corollary includes all games where S_2 is an arithmetic progression, with the first term no larger than the common difference (so that $d_2 \geq 2d_1$) and S_1 a translate of S_2 . Although $d_2 \geq 2d_1$ is not a necessary condition (for some exceptions see Corollary 6.12 and the paragraph preceding it; also Theorem 6.14) we cannot simply remove it. If $S_1 = \{10r - 5 : r \in \mathbb{N}\}$, $S_2 = \{10r + 4 : r \in \mathbb{N}\}$, and $T = 4$ we have $T^* = 2.8$ and $T \in J_{6,1} \setminus J_{7,1}$, so $m = 6$. Then $d_m = 64$ while $c_2 T = 60$, so (6.2.3) fails. The solution of this game is not like that described in Theorem 6.2, but rather, as we may verify, $V = (1 + \alpha - \alpha^2)/(1 + \alpha + \alpha^2)$, $P = p(\alpha^2, \alpha, 0, 0, 0, 1)$, and $Q = p(1, \alpha, 0, 0, 0, \alpha^2)$, where $\alpha = 2/(\nu + 1)$ and $p = (1 + \alpha + \alpha^2)^{-1}$. (This is a class 5A.2 game.)

Another collection of games satisfying the conditions of Theorem 6.2 is given in the next corollary.

COROLLARY 6.4. Let S_1 , S_2 , and ν be as in Theorem 6.2, and assume that T satisfies (6.2.1) and (6.2.2). If $d_2 < c_2 T$ and

$$(6.4.1) \quad d_1 < c_2 c_r / d_r \quad \text{for all } r \geq 3,$$

then the solution of the game is as described in Theorem 6.2.

Proof. That T is in $J_{m,1}$ implies that $c_m/d_1 < T$. We have assumed $d_2 < c_2 T$. If $m \geq 3$, then by (6.4.1), $d_m < c_2 c_m / d_1 < c_2 T$, so (6.2.3) is satisfied. $\square$

Example 6.5. Let $p > 0$, $S_1 = \{n^p : n \in \mathbb{N}\}$ and $S_2 = \{d_1, d_2, \dots\}$ be any sequence interlacing S_1 with $d_1 < (\frac{3}{2})^p$. Then $c_2 c_r / d_r > c_2 c_r / c_{r+1} = (2r/(r+1))^p \geq (\frac{3}{2})^p \geq d_1$, so Corollary 6.4 applies for all $T > \max \{d_1/c_1, c_2/d_1, d_2/c_2\}$ lying in any $J_{m,1}$.

Example 6.6. Let $S_1 = \{\log(n+a) : n \in \mathbb{N}\}$ where $a > 0$, and let S_2 be any sequence interlacing S_1 with $d_1 < c_2 c_3 / c_4$. Since $(\log(n+a))/(\log(n+a+1))$ increases with n , $d_1 < c_2 c_r / c_{r+1} < c_2 c_r / d_r$ for all $r \geq 3$, and Corollary 6.4 applies for all $T > \max \{d_1/c_1, c_2/d_1, d_2/c_2\}$ lying in any $J_{m,1}$. If $a = 1$, then $c_1 \doteq .6931$, $c_2 \doteq 1.0986$, and (6.4.1) holds when $d_1 < .9462$. If $a = .01$ then $c_1 \doteq .0100$, $c_2 \doteq .6981$, and (6.4.1) holds when $d_1 < .5539$.

Our next theorem shows that for many Silverman games, including all where $\{c_n\}$ has subexponential growth rate, class 3A is the predominant one in that there is a number $\bar{T}$ such that class 3A occurs for all $T > \bar{T}$ when $\nu \geq 1$.

THEOREM 6.7. Let S_1 , S_2 , and ν be as in Theorem 6.2, and assume that

$$(6.7.1) \quad c_{n+1}/c_n \leq T_* := \min \{d_1/c_1, c_2/d_1\} \quad \text{for all } n \geq N \text{ (for some } N \geq 2).$$

Then for all $T > \bar{T} = \max \{c_N/d_1, d_N/c_2, d_1/c_1\}$ the solution of the game is that given in Theorem 6.2.

Proof. Since $N \geq 2$, $\bar{T} \geq T^*$ and (6.2.1) holds. We have $T > c_N/d_1$, the left endpoint of $J_{N,1}$ so to establish (6.2.2) it suffices to show that $J_{m,1}$ and $J_{m+1,1}$ connect for $m \geq N$. They connect provided that $c_{m+1}/d_1 \leq d_m/c_1$, i.e., provided $c_{m+1}/d_m \leq d_1/c_1$. But $c_{m+1}/d_m < c_{m+1}/c_m \leq d_1/c_1$ by (6.7.1).

The m of (6.2.2) is clearly $\geq N$, so $d_m/c_m < c_{m+1}/c_m \leq c_2/d_1$ by (6.7.1). Thus, $d_m/c_2 < c_m/d_1 < T$; so (6.2.3) holds. $\square$

COROLLARY 6.8. *If $S_1 = \{f(n): n \in \mathbb{N}\}$, where f is a positive strictly increasing unbounded function on $[1, \infty)$ satisfying*

$$(6.8.1) \quad f'(x)/f(x-1) \rightarrow 0 \quad \text{as } x \rightarrow \infty, \quad \text{and } S_2 \text{ interlaces } S_1,$$

then the Silverman game on $S_1 \times S_2$ is in Class 3A for all sufficiently large T and $\nu \geq 1$.

Note that condition (6.8.1) will be satisfied by all rational functions.

Proof. By the mean value theorem there is a number x_n in $(n, n+1)$ such that $f(n+1)/f(n) = 1 + f'(x_n)/f(n) < 1 + f'(x_n)/f(x_n-1) \rightarrow 1$ as $n \rightarrow \infty$ by (6.8.1). Since $T_* > 1$, (6.7.1) follows. $\square$

We now turn to class 3A for $\nu < 1$. To reduce the number of levels of subscripts we use function notation for the elements of S_1 and S_2 .

THEOREM 6.9 (Class 3A, $\nu < 1$). *Let S_1 and S_2 be as in (6.0.1), $0 < \nu < 1$, and $T > T^*$. Let $\theta = \cos^{-1}(1/(\nu+1))$ and t be the integer such that $t < \pi/\theta \leq t+1$. Suppose there are integers $1 = u_1 < u_2 < \cdots < u_t$ such that*

$$(6.9.1) \quad \max_{1 \leq i \leq t-1} \frac{c(u_{i+1})}{d(u_i)} < T \leq \min_{1 \leq i \leq t-1} \frac{d(u_{i+1})}{c(u_i)}.$$

Let $v_1 = 1$ and for $1 \leq k \leq t-1$ let v_{k+1} be the integer such that

$$(6.9.2) \quad d(v_{k+1}) < c(v_{k+1})T \leq d(v_{k+1}+1).$$

Assume that

$$(6.9.3) \quad d(v_{k-1}) \leq c(v_{k+1})/T \quad \text{for } k = 2, 3, \dots, t.$$

Let $\alpha = 2/(\nu+1)$ and define $x_1, x_2, \dots, x_t$ by

$$(6.9.4) \quad \begin{cases} x_2 = \alpha x_1, \\ x_{k+1} = \alpha x_k - x_{k-1} & \text{for } 2 \leq k \leq t-1, \\ \sum_{k=1}^t x_k = 1. \end{cases}$$

Then $x_k \geq 0$ for $k = 1, \dots, t$; $V = \nu - (\nu+1)x_1$, and the strategies P and Q given by $p_{u_{t+1-i}} = x_i = q_{v_i}$, $i = 1, \dots, t$, are optimal.

Remarks. (i) Condition (6.9.1) is equivalent to $T \in \bigcap_{i=1}^{t-1} J_{u_{i+1}, u_i}$.

(ii) The condition defining t is equivalent to $\sec \pi/(t+1) - 1 \leq \nu < \sec \pi/t - 1$. For $\nu \geq 1$ we would have $\theta \geq \pi/3$ and $t = 2$. Condition (6.9.1) then reduces to $T \in J_{m,1}$ for some m and we would have Theorem 6.2.

(iii) From (6.9.4) it is clear that x_1 , and hence V , is a piecewise rational function of ν , each piece corresponding to an interval $\sec \pi/(t+1) - 1 \leq \nu < \sec \pi/t - 1$ on which t is constant. Here V is the bouncing ball function of [3]. As shown in that paper $V \geq 0$, and $V = 0$ if and only if $\nu = \sec \pi/t - 1$ for some integer $t \geq 3$.

Proof of Theorem 6.9. We first show that $E(P, d(n)) \geq \nu - (\nu+1)x_1$ for all n .

(i) Let $1 \leq n \leq u_2 - 1$. Since $T > c(u_2)/d(1)$ by (6.9.1), we have $c(u_2) < d(1)T \leq d(n)T$. Also, $d(n) \leq d(u_2 - 1) < c(u_2)$, so $d(n) < c(u_2) \leq d(n)T$, and thus p_{u_2} is in the third sum in (6.1.1). Then p_{u_1} is in one of the three sums, so $E(P, d(n)) \geq$

$(\nu - 1)p_{u_1} + (\nu + 1)p_{u_2} - \nu = (\nu - 1)x_t + (\nu + 1)x_{t-1} - \nu = V_1$, say. We shall show later that $V_1 = \nu - (\nu + 1)x_1$.

(ii) Let $u_k \leq n \leq u_{k+1} - 1$ ($k = 2, \dots, t-1$). Since by (6.9.1), $T \leq d(u_k)/c(u_{k-1}) \leq d(n)/c(u_{k-1})$, we have for $1 \leq i \leq k-1$ that $c(u_i) \leq c(u_{k-1}) \leq d(n)/T$ and therefore that p_{u_i} is in the first sum in (6.1.1). Also, $d(n) \leq d(u_{k+1} - 1) < c(u_{k+1}) < d(u_k)T$ by (6.9.1), and $d(u_k)T \leq d(n)T$, so $p_{u_{k+1}}$ is in the third sum. Thus

$$\begin{aligned} E(P, d(n)) &\geq 2\nu \sum_{i=1}^{k-1} p_{u_i} + (\nu - 1)p_{u_k} + (\nu + 1)p_{u_{k+1}} - \nu \\ &= 2\nu \sum_{i=t-k+2}^t x_i + (\nu - 1)x_{t-k+1} + (\nu + 1)x_{t-k} - \nu = V_k, \end{aligned}$$

say. Later we show that each $V_k = \nu - (\nu + 1)x_1$.

(iii) Let $n \geq u_t$. By (6.9.1) we have $c(u_{t-1}) \leq d(u_t)/T \leq d(n)/T$, so $p_{u_{t-1}}$ is in the first sum in (6.1.1). Then

$$\begin{aligned} E(P, d(n)) &\geq 2\nu \sum_{i=1}^{t-1} p_{u_i} + (\nu - 1)p_{u_t} - \nu \\ &= 2\nu \sum_{i=2}^t x_i + (\nu - 1)x_1 - \nu \\ &= 2\nu(1 - x_1) + (\nu - 1)x_1 - \nu \\ &= \nu - (\nu + 1)x_1 = V_t, \end{aligned}$$

say.

Now $V_t - V_{t-1} = (\nu + 1)x_2 - 2x_1 = (\nu + 1)(x_2 - \alpha x_1) = 0$, and for $k = 2, \dots, t-1$,

$$\begin{aligned} V_k - V_{k-1} &= (\nu + 1)x_{t-k+2} - 2x_{t-k+1} + (\nu + 1)x_{t-k} \\ &= (\nu + 1)(x_{t-k+2} - \alpha x_{t-k+1} + x_{t-k}) = 0. \end{aligned}$$

Thus $V_k = V_t$ for $k = 1, 2, \dots, t-1$, and we have established that $E(P, d(n)) \geq \nu - (\nu + 1)x_1$ for all n .

We complete the proof by showing that $E(c(n), Q) \leq \nu - (\nu + 1)x_1$ for all n .

(i) Let $n = 1$. Since $T > T^* \geq d(1)/c(1)$, we have $c(1) < d(1) < c(1)T$, and q_1 is in the third sum in (6.1.2). Consequently, $E(c(1), Q) \leq -(\nu + 1)q_1 + \nu = \nu - (\nu + 1)x_1 = W_1$, say.

(ii) Let $2 \leq n \leq v_2$. We have $c(n) \leq c(v_2) < d(v_2) < c(2)T$ by definition of v_2 , and $c(2)T \leq c(n)T$ so q_{v_2} is in the third sum in (6.1.2). It follows that $E(c(n), Q) \leq -(\nu - 1)q_1 - (\nu + 1)q_{v_2} + \nu = -(\nu - 1)x_1 - (\nu + 1)x_2 + \nu = W_2$, say.

(iii) For fixed k in the range $3 \leq k \leq t$, let $v_{k-1} + 1 \leq n \leq v_k$. From (6.9.3) we have that $d(v_{k-2}) \leq c(v_{k-1} + 1)/T \leq c(n)/T$, so that $q_{v_{k-2}}$ is in the first sum in (6.1.2). Since $c(n) \leq c(v_k) < d(v_k) < c(v_{k-1} + 1)T$ by (6.9.2), and $c(v_{k-1} + 1)T \leq c(n)T$, q_{v_k} is in the third sum in (6.1.2). Thus

$$\begin{aligned} E(c(n), Q) &\leq -2\nu \sum_{i=1}^{k-2} q_{v_i} - (\nu - 1)q_{v_{k-1}} - (\nu + 1)q_{v_k} + \nu \\ &= -2\nu \sum_{i=1}^{k-2} x_i - (\nu - 1)x_{k-1} - (\nu + 1)x_k + \nu = W_k, \end{aligned}$$

say.

(iv) Let $n \geq v_t + 1$. From (6.9.3) we have $d(v_{t-1}) \leq c(v_t + 1)/T \leq c(n)/T$, so $q_{v_{t-1}}$ is in the first sum in (6.1.2). It follows that $E(c(n), Q) \leq -2\nu \sum_{i=1}^{t-1} q_{v_i} - (\nu - 1)q_{v_t} + \nu = -2\nu \sum_{i=1}^{t-1} x_i - (\nu - 1)x_t + \nu = W_{t+1}$, say.

Now $W_2 - W_1 = 2x_1 - (\nu + 1)x_2 = (\nu + 1)(\alpha x_1 - x_2) = 0$. For $k = 3, \dots, t$,

$$\begin{aligned} W_k - W_{k-1} &= -2\nu \sum_{i=1}^{k-2} x_i - (\nu - 1)x_{k-1} - (\nu + 1)x_k + 2\nu \sum_{i=1}^{k-3} x_i \\ &\quad + (\nu - 1)x_{k-2} + (\nu + 1)x_{k-1} \\ &= -2\nu x_{k-2} + (\nu - 1)x_{k-2} + 2x_{k-1} - (\nu + 1)x_k \\ &= -(\nu + 1)x_{k-2} + 2x_{k-1} - (\nu + 1)x_k \\ &= -(\nu + 1)(x_{k-2} - \alpha x_{k-1} + x_k) = 0. \end{aligned}$$

($\sum_{i=1}^0 x_i$ is understood to be zero.) Finally,

$$\begin{aligned} W_{t+1} - W_t &= -2\nu \sum_{i=1}^{t-1} x_i - (\nu - 1)x_t + 2\nu \sum_{i=1}^{t-2} x_i + (\nu - 1)x_{t-1} + (\nu + 1)x_t \\ &= -2\nu x_{t-1} + (\nu - 1)x_{t-1} + 2x_t \\ &= -(\nu + 1)x_{t-1} + 2x_t \\ &= (\nu + 1)(\alpha x_t - x_{t-1}). \end{aligned}$$

Let $a_k = \sin k\theta$. Then as shown in the proof of Theorem 7 in [3] $x_k = a_k / \sum_{i=1}^t a_i$ and $(\alpha x_t - x_{t-1}) \sum_{i=1}^t a_i = \alpha a_t - a_{t-1} = 2 \cos \theta \sin t\theta - \sin(t-1)\theta = \sin(t+1)\theta \leq 0$ because $t\theta < \pi \leq (t+1)\theta$. Thus $W_{t+1} \leq W_t = W_1$ and we have $E(c(n), Q) \leq W_1 = \nu - (\nu + 1)x_1$ for all n . From the above expression for x_k we see also that each $x_k \geq 0$. $\square$

It is not always immediately evident whether integers u_i satisfying (6.9.1) exist. For a given S_1 and S_2 we shall denote by $\mathcal{T}_{3A}$ the set of all T for which the game is in class 3A (notation is similar for the other classes). Given $\nu < 1$, let $\tilde{\mathcal{T}}_{3A}$ be the subset of (T^*, ∞) such that for each T in $\tilde{\mathcal{T}}_{3A}$, $u_1, \dots, u_t$ exist satisfying (6.9.1). The subset of $\tilde{\mathcal{T}}_{3A}$ where (6.9.3) holds is contained in $\mathcal{T}_{3A}$. In attempting to apply Theorem 6.9 the questions are: (1) What part of (T^*, ∞) is $\tilde{\mathcal{T}}_{3A}$? (2) Given that T is in $\tilde{\mathcal{T}}_{3A}$, is (6.9.3) satisfied? With regard to the first, we could perhaps develop an algorithm to generate $\tilde{\mathcal{T}}_{3A}$. We can construct $\cup_m J_{m,1}$, $\cup_m J_{m,2}$, etc., and presumably from these construct $\tilde{\mathcal{T}}_{3A}$. In § 10 we shall determine $\mathcal{T}_{3A}$, as well as the other $\mathcal{T}_i$ corresponding to games in class i , in some examples.

In the case of odd versus even integers, $\mathcal{T}_{3A}$ was seen to be the entire interval (T^*, ∞) [3]. This is frequently the case when S_1 and S_2 are arithmetic sequences, especially if the first term of S_2 is no larger than the common difference (or when S_1 and S_2 are powers of such sequences), as we now show. Let $S_1 = \{c, c + a, c + 2a, \dots\}$ and $S_2 = \{d, d + a, d + 2a, \dots\}$, where $0 < c < d < c + a$. In view of Theorem 2.1 we may replace S_1 and S_2 by the equivalent sets $S_1/a = \{\gamma, \gamma + 1, \gamma + 2, \dots\}$ and $S_2/a = \{\delta, \delta + 1, \delta + 2, \dots\}$, where $\gamma = c/a$ and $\delta = d/a$. First consider condition (6.9.3). The v_k which occur are defined by (6.9.2), from which we have $\delta + v_{k+1} - 1 < (\gamma + v_k)T \leq \delta + v_{k+1}$ for $k \geq 1$ and $v_1 = 1$. Thus

$$(6.9.5) \quad \gamma + v_{k-1} \leq (\delta + v_k)/T = ((\gamma + v_k) + (\delta - \gamma))/T,$$

and for (6.9.3) we need that $\delta + v_{k-1} - 1 \leq (\gamma + v_k)/T$. But

$$\delta + v_{k-1} - 1 = (\gamma + v_{k-1}) - (1 - \delta + \gamma) \leq (\gamma + v_k)/T + (\delta - \gamma)/T - (1 - \delta + \gamma)$$

by (6.9.5). Thus (6.9.3) holds if $(\delta - \gamma)/T \leq (1 - \delta + \gamma)$, and consequently holds for all $T > T^*$ whenever $(\delta - \gamma)/(1 - \delta + \gamma) \leq T^* = \max \{ \delta/\gamma, (\gamma + 1)/\delta \}$.

As it happens, $(\delta - \gamma)/(1 - \delta + \gamma) \leq (\gamma + 1)/\delta$ implies $(\delta - \gamma)/(1 - \delta + \gamma) \leq \delta/\gamma$. To see this, clear of fractions and regroup terms to obtain the equivalent implication $\delta(\delta + 1) \leq (\gamma + 1)^2 \Rightarrow \delta(\delta - 1) \leq \gamma^2$. Here we consider three cases. If $0 < \delta \leq 1$, then $\delta(\delta - 1) \leq 0 \leq \gamma^2$. If $1 < \delta \leq \gamma + .5$, then $\delta(\delta - 1) \leq (\gamma + .5)(\gamma - .5) < \gamma^2$. If $\delta > \gamma + .5$, then $\delta(\delta - 1) = \delta(\delta + 1) - 2\delta < (\gamma + 1)^2 - 2(\gamma + .5) = \gamma^2$. Finally, in this connection, we observe that $\delta \leq \gamma + .5 \Rightarrow \delta(\delta + 1) < (\gamma + 1)^2$ also. We summarize these results in the next theorem.

THEOREM 6.10. *Let $S_1 = \{\gamma, \gamma + 1, \gamma + 2, \dots\}$ and $S_2 = \{\delta, \delta + 1, \delta + 2, \dots\}$, where $0 < \gamma < \delta < \gamma + 1$. Then*

$$\begin{aligned} \delta \leq \gamma + .5 &\Rightarrow \delta(\delta + 1) < (\gamma + 1)^2 \Leftrightarrow (\delta - \gamma)/(1 - \delta + \gamma) < (\gamma + 1)/\delta \\ &\Rightarrow \delta(\delta - 1) < \gamma^2 \Leftrightarrow (\delta - \gamma)/(1 - \delta + \gamma) < \delta/\gamma \\ &\Rightarrow (\delta - \gamma)/(1 - \delta + \gamma) < T^* \Rightarrow (6.9.3) \end{aligned}$$

holds for all $T > T^*$. $\square$

Note that (6.9.3) holds for all $T > T^*$ when $\delta \leq 1$ (i.e., the first term of S_2 does not exceed the common difference), because then $\delta(\delta - 1) < \gamma^2$. The region where $\delta(\delta - 1) < \gamma^2$ is that beneath the upper curve inside the diagonal band in Fig. 2.

We now give sufficient conditions on S_1 and S_1 for the existence of integers u_i satisfying (6.9.1).

THEOREM 6.11. *Let S_1 and S_2 be as in Theorem 6.10, and $T > T^*$. Let integers u_k be defined recursively by*

$$\begin{aligned} (6.11.1) \quad u_1 &= 1, \quad \text{for } k \geq 1, \\ (\delta + u_k - 1)T &\leq \gamma + u_{k+1} < (\delta + u_k - 1)T + 1. \end{aligned}$$

If $(1 - \delta + \gamma)/(\delta - \gamma) \leq T$, then (6.9.1) holds. Furthermore,

$$\begin{aligned} (6.11.2) \quad \delta \geq \gamma + .5 &\Rightarrow \delta^2 > \gamma(\gamma + 1) \Leftrightarrow \\ (1 - \delta + \gamma)/(\delta - \gamma) &< (\gamma + 1)/\delta \Leftrightarrow (1 - \delta + \gamma)/(\delta - \gamma) < \delta/\gamma. \end{aligned}$$

Thus $\delta^2 \geq \gamma(\gamma + 1) \Leftrightarrow (1 - \delta + \gamma)/(\delta - \gamma) \leq T^* \Rightarrow (6.9.1)$.

Proof. From (6.11.1) we have $(\gamma + u_{k+1} - 1)/(\delta + u_k - 1) < T$ for each k , so to establish (6.9.1) it remains only to show that $T \leq (\delta + u_{k+1} - 1)/(\gamma + u_k - 1)$ for each k . From the other part of (6.11.1) we have $(\delta + u_k - 1)T - 1 + (\delta - \gamma) \leq \delta + u_{k+1} - 1$. Divide by T to obtain $(\gamma + u_k - 1) + (\delta - \gamma) - (1 - \delta + \gamma)/T \leq (\delta + u_{k+1} - 1)/T$. By our assumption, $(\delta - \gamma) - (1 - \delta + \gamma)/T \geq 0$, so

$$(\gamma + u_k - 1) \leq (\delta + u_{k+1} - 1)/T$$

as desired.

As for (6.11.2), the first implication is obvious, and the two equivalences are immediate upon clearing of fractions. The final statement follows from the fact that $T > T^* = \max \{ \delta/\gamma, (\gamma + 1)/\delta \}$. $\square$

The region where $\delta^2 > \gamma(\gamma + 1)$ is that which lies above the lower curve in Fig. 2. In the central region of Fig. 2 both $\delta^2 > \gamma(\gamma + 1)$ and $\delta(\delta - 1) < \gamma^2$ hold, and consequently both (6.9.1) and (6.9.3) hold for all $T > T^*$; i.e., $\mathcal{T}_{3A} = (T^*, \infty)$.

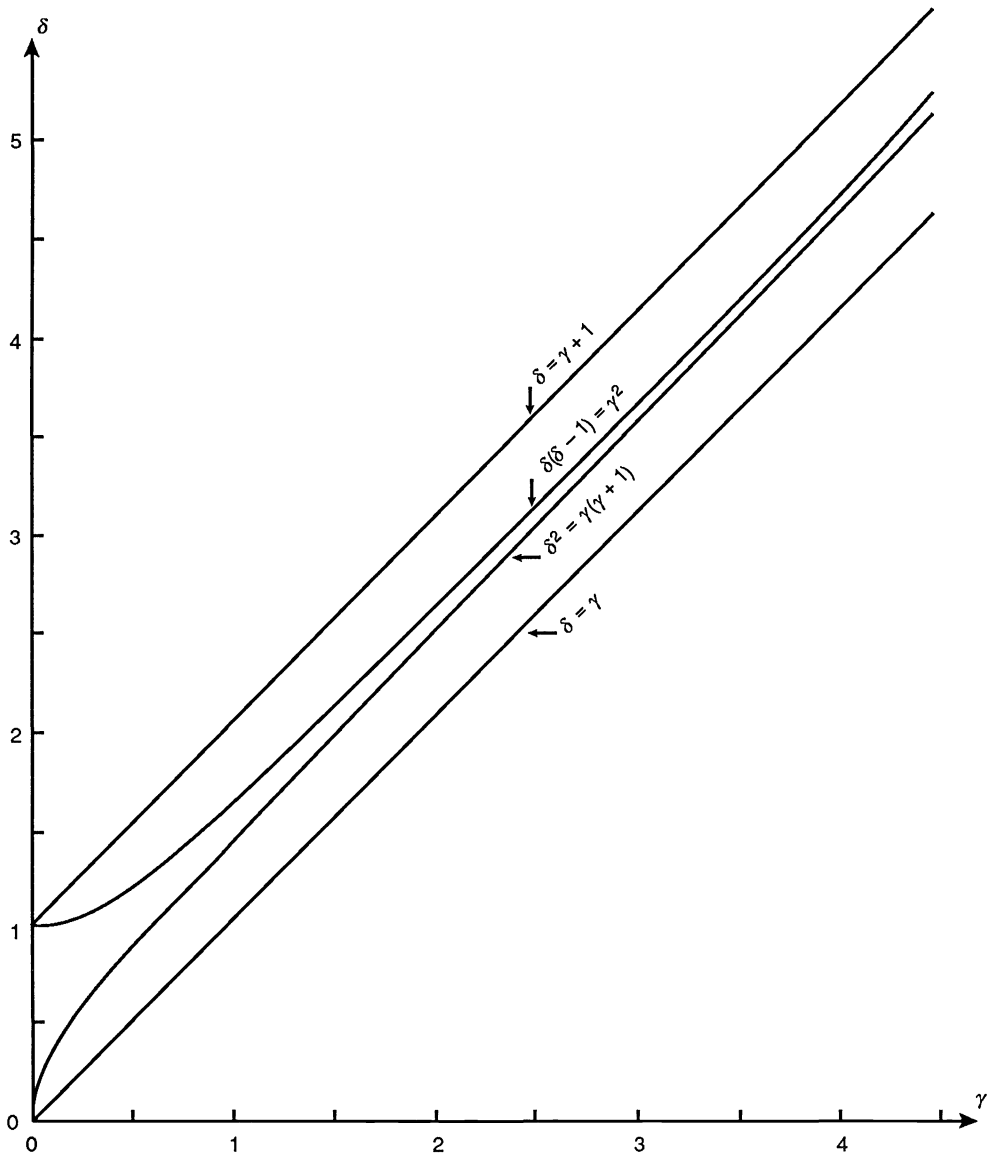


FIG. 2

COROLLARY 6.12. If $S_1 = \{c, c + a, c + 2a, \dots\}$ and $S_2 = \{d, d + a, d + 2a, \dots\}$ with $d = c + a/2$, then $\mathcal{T}_{3A} = (T^*, \infty)$.

Proof. Since $\delta = \gamma + .5$, both (6.9.1) and (6.9.3) hold for all $T > T^*$, by Theorems 6.10 and 6.11. $\square$

Remark. We have $0 < \delta - \gamma < 1$. If $\delta - \gamma$ is too near zero, the condition $(1 - \delta + \gamma)/(\delta - \gamma) \leq T$ fails and (6.9.1) is at risk. When $\delta - \gamma$ is too near one the condition $(\delta - \gamma)/(1 - \delta + \gamma) \leq T$ fails and (6.9.3) is at risk if $\delta > 1$.

COROLLARY 6.13. Let S_1 and S_2 be as in Theorem 6.10. Let

$$\bar{T} = \max \{ (\delta - \gamma)/(1 - \delta + \gamma), (1 - \delta + \gamma)/(\delta - \gamma) \}.$$

Then $(\bar{T}, \infty) \subset \mathcal{T}_{3A}$ for all $\nu > 0$.

Proof. Theorems 6.10 and 6.11 show that both (6.9.1) and (6.9.3) hold when $T > \bar{T}$. $\square$

In Corollary 6.12 each d_n is the arithmetic mean of c_n and c_{n+1} , and S_1 is in arithmetic sequence. A much larger class of games in which $\mathcal{T}_{1A} = (1, d_1/c_1]$ and $\mathcal{T}_{3A} = (d_1/c_1, \infty)$ for all $\nu > 0$ is obtained by taking d_n to be the geometric mean of c_n and c_{n+1} .

THEOREM 6.14. *Let S_1 and S_2 be as in (6.0.1), with $d_n = (c_n c_{n+1})^{1/2}$ and assume that*

$$(6.14.1) \quad c_{n+2}/c_{n+1} \leq c_{n+1}/c_n \quad \text{for all } n \geq 1.$$

Then for all $\nu > 0$ we have $\mathcal{T}_{1A} = (1, d_1/c_1]$ and $\mathcal{T}_{3A} = (d_1/c_1, \infty)$.

Proof. $T^* = d_1/c_1 = c_2/d_1$, and for $1 < T \leq T^*$, Theorem 3.1 or Theorem 3.2 applies.

For $T > T^*$ and $\nu \geq 1$ we show that the conditions of Theorem 6.2 are satisfied. Since $c_{m+1}/d_m = (c_{m+1}/c_m)^{1/2} \leq (c_2/c_1)^{1/2} = d_1/c_1$, we have $c_{m+1}/d_1 \leq d_m/c_1$. This means that the intervals $J_{m,1}$ and $J_{m+1,1}$ connect, so (6.2.2) is satisfied for all $T > T^*$. Also, from $c_{m+1}/c_m \leq c_2/c_1$ we have $d_m/c_2 = (c_m c_{m+1})^{1/2}/c_2 \leq c_m/(c_1 c_2)^{1/2} = c_m/d_1 < T$, so (6.2.3) holds.

For $T > T^*$ and $\nu < 1$ we show that the conditions of Theorem 6.9 are satisfied. Let $u_1 = 1$ and for $i \geq 1$, when u_i has been determined let u_{i+1} be the largest integer such that $c(u_{i+1})/d(u_i) < T$. (Recall that by (6.0.1), $c_n \rightarrow \infty$.) Then the first inequality in (6.9.1) holds, and the second, that $T \leq d(u_{i+1})/c(u_i)$, follows from (6.14.1) and the fact that $T \leq c(u_{i+1} + 1)/d(u_i)$, using that $d_n = (c_n c_{n+1})^{1/2}$. With the v_k as defined by (6.9.2) we have for each k that $T \leq d(v_{k+1} + 1)/c(v_k + 1) \leq c(v_{k+1} + 1)/d(v_k)$, as is readily shown again with the use of (6.14.1), so (6.9.3) holds. $\square$

Finally we remark that Theorems 6.2 and 6.9 give sufficient conditions for the game to be in class 3A. The remaining cases that fall into class 3A for $\nu \geq 1$ are given in Theorems 8.2, 8.3, 8.7, and 8.8.

7. Class 3B. Recall that when $T \leq T^*$ the game is in class 1A or 2B. For $T > T^*$, when $T \in J_{m,1}$ for some m and certain other conditions are obtained, the game is in class 3A. If T is in no $J_{m,1}$, let m_1 denote the integer such that T falls between $J_{m_1,1}$ and $J_{m_1+1,1}$:

$$(7.0.1) \quad d_{m_1}/c_1 < T \leq c_{m_1+1}/d_1.$$

Class 3B games occur only when $T > T^*$, T is in no $J_{m,1}$ but T is in $J_{m,n}$ for some n with $2 \leq n \leq m_1$. The game value is $V = -H_3(\nu)$, the negative of that in class 3A, and optimal strategies of the two players are essentially interchanged from those in 3A. As in the preceding section we shall formulate these results in separate theorems for $\nu \geq 1$ and $\nu < 1$.

THEOREM 7.1 (Class 3B, $\nu \geq 1$). *Let S_1 and S_2 be as in (6.0.1) and $\nu \geq 1$. Assume that $T > T^*$, that T is in none of the intervals $J_{m,1}$ but that*

$$(7.1.1) \quad T \in J_{m,n} \text{ for some } m, n, \text{ with } 2 \leq n \leq m_1 < m \text{ (} m_1 \text{ defined by (7.0.1))}.$$

Then $V = (-\nu + 1)/(\nu + 3)$, and the strategies P and Q with $p_m = q_1 = 2/(\nu + 3)$, $p_n = q_{m_1} = (\nu + 1)/(\nu + 3)$ are optimal for Players I and II, respectively.

Remarks. (i) $n \leq m_1$ implies $c_n \leq c_{m_1} < d_{m_1} < c_1 T$ by (7.0.1), and thus

$$(7.1.2) \quad c_n < d_1 T.$$

(ii) $1 < T \in J_{m,n} = (c_m/d_n, d_m/c_n]$ implies that $m \geq n$. If $m = n$ we would have a class 2A game: (7.1.2) implies that $A(c_n, d_i) = 1$ for $1 \leq i \leq n - 1$, and $T \leq d_n/c_n$ implies that $A(c_n, d_i) = \nu$ for $i \geq n$.

Proof of Theorem 7.1. We first show that $E(P, d_s) \geq (-\nu + 1)/(\nu + 3) = -H_3(\nu)$ for all s .

(i) Let $1 \leq s \leq n - 1$. From (7.1.2) we see that $d_s \leq d_{n-1} < c_n < d_1 T \leq d_s T$, so p_n is in the third sum in (6.1.1). It follows that $E(P, d_s) \geq (\nu + 1)^2/(\nu + 3) - \nu = -H_3(\nu)$.

(ii) Let $n \leq s \leq m - 1$. Then $d_s < c_m$, and because T is in $J_{m,n}$, $c_m < d_n T \leq d_s T$, so p_m is in the third sum in (6.1.1). Since $n < m$, p_n is in one of the three sums, and thus $E(P, d_s) \geq ((\nu - 1)/(\nu + 1) + 2(\nu + 1))/(\nu + 3) - \nu = -H_3(\nu)$.

(iii) Let $s \geq m$. $T \in J_{m,n}$ implies that $d_s \geq d_m \geq c_n T$, so p_n is in the first sum in (6.1.1). Then $E(P, d_s) \geq 2\nu(\nu + 1)/(\nu + 3) - \nu \geq 0 \geq -H_3(\nu)$.

We complete the proof by showing that $E(c_r, Q) \leq -H_3(\nu)$ for all r .

(i) Let $1 \leq r \leq m_1$. Then in view of (7.0.1) we have $c_r < d_{m_1} < c_1 T \leq c_r T$, so q_{m_1} is in the third sum in (6.1.2). Also, q_1 is in one of the three sums, so $E(c_r, Q) \leq (-2(\nu - 1) - (\nu + 1)^2)/(\nu + 3) + \nu = -H_3(\nu)$.

(ii) Let $r > m_1$. From (7.0.1) we have $d_1 T \leq c_{m_1+1} \leq c_r$, so q_1 is in the first sum. Also, $d_{m_1} \leq d_{r-1} < c_r$, so q_{m_1} is in one of the first two sums. Thus $E(c_r, Q) \leq (-2(2\nu) - (\nu - 1)(\nu + 1))/(\nu + 3) + \nu = -H_3(\nu)$. $\square$

THEOREM 7.2 (Class 3B, $\nu < 1$). Let $S_1, S_2, \nu, \theta, \alpha, t$, and $x_1, \dots, x_t$ be as in Theorem 6.9. Assume that $T > T^*$, that T is in none of the intervals $J_{m,1}$, but that there are integers $1 < u_1 < u_2 < \dots < u_t$ such that

$$(7.2.1) \quad c(u_1) < d(1)T,$$

and (6.9.1) holds. Let $v_1 = 1$ and v_2 be the integer such that T falls between $J_{v_2,1}$ and $J_{v_2+1,1}$:

$$(7.2.2) \quad d(v_2)/c(1) < T \leq c(v_2 + 1)/d(1).$$

Then let $v_3, \dots, v_t$ be determined as in Theorem 6.9:

$$(7.2.3) \quad d(v_{k+1}) < c(v_k + 1)T \leq d(v_{k+1} + 1), \quad k = 2, \dots, t-1,$$

and assume that (6.9.3) holds.

Then $V = (\nu + 1)x_1 - \nu = -H_3(\nu)$, and the strategies P and Q given by $p_{u_i} = x_i = q_{v_{i+1}-1}$, $1 \leq i \leq t$, are optimal for Players I and II, respectively.

Remark. With $\nu \geq 1$, the t in this theorem would be two and the theorem would reduce to Theorem 7.1.

Proof of Theorem 7.2. We shall use $V_1, \dots, V_t$ and $W_1, \dots, W_t$ with the same meanings as in the proof of Theorem 6.9. We first show that $E(P, d_s) \geq -H_3(\nu)$ for all s .

(i) Let $1 \leq s \leq u_1 - 1$. With the help of (7.2.1) we see that $d(s) < c(s + 1) \leq c(u_1) < d(1)T \leq d(s)T$, so p_{u_1} is in the third sum in (6.1.1). Thus $E(P, d_s) \geq -H_3(\nu)$.

(ii) Let $u_1 \leq s < u_2$. Then $d(s) < c(u_2)$, and from (6.9.1) we have $c(u_2) < d(u_1)T \leq d(s)T$, so p_{u_2} is in the third sum in (6.1.1). Consequently $E(P, d_s) \geq (\nu - 1)p_{u_1} + (\nu + 1)p_{u_2} - \nu = (\nu - 1)x_1 + (\nu + 1)x_2 - \nu = -W_2$.

(iii) For fixed k , $3 \leq k \leq t$, let $u_{k-1} \leq s < u_k$. From (6.9.1) we have $c(u_{k-2}) \leq d(u_{k-1})/T \leq d(s)/T$, so $p_{u_{k-2}}$ is in the first sum in (6.1.1). Also, $d(s) < c(u_k) < d(u_{k-1})T$ by (6.9.1), and $d(u_{k-1})T \leq d(s)T$, so p_{u_k} is in the third sum. Thus $E(P, d_s) \geq 2\nu \sum_{i=1}^{k-2} x_i + (\nu - 1)x_{k-1} + (\nu + 1)x_k - \nu = -W_k$.

(iv) Let $s \geq u_t$. In view of (6.9.1), $c(u_{t-1}) \leq d(u_t)/T \leq d(s)/T$, so $p_{u_{t-1}}$ is in the first sum in (6.1.1). Then $E(P, d_s) \geq 2\nu \sum_{i=1}^{t-1} x_i + (\nu - 1)x_t - \nu = -W_{t+1}$. Thus we have $E(P, d_s) \geq -H_3(\nu)$ for all s .

We now show that $E(c(r), Q) \leq -H_3(\nu)$ for all r .

(i) Let $1 \leq r \leq v_2$. Then $c(r) < d(v_2)$, and by (7.2.2) we see that q_{v_2} is in the third sum in (6.1.2). Hence $E(c(r), Q) \leq -(\nu - 1)q_1 - (\nu + 1)q_{v_2} + \nu = -(\nu - 1)x_t - (\nu + 1)x_{t-1} + \nu = -V_1$.

(ii) Let $v_2 < r \leq v_3$. Formula (7.2.2) implies that q_1 is in the first sum in (6.1.2), and from (7.2.3) we see that $c(v_2 + 1) \leq c(r) < d(v_3) < c(v_2 + 1)T$. Thus q_{v_3} is in the third sum in (6.1.2), and $E(c(r), Q) \leq -2\nu q_1 - (\nu - 1)q_{v_2} - (\nu + 1)q_{v_3} + \nu = -V_2$.

(iii) Let $v_{k-1} < r \leq v_k$, for $4 \leq k \leq t$. From (6.9.3) we have

$$d(v_{k-2}) \leq c(v_{k-1} + 1)/T \leq c(r)/T,$$

so $q_{v_{k-2}}$ is in the first sum in (6.1.2). Moreover, $c(r) < d(v_k) < c(v_{k-1} + 1)T \leq c(r)T$ by (7.2.3), so q_{v_k} is in the third sum. Thus $E(c(r), Q) \leq -2\nu \sum_{i=1}^{k-2} q_{v_i} - (\nu - 1)q_{v_{k-1}} - (\nu + 1)q_{v_k} + \nu = -V_{k-1}$.

(iv) Let $r > v_t$. By (6.9.3), $d(v_{t-1}) \leq c(v_t + 1)/T \leq c(r)/T$, so $q_{v_{t-1}}$ is in the first sum in (6.1.2). It follows that $E(c(r), Q) \leq -2\nu \sum_{i=1}^{t-1} q_{v_i} - (\nu - 1)q_{v_t} + \nu = -2\nu \sum_{i=2}^t x_i - (\nu - 1)x_1 + \nu = -2\nu(1 - x_1) - (\nu - 1)x_1 + \nu = (\nu + 1)x_1 - \nu$. Thus we have $E(c(r), Q) \leq -H_3(\nu)$ for all r . $\square$

8. Other games with T in some $J_{m,1}$. For the remainder of the paper we shall assume $\nu \geq 1$, and in this section we deal with the exceptions to Theorem 6.2, namely those games where (6.2.1) and (6.2.2) hold with $m \geq 3$ but (6.2.3) fails. It is readily seen that we then have $d_{m-1} < c_m < d_1T < c_2T \leq d_m$ and $d_1T \leq c_{m+1}$ for some $m \geq 3$. The condition $T > T^*$ simplifies to $T > d_1/c_1$ since $T > c_2/d_1$ is implied by $c_m < d_1T$, and the condition $T \leq d_m/c_1$ from (6.2.2) is redundant in the presence of $c_2T \leq d_m$. If $c_mT \leq d_m$ then (since $c_m < d_1T$) we have $A(c_m, d_j) = 1$ for $1 \leq j < m$ and $A(c_m, d_j) = \nu$ for $j \geq m$. This is a class 2A game (Theorem 4.1). Thus for $\nu \geq 1$ and T in some $J_{m,1}$ the games which remain to be examined are those satisfying

$$(8.0.1) \quad d_1 < c_1T,$$

$$(8.0.2) \quad c_m < d_1T \leq c_{m+1},$$

$$(8.0.3) \quad c_2 < d_{m-1} < c_2T \leq d_m < c_mT.$$

We observe that in the notation of (6.0.3) and (6.0.4) we have $r_1 = m$ and $s_2 = m - 1$ (cf. remark (iii) after Theorem 6.2).

The conditions (8.0.1)–(8.0.3) hold if and only if the game matrix has the form shown in Fig. 3. Of course each row beginning at the diagonal element is nondecreasing

	d_1	d_2	d_3	$\cdots$	d_{m-1}	d_m	$\cdots$
c_1	-1	-1 or ν	-1 or ν	$\cdots$	-1 or ν	ν	$\cdots$
c_2	1	-1	-1	$\cdots$	-1	ν	$\cdots$
c_3	1	1	-1	$\cdots$	-1	-1 or ν	
$\vdots$							
c_m	1	1	-1	$\cdots$	1	-1	
c_{m+1}	$-\nu$	1 or $-\nu$	1 or $-\nu$	$\cdots$	1 or $-\nu$	1 or $-\nu$	
$\vdots$							

FIG. 3. Game matrix under conditions (8.0.1)–(8.0.3).

and each column below the diagonal element is nonincreasing. Note that $3 \leq m = r_1 \leq r_2 \leq \dots \leq r_{m-1}$. From (6.0.3) it follows that $d_{r_j-1} < c_{j+1}T$, so that

$$(8.0.4) \quad a_{j+1, r_j-1} = -1 \quad \text{if } j+1 \leq r_j-1.$$

Our theorems will separate these games into classes 3A, 4A.k, and 5A.k. The overall solution is most readily described by means of the flow chart in Fig. 4, and the argument in part is by induction on m . We show in Theorems 8.1–8.4 that the solution for $m = 3$ is as indicated in the chart; Theorems 8.5–8.9 give the solution for $m > 3$, using the

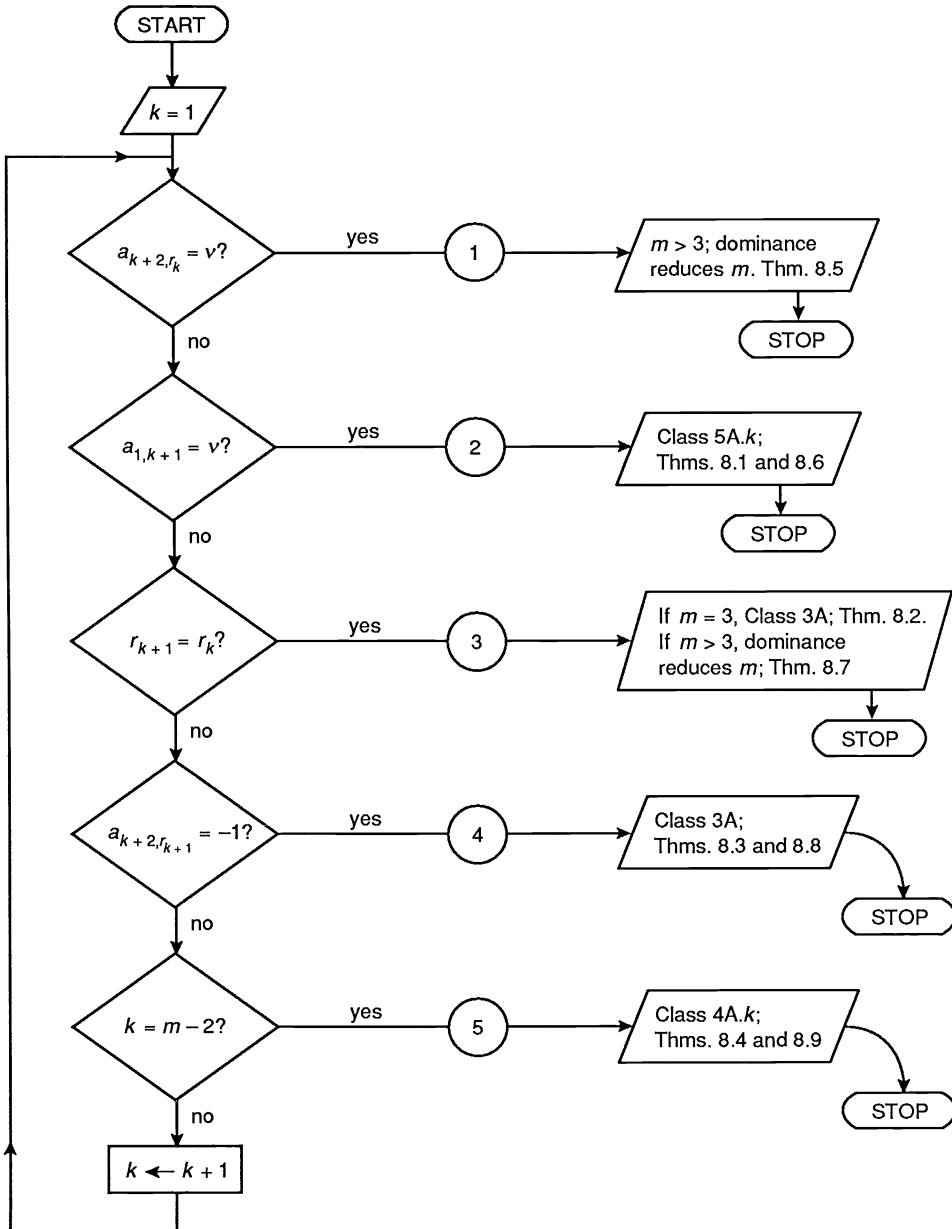


FIG. 4. Algorithm for solution to games satisfying (8.0.1)–(8.0.3).

induction hypothesis at points 1 and 3 of the chart where dominance allows us to reduce m by one.

As suggested earlier, the notation $\alpha = 2/(\nu + 1)$ will simplify expression of the solutions. If P or Q is written as a vector with finitely many components it is understood that all other components are zero. Observe that with $m = 3$ and $k = 1$ the answer at the first question box in the chart, whether $a_{3,3} = \nu$, is "no" because $d_3 < c_3 T$ by (8.0.3). Our first theorem deals with the case at point 2 on the chart, with $m = 3$.

THEOREM 8.1 (Class 5A.1, $\nu \geq 1$). *Let $m = 3$, $\nu \geq 1$ and assume (8.0.1)–(8.0.3). If $a_{1,2} = \nu$, then the game value is $(1 + \alpha - \alpha^2)p_3$, and optimal strategies are $P = p_3(\alpha^2, \alpha, 1)$ and $Q = p_3(1, \alpha, \alpha^2)$, where $p_3 = (1 + \alpha + \alpha^2)^{-1}$.*

Proof. Let $V = (1 + \alpha - \alpha^2)p_3$, the asserted game value, and note that $\nu\alpha = 2 - \alpha$. The game matrix is as follows. The probabilities (divided by p_3) for the asserted optimal strategies are written in the left and top margins:

		1	α	α^2	
		d_1	d_2	d_3	$d_4 \quad \dots$
α^2	c_1	-1	ν	ν	ν
α	c_2	1	-1	ν	ν
1	c_3	1	1	-1	-1 or ν
	c_4	$-\nu$	1 or $-\nu$	1 or $-\nu$	-1 or ν
	$\vdots$				

We have $E(P, d_1) = E(c_3, Q) = p_3(-\alpha^2 + \alpha + 1) = V$;

$$E(P, d_2) = E(c_2, Q) = p_3(\nu\alpha^2 - \alpha + 1) = p_3(2\alpha - \alpha^2 - \alpha + 1) = V;$$

and

$$E(P, d_3) = E(c_1, Q) = p_3(\nu\alpha^2 + \nu\alpha - 1) = p_3(2\alpha - \alpha^2 + 2 - \alpha - 1) = V.$$

For $j > 3$, $E(P, d_j) \geq E(P, d_3)$ and

$$\begin{aligned} E(c_j, Q) &\leq E(c_4, Q) \leq p_3(-\nu + \alpha + \alpha^2) = p_3(-2/\alpha + 1 + \alpha + \alpha^2) \\ &= p_3((1 + \alpha - \alpha^2) + 2(\alpha^3 - 1)/\alpha) \leq V \end{aligned}$$

since $\alpha \leq 1$. Thus $E(c_i, Q) \leq V \leq E(P, d_j)$ for all i, j . $\square$

Now suppose that we arrive at point 3 of the flow chart with $m = 3$. The negative answers to the first two questions mean that $a_{3,3} = a_{1,2} = -1$, and $r_2 = r_1 = 3$ implies that $a_{4,2} = -\nu$ (see (6.0.3)), so the game matrix is as follows:

	d_1	d_2	d_3	d_4	$\dots$
c_1	-1	-1	ν	ν	
c_2	1	-1	ν	ν	
c_3	1	1	-1	-1 or ν	
c_4	$-\nu$	$-\nu$	1 or $-\nu$	-1 or ν	
$\vdots$					

The second column dominates the first, and when the first is deleted, the first row dominates the second. Upon deleting row 2 we find that the game satisfies the hypotheses of Theorem 6.2 with $m = 2$. Thus we have the following theorem.

THEOREM 8.2. *If $m = 3$ and we arrive at point 3 of the flow chart of Fig. 4, then the game reduces by dominance to one satisfying the hypotheses of Theorem 6.2 with $m = 2$, so the game is in Class 3A.*

The next theorem deals with the situation at point 4 of the chart.

THEOREM 8.3 (Class 3A, $\nu \geq 1$). *Let $m = 3$, $\nu \geq 1$ and assume conditions (8.0.1)–(8.0.3). Assume further that $a_{3,3} = a_{1,2} = -1$, $r_2 > 3$ and that*

$$(8.3.1) \quad a_{3,r_2} = -1.$$

Then the game value is $(1 - \alpha)/(1 + \alpha)$ and optimal strategies P and Q are given by $p_2 = q_{r_2} = \alpha/(1 + \alpha)$, $p_3 = q_2 = 1/(1 + \alpha)$.

Remark. As we see from the game matrix below, the second row dominates the first. The game on $S'_1 \times S'_2$ where $S'_1 = S_2$ and $S'_2 = S_1 \setminus \{c_1\}$ may be seen to satisfy the conditions of Theorem 7.1, providing an alternate proof of the present theorem.

Proof of Theorem 8.3. The hypotheses are readily seen to imply that $r_2 \geq 4$ and the payoff matrix is as follows:

		1				α		
		d_1	d_2	d_3	$\cdots$	d_{r_2}	d_{r_2+1}	$\cdots$
α	c_1	-1	-1	ν	$\cdots$	ν	ν	
	c_2	1	-1	ν	$\cdots$	ν	ν	
1	c_3	1	1	-1	$\cdots$	-1	-1 or ν	
	c_4	$-\nu$	1	1	$\cdots$	-1	-1 or ν	
	$\vdots$							
	c_{r_2}	$-\nu$	1	1	$\cdots$	-1	-1 or ν	
	c_{r_2+1}	$-\nu$	$-\nu$	1 or $-\nu$	$\cdots$	1 or $-\nu$	-1 or ν	
	$\vdots$							

Let $V = (1 - \alpha)/(1 + \alpha)$. $E(P, d_1) = 1 > V$, $E(P, d_2) = V = E(c_i, Q)$ for $3 \leq i \leq r_2$, and $E(c_1, Q) = E(c_2, Q) = E(P, d_j) = (\nu\alpha - 1)/(1 + \alpha) = V$ for $3 \leq j \leq r_2$. (Recall that $\nu\alpha = 2 - \alpha$.) For $j > r_2$, $E(P, d_j) \geq E(P, d_{r_2})$ and $E(c_j, Q) \leq (-\nu + \alpha)/(1 + \alpha) = V + 2(\alpha^2 - 1)/\alpha(1 + \alpha) \leq V$. Thus $E(c_i, Q) \leq V \leq E(P, d_j)$ for all i, j . $\square$

If we arrive at question box 5 in the flow chart and $m = 3$, the answer is “yes” when $k = 1$ and we are at point 5. The next theorem deals with this case.

THEOREM 8.4 (Class 4A.1, $\nu \geq 1$). *Let $m = 3$, $\nu \geq 1$ and assume conditions (8.0.1)–(8.0.3). Assume further that $a_{3,3} = a_{1,2} = -1$, $r_2 > 3$ and*

$$(8.4.1) \quad a_{3,r_2} = \nu.$$

Then the game value is $(1 - \alpha + \alpha^2)/(1 + \alpha + \alpha^2)$ and optimal strategies are given by $p_3 = q_2 = (1 + \alpha + \alpha^2)^{-1}$, $p_2 = q_3 = \alpha p_3$, $p_{r_2} = q_1 = \alpha^2 p_3$.

Proof. Since by definition of r_2 (see 6.0.3)) $a_{r_2,2} = 1$, we have $a_{i,j} = 1$ whenever $j < i \leq r_2$. From (8.0.4) we have $a_{3,r_2-1} = -1$, and the game matrix is as follows:

		α^2		1	α			
		d_1	d_2	d_3	$\cdots$	d_{r_2-1}	d_{r_2}	$\cdots$
α	c_1	-1	-1	ν	$\cdots$	ν	ν	
	c_2	1	-1	ν	$\cdots$	ν	ν	
1	c_3	1	1	-1	$\cdots$	-1	ν	
	c_4	$-\nu$	1	1	$\cdots$	± 1	-1 or ν	
	$\vdots$							
α^2	c_{r_2}	$-\nu$	1	1	$\cdots$	1	-1 or ν	
	c_{r_2+1}	$-\nu$	$-\nu$	1 or $-\nu$				

Let $V = (1 - \alpha + \alpha^2)p_3$, the asserted game value. The nonzero components of the asserted optimal strategies P and Q , divided by p_3 , are again written in the left and top margins. We see from the matrix that

$$E(P, d_1) = (\alpha + 1 - \nu\alpha^2)p_3 = (\alpha + 1 - 2\alpha + \alpha^2)p_3 = V = E(P, d_2),$$

and for $3 \leq j < r_2$, $E(P, d_j) = (\nu\alpha - 1 + \alpha^2)p_3 = V$. For $j \geq r_2$,

$$\begin{aligned} E(P, d_j) &\geq (\nu\alpha + \nu - \alpha^2)p_3 = (2 - \alpha + 2/\alpha - 1 - \alpha^2)p_3 \\ &= ((1 - \alpha + \alpha^2) + (1 - \alpha^3)2/\alpha)p_3 \geq V, \end{aligned}$$

so $E(P, d_j) \geq V$ for all j . $E(c_1, Q) < E(c_2, Q) = (\alpha^2 - 1 + \nu\alpha)p_3 = V = E(c_3, Q)$. For $3 < i \leq r_2$, $E(c_i, Q) = (-\nu\alpha^2 + 1 + \alpha)p_3 = V$, and for $i > r_2$, $E(c_i, Q) < E(c_4, Q)$. Thus $E(c_i, Q) \leq V$ for all i . $\square$

Now we turn to the case $m > 3$, and our induction hypothesis is that the solutions indicated in Fig. 4 are correct for values of m smaller than the present one. We begin at point 1.

THEOREM 8.5. *Let $m > 3$, $1 \leq k \leq m - 2$, $\nu \geq 1$ and assume conditions (8.0.1)–(8.0.3). Assume further that*

$$(8.5.1) \quad a_{j+2, r_j} = -1 \text{ for } 1 \leq j < k, \text{ but } a_{k+2, r_k} = \nu,$$

$$(8.5.2) \quad a_{1k} = -1,$$

$$(8.5.3) \quad m = r_1 < r_2 < \cdots < r_k,$$

$$(8.5.4) \quad a_{j+1, r_j} = \nu \text{ for } 2 \leq j \leq k.$$

Then the game may be reduced by dominance to one in which (8.0.1)–(8.0.3) hold with $m - 1$ in place of m .

Proof. By (8.0.4) and our hypotheses, we see that the game matrix is as follows:

	d_1	d_2	$\cdots$	d_k	d_{k+1}	$\cdots$	d_{r_1-1}	d_{r_1}	$\cdots$	d_{r_k-1}	d_{r_k}	$\cdots$
c_1	-1	-1	$\cdots$	-1	-1 or ν	$\cdots$	-1 or ν	ν	$\cdots$	ν	ν	
c_2	1	-1	$\cdots$	-1	-1	$\cdots$	-1	ν	$\cdots$	ν	ν	
$\vdots$												
c_k	1	1	$\cdots$	-1	-1	$\cdots$	-1	-1 or ν	$\cdots$	-1 or ν	ν	
c_{k+1}	1	1	$\cdots$	1	-1	$\cdots$	-1	-1	$\cdots$	-1	ν	
c_{k+2}	1	1	$\cdots$	1	1	$\cdots$	-1	-1	$\cdots$	-1	ν	
$\vdots$												
c_{r_1}	1	1	$\cdots$	1	1	$\cdots$	1	-1	$\cdots$	-1	-1 or ν	
c_{r_1+1}	$-\nu$	1	$\cdots$	1	1	$\cdots$	1	1				
$\vdots$												
c_{r_k}	$-\nu$	1 or $-\nu$	$\cdots$	1	1		1			1	-1	
c_{r_k+1}	$-\nu$	$-\nu$	$\cdots$	$-\nu$	1 or $-\nu$							
$\vdots$												

Inspection of the matrix shows that row $(k + 2)$ dominates $(k + 1)$ and when the latter is deleted, column k dominates $(k + 1)$. We leave it to the reader to check that when

column $(k + 1)$ is deleted, conditions (8.0.1)–(8.0.3) with $m - 1$ in place of m are satisfied. $\square$

The next theorem deals with point 2 of Fig. 4 when $m > 3$.

THEOREM 8.6 (Class 5A. k , $v \geq 1$). *Let $m > 3$, $v \geq 1$, $1 \leq k \leq m - 2$ and assume (8.0.1)–(8.0.3). Assume further that*

$$(8.6.1) \quad a_{j+2,r_j} = -1 \quad \text{for } 1 \leq j \leq k,$$

$$(8.6.2) \quad a_{1,k} = -1 \quad \text{but } a_{1,k+1} = v,$$

and that (8.5.3) and (8.5.4) hold.

Then optimal strategies P and Q are given by

$$\begin{aligned} p_m &= \left(\sum_{i=0}^{2k} \alpha^i \right)^{-1}, & p_1 &= \alpha^{2k} p_m, \\ p_j &= \alpha^{2j-3} p_m & \text{for } 2 \leq j \leq k+1, \\ p_{r_j} &= \alpha^{2j-2} p_m & \text{for } 2 \leq j \leq k \quad (\text{recall that } r_1 = m), \\ q_j &= \alpha^{2k-2j} p_m & \text{for } 1 \leq j \leq k, \\ q_{r_j-1} &= \alpha^{2k-2j+1} p_m & \text{for } 1 \leq j \leq k, \\ q_{r_k} &= \alpha^{2k} p_m. \end{aligned}$$

The game value is $(\sum_{i=0}^{2k-2} (-\alpha)^i + \alpha^{2k-1} - \alpha^{2k}) p_m$.

Proof. We show that for each i and j , $E(c_i, Q) \leq V \leq E(P, d_j)$, where V is the asserted game value.

From (8.6.2) it follows that $a_{i,k} = -1$ for $1 \leq i \leq k$. By definition of r_k (see (6.0.3)) and by (8.5.3) we have $a_{r_k,k} = 1$ and therefore $a_{i,k} = 1$ for $k+1 \leq i \leq r_k$. Then

$$\begin{aligned} E(P, d_k) &= \sum_i a_{i,k} p_i \\ &= (-\alpha^{2k} - \alpha - \alpha^3 - \dots - \alpha^{2k-3} + \alpha^{2k-1} + 1 + \alpha^2 + \dots + \alpha^{2k-2}) p_m \\ &= V. \end{aligned}$$

Next consider columns j and $j+1$ of A , for $1 \leq j \leq k-1$. By (8.6.2), $a_{1,j+1} = -1$, and it follows that all entries on or above the diagonal in these two columns are -1 . Below the diagonal we have $a_{i,j} = 1$ for $j < i \leq r_j$ and $a_{i,j} = -v$ for $i > r_j$; the same is true with $j+1$ in place of j . In the active rows (i.e., those to which P assigns positive probabilities) the two columns are the same except in rows $j+1$ and r_{j+1} , and we see that

$$\begin{aligned} E(P, d_j) - E(P, d_{j+1}) &= p_{j+1}(a_{j+1,j} - a_{j+1,j+1}) + p_{r_{j+1}}(a_{r_{j+1},j} - a_{r_{j+1},j+1}) \\ &= (2\alpha^{2j-1} - (v+1)\alpha^{2j}) p_m = 0 \end{aligned}$$

since $(v+1)\alpha = 2$. Thus $E(P, d_j) = V$ for $1 \leq j \leq k$. Now consider column j , where $k < j < r_1 = m$. By (8.6.2), $a_{i,j} = v$, and (8.0.3) implies that $a_{i,j} = -1$ for $2 \leq i \leq j$. Equation (8.5.3) and (6.0.3) imply that $a_{i,j} = 1$ for $j < i \leq r_k$. Thus, in the active rows, columns k and j differ only in rows 1 and $k+1$, and $E(P, d_k) - E(P, d_j) = (- (v+1)\alpha^{2k} + 2\alpha^{2k-1}) p_m = 0$. Hence $E(P, d_j) = V$ for $1 \leq j < r_1$.

Next we show that for $1 \leq i \leq k-1$, columns $r_i - 1$ through $r_{i+1} - 1$ are as follows in the first r_k rows:

	$\cdots$	d_{r_i-1}	d_{r_i}	$\cdots$	$d_{r_{i+1}-1}$	$\cdots$
c_1		ν	ν	$\cdots$	ν	
$\vdots$						
c_i		ν	ν	$\cdots$	ν	
c_{i+1}		-1	ν	$\cdots$	ν	
c_{i+2}		-1	-1	$\cdots$	-1	
$\vdots$						
c_{k+1}		-1	-1	$\cdots$	-1	
$\vdots$						
c_{r_i-1}		-1	-1	$\cdots$	-1	
c_{r_i}		1	-1	$\cdots$	-1	
$\vdots$						
c_{r_i+1}		1	1	$\cdots$	1	
$\vdots$						
c_{r_k}		1	1	$\cdots$	1	

In (8.5.4), or in (8.0.3) if $i = 1$, we have that $a_{i+1, r_i} = \nu$. All entries above and to the right must then be ν as well, including $a_{i+1, r_{i+1}-1}$. If $i > 1$ it follows that $a_{i, r_i-1} = \nu$ (because $r_i - 1 \geq r_{i-1}$), and $a_{1, m-1} = \nu$ by (8.6.2) since, $m - 1 > k$. That $a_{i+1, r_i-1} = -1 = a_{i+2, r_{i+1}-1}$ we have from (8.0.4), which applies because of (8.5.3) and the fact that $k \leq m - 2$. This accounts for all the entries on or above the diagonal, i.e., the -1 and ν entries. If $a_{r_k, r_i-1} = 1$ it follows that the remaining entries are all one. This in turn is true if $r_k \leq r_{r_i-1}$, and this follows from (8.5.3) since $k \leq m - 2 = r_1 - 2$. From the displayed columns we see that for $r_i \leq j < r_{i+1}$,

$$E(P, d_{r_i-1}) - E(P, d_j) = (\alpha^{2i-1}(-1 - \nu) + 2\alpha^{2i-2})p_m = 0.$$

Thus we have $E(P, d_j) = V$ for $1 \leq j < r_k$.

We show similarly that columns $r_k - 1$ and beyond are as follows, using (8.5.4), (8.0.3) if necessary, and (8.0.4):

	$\cdots$	d_{r_k-1}	d_{r_k}	d_{r_k+1}	$\cdots$
c_1		ν	ν	ν	
$\vdots$					
c_k		ν	ν	ν	
c_{k+1}		-1	ν	ν	
c_{k+2}		-1	-1	-1 or ν	
$\vdots$					
c_{r_k-1}		-1	-1	-1 or ν	
c_{r_k}		1	-1	-1 or ν	
$\vdots$					

It follows from (6.0.3) and the fact $k < r_k - 1$, that $a_{r_k, r_k-1} = 1$. Now $E(P, d_{r_k-1}) - E(P, d_{r_k}) = (\alpha^{2k-1}(-1 - \nu) + 2\alpha^{2k-2})p_m = 0$, and for $j \geq r_k$, $E(P, d_j) \geq E(P, d_r)$. Thus $E(P, d_j) \geq V$ for all j .

It remains to show that $E(c_i, Q) \leq V$ for all i . Consider row m . By definition of r_1 , $a_{r_1, 1} = 1$ and therefore $a_{r_1, j} = 1$ for $1 \leq j < r_1$. From (8.6.1) we have $d_{r_k} < c_{k+2}T$,

and thus for $j \leq r_k$, $d_j < c_{k+2}T \leq c_{r_1}T$; i.e., $a_{r_1,j} = -1$ for $r_1 \leq j \leq r_k$. This gives us $E(c_{r_1}, Q) = p_m(\alpha^{2k-2} + \alpha^{2k-4} + \dots + 1 + \alpha^{2k-1} - \alpha^{2k-3} - \dots - \alpha - \alpha^{2k}) = V$. By (8.6.2), (8.0.4), and (8.0.3) the first two rows of A are as follows:

	α^{2k-2}				0		α^{2k-1}		
	d_1	d_2	$\dots$	d_k	d_{k+1}	$\dots$	d_{r_1-1}	d_{r_1}	$\dots$
c_1	-1	-1	$\dots$	-1	ν	$\dots$	ν	ν	
c_2	1	-1	$\dots$	-1	-1	$\dots$	-1	ν	

Then $E(c_1, Q) - E(c_2, Q) = (-2\alpha^{2k-2} + (\nu + 1)\alpha^{2k-1})p_m = 0$. From (8.0.4) and (8.5.4) we see that rows j and $j + 1$ of A , for $2 \leq j \leq k$, are as follows:

			α^{2k-2j}						$\alpha^{2k-2j+1}$	
	d_1	$\dots$	d_{j-1}	d_j	d_{j+1}	$\dots$	d_{r_j-1}	d_{r_j}	$\dots$	
$\vdots$										
c_j	1	$\dots$	1	-1	-1	$\dots$	-1	ν	$\dots$	ν
c_{j+1}	1	$\dots$	1	1	-1	$\dots$	-1	-1	$\dots$	-1

Hence, $E(c_j, Q) - E(c_{j+1}, Q) = (-2\alpha^{2k-2j} + (\nu + 1)\alpha^{2k-2j+1})p_m = 0$. Thus $E(c_i, Q) = E(c_{k+1}, Q)$ for $1 \leq i \leq k + 1$.

By hypothesis (8.6.1), $a_{k+2,r_k} = -1$ and it follows that in row i for $k + 2 \leq i \leq r_1$ we have $a_{i,j} = 1$ for $j < i$ and $a_{i,j} = -1$ for $i \leq j \leq r_k$. Thus rows $k + 1$ through r_1 are as follows:

				0		α^{2k-1}			α^{2k}	
	d_1	$\dots$	d_k	d_{k+1}	$\dots$	d_{r_1-1}	d_{r_1}	$\dots$	d_{r_k-1}	d_{r_k}
$\vdots$										
c_{k+1}	1	$\dots$	1	-1	$\dots$	-1	-1	$\dots$	-1	ν
c_{k+2}	1	$\dots$	1	1	$\dots$	± 1	-1	$\dots$	-1	-1
$\vdots$										
c_{r_1}	1	$\dots$	1	1	$\dots$	1	-1	$\dots$	-1	-1
$\vdots$										

Since $q_j = 0$ for $k < j < r_1 - 1$ and for $j > r_k$, we see that $E(c_{k+1}, Q) - E(c_{r_1}, Q) = (-2\alpha^{2k-1} + (\nu + 1)\alpha^{2k})p_m = 0$ and that $E(c_i, Q) \leq E(c_{r_1}, Q)$ for $k + 1 < i < r_1$. Thus $E(c_i, Q) \leq V$ for all $i \leq r_1$.

Next consider rows r_i through r_{i+1} , for $1 \leq i < k$. From (6.0.3) and (8.5.3) we ascertain that these rows are as follows:

			α^{2k-2i}				0		$\alpha^{2k-2i-1}$	
	d_1	$\dots$	d_{i-1}	d_i	d_{i+1}	$\dots$	d_{r_i-1}	d_{r_i}	$\dots$	d_{r_k}
$\vdots$										
c_{r_i}	$-\nu$	$\dots$	$-\nu$	1	1	$\dots$	1	-1	$\dots$	-1
$c_{r_{i+1}}$	$-\nu$	$\dots$	$-\nu$	$-\nu$	1	$\dots$	1	1	$\dots$	-1
$\vdots$										
$c_{r_{i+1}-1}$	$-\nu$	$\dots$	$-\nu$	$-\nu$	1	$\dots$	1	1	$\dots$	-1
$c_{r_{i+1}}$	$-\nu$	$\dots$	$-\nu$	$-\nu$	1	$\dots$	1	1	$\dots$	-1

For $r_i < j < r_{i+1}$, $E(c_j, Q) < E(c_{r_i}, Q)$, and

$$E(c_{r_i}, Q) - E(c_{r_{i+1}}, Q) = ((1 + \nu)\alpha^{2k-2i} - 2\alpha^{2k-2i-1})p_m = 0,$$

so we have $E(c_j, Q) \leq V$ for $j \leq r_k$ and $E(c_{r_i}, Q) = V$ for $1 \leq i \leq k$.

Finally, we examine rows r_k and beyond. These rows, in view of (6.0.3) and (8.5.3) again, are as follows:

	1							α^{2k}	
	d_1	$\cdots$	d_{k-1}	d_k	d_{k+1}	$\cdots$	d_{r_k-1}	d_{r_k}	$\cdots$
$\vdots$									
c_{r_k}	$-\nu$	$\cdots$	$-\nu$	1	1	$\cdots$	1	-1	
$\vdots$									
c_{r_k+j}	$-\nu$	$\cdots$	$-\nu$	$-\nu$	1 or $-\nu$	$\cdots$	1 or $-\nu$	1 or $-\nu$	
$\vdots$									

For $j \geq 1$, $E(c_{r_k+j}, Q) - E(c_{r_k}, Q) \leq (1(-\nu - 1) + 2\alpha^{2k})p_m = (2\alpha^{2k} - 2/\alpha)p_m \leq 0$ since $0 < \alpha \leq 1$. Thus we have $E(c_i, Q) \leq V$ for all i . $\square$

At point 3 of the flow chart we apply the following theorem.

THEOREM 8.7. *Let $m > 3$, $\nu \geq 1$, $1 \leq k \leq m - 2$ and assume conditions (8.0.1)–(8.0.3). Assume further that (8.5.4) and (8.6.1) hold and that*

$$(8.7.1) \quad a_{1,k+1} = -1, \quad \text{and}$$

$$(8.7.2) \quad r_{k+1} = r_k \quad \text{but } r_i < r_{i+1} \quad \text{for } 1 \leq i < k.$$

Then the game may be reduced by dominance to one where (8.0.1)–(8.0.3) hold with $m - 1$ in place of m , or to one that satisfies the hypotheses of Theorem 6.2.

Proof. If $k = 1$, the matrix is as follows:

	d_1	d_2	d_3	$\cdots$	d_{r_1-1}	d_{r_1}	$\cdots$
c_1	-1	-1	-1 or ν	$\cdots$	-1 or ν	ν	
c_2	1	-1	-1	$\cdots$	-1	ν	
c_3	1	1	-1	$\cdots$	-1	-1	
$\vdots$							
c_{r_1}	1	1	1	$\cdots$	1	-1	
c_{r_1+1}	$-\nu$	$-\nu$	1 or $-\nu$	$\cdots$			
$\vdots$							

Column two dominates column one, and when column one is deleted, the first row dominates the second, which we then also delete. In the reduced matrix the hypotheses of Theorem 6.2 are satisfied so the game is of class 3A.

If $k > 1$ our hypotheses imply that the game matrix is in part as follows:

	$\cdots$	d_{k-1}	d_k	d_{k+1}	$\cdots$	d_{r_1-1}	d_{r_1}	$\cdots$	d_{r_k-1}	d_{r_k}	$\cdots$
$\vdots$											
c_k		1	-1	-1	$\cdots$	-1	-1	$\cdots$	-1	ν	$\cdots$
c_{k+1}		1	1	-1	$\cdots$	-1	-1	$\cdots$	-1	-1	ν
$\vdots$											
c_{r_k}		ν	1	1	$\cdots$						
c_{r_k+1}		ν	ν	ν	$\cdots$						

Now column $k + 1$ dominates column k , and when the latter is deleted, row k dominates row $k + 1$. When row $k + 1$ is deleted the matrix satisfies (8.0.1)–(8.0.3) with $m - 1$ in place of m . $\square$

The next theorem applies to the situation at point 4 in the flow chart.

THEOREM 8.8 (Class 3A, $\nu \geq 1$). *Let $m > 3$, $\nu \geq 1$, $1 \leq k \leq m - 2$, and assume (8.0.1)–(8.0.3). Assume further that (8.6.1) and (8.7.1) hold, that*

$$(8.8.1) \quad r_i < r_{i+1} \quad \text{for } 1 \leq i \leq k,$$

$$(8.8.2) \quad a_{j+2, r_{j+1}} = \nu \quad \text{for } j < k \text{ but } a_{k+2, r_{k+1}} = -1.$$

Then the game value is $(1 - \alpha)/(1 + \alpha)$ and optimal strategies P and Q are given by $p_2 = q_{r_{k+1}} = \alpha/(1 + \alpha)$, $p_{r_1} = q_{k+1} = 1/(1 + \alpha)$.

Proof. Our hypotheses imply that the game matrix is as follows:

		1										α	
		d_1	$\cdots$	d_k	d_{k+1}	$\cdots$	d_{r_1-1}	d_{r_1}	$\cdots$	$d_{r_{k+1}-1}$	$d_{r_{k+1}}$	$\cdots$	
α	c_1	-1	$\cdots$	-1	-1	$\cdots$	-1 or ν	ν	$\cdots$	ν	ν		
	c_2	1	$\cdots$	± 1	-1	$\cdots$	-1	ν	$\cdots$	ν	ν		
	$\vdots$												
	c_{k+1}	1	$\cdots$	1	-1	$\cdots$	-1	-1	$\cdots$	ν	ν		
	c_{k+2}	1	$\cdots$	1	1	$\cdots$	-1	-1	$\cdots$	-1	-1		
1	$\vdots$												
	c_{r_1}	1	$\cdots$	1	1	$\cdots$	1	-1	$\cdots$	-1	-1		
	$\vdots$												
	$c_{r_{k+1}}$	$-\nu$	$\cdots$	$-\nu$	1	$\cdots$	1	1	$\cdots$	1	-1		
	$c_{r_{k+1}+1}$	$-\nu$	$\cdots$	$-\nu$	$-\nu$	$\cdots$					1 or $-\nu$		

Let $V = (1 - \alpha)/(1 + \alpha)$. From the matrix we see that $E(P, d_1) = 1 > V$ and for $2 \leq j \leq r_1 - 1$, $E(P, d_j) = V$. For $j \geq r_1$, $E(P, d_j) \geq (\nu\alpha - 1)/(1 + \alpha) = V$, so $E(P, d_j) \geq V$ for all j .

For $1 \leq i \leq k + 1$, $E(c_i, Q) = (-1 + \nu\alpha)/(1 + \alpha) = V$. For $k + 2 \leq i \leq r_{k+1}$, $E(c_i, Q) = V$, and for $i > r_{k+1}$, $E(c_i, Q) \leq (-\nu + \alpha)/(1 + \alpha) \leq V$ since $-\nu + \alpha = 1 - \alpha + 2\alpha - (\nu + 1) = 1 - \alpha + 2(\alpha - 1/\alpha) \leq 1 - \alpha$. $\square$

Finally, at point 5 of the flow chart the next theorem applies.

THEOREM 8.9 (Class 4A.k, $\nu \geq 1$). *Let $m > 3$, $\nu \geq 1$, $k = m - 2$ and assume (8.0.1)–(8.0.3). Assume further that (8.6.1), (8.7.1), (8.8.1), and*

$$(8.9.1) \quad a_{j+2, r_{j+1}} = \nu \quad \text{for } 1 \leq j \leq k,$$

hold. Then optimal strategies P and Q are given by

$$p_m = q_{m-1} = \left(\sum_{i=0}^{2k} \alpha^i \right)^{-1},$$

$$p_i = \alpha^{2i-3} p_m \quad \text{for } 2 \leq i \leq k + 1,$$

$$p_{r_1} = \alpha^{2i-2} p_m \quad \text{for } 1 \leq i \leq k + 1,$$

$$q_j = \alpha^{2k-2j+2} p_m \quad \text{for } 1 \leq j \leq k,$$

$$q_{r_j} = \alpha^{2k-2j+1} p_m \quad \text{for } 1 \leq j \leq k,$$

and the game value is $\sum_{i=0}^{2k} (-\alpha)^i p_m$.

Proof. Let V be the asserted game value. Equation (8.7.1) implies that $a_{i,m-1} = -1$ for all $i \leq m-1$, and the definition of r_{k+1} , (6.0.3), implies that $a_{i,m-1} = 1$ for $m \leq i \leq r_{k+1}$ (since $k+1 = m-1$). Thus

$$E(P, d_{m-1}) = \left(- \sum_{j=1}^k \alpha^{2j-1} + \sum_{j=0}^k \alpha^{2j} \right) p_m = V.$$

Consider columns j and $j+1$ of the matrix A , $1 \leq j \leq k = m-2$. All entries on or above the diagonal are -1 . Below the diagonal we have $a_{i,j} = 1$ for $j < i \leq r_j$ and $a_{i,j} = -\nu$ for $i > r_j$ by definition of r_j . Similarly in column $j+1$, and moreover $a_{r_j+1,j+1} = 1$ because $r_{j+1} > r_j$. Among the rows to which P assigns positive probability, columns j and $j+1$ differ only in rows $j+1$ and r_{j+1} , and

$$E(P, d_j) - E(P, d_{j+1}) = (2\alpha^{2j-1} - (\nu+1)\alpha^{2j})p_m = 0.$$

$E(P, d_j) = V$ for $1 \leq j \leq m-1$. Columns m and $m-1$ differ only in rows two and m of the rows active in P ; $a_{2,m-1} = -1$, $a_{2,m} = \nu$ (by (8.0.3)), $a_{m,m-1} = 1$ and $a_{m,m} = -1$. Thus

$$E(P, d_{m-1}) - E(P, d_m) = (\alpha(-1-\nu) + 2)p_m = 0,$$

and we have $E(P, d_j) = V$ for $1 \leq j \leq m = r_1$.

Next we examine columns r_t through r_{t+1} , where $t < k$. By (8.9.1), $a_{t+1,r_t} = a_{t+2,r_{t+1}} = \nu$, and it follows that the top $t+1$ elements of these columns are ν . From (8.0.4) and (8.8.1) we may infer that $a_{i,j} = -1$ for $r_t \leq j \leq r_{t+1}-1$ and $t+2 \leq i \leq j$, and that $a_{i,r_{t+1}} = -1$ for $t+3 \leq i \leq r_{t+1}$. Below the diagonal but in the first r_{k+1} rows, all the entries in these columns are 1 by (6.0.3). If $r_t < j < r_{t+1}$, columns j and r_{t+1} differ only in rows $t+2$ and r_{t+1} among rows active in P .

$$\begin{aligned} E(P, d_j) - E(P, d_{r_{t+1}}) &= \alpha^{2t+1} p_m (a_{t+2,j} - a_{t+2,r_{t+1}}) + \alpha^{2t} p_m (a_{r_{t+1},j} - a_{r_{t+1},r_{t+1}}) \\ &= (\alpha^{2t+1}(-1-\nu) + 2\alpha^{2t}) p_m = 0. \end{aligned}$$

Thus $E(P, d_j) = V$ for $1 \leq j \leq r_k$.

As we infer from (8.9.1) and (8.0.4), in columns r_k through r_{k+1} the relevant parts are as follows:

		$\cdots$	d_{r_k}	$\cdots$	$d_{r_{k+1}-1}$	$d_{r_{k+1}}$	$\cdots$
	$\vdots$						
	c_{k+1}	$\cdots$	ν	$\cdots$	ν	ν	
1	c_{k+2}	$\cdots$	-1	$\cdots$	-1	ν	
	c_{k+3}	$\cdots$	-1	$\cdots$	-1	-1 or ν	
	$\vdots$						
α^{2k}	$c_{r_{k+1}}$	$\cdots$	1	$\cdots$	1	-1 or ν	
	$\vdots$						

From this we see that for $r_k \leq j < r_{k+1}$, $E(P, d_j) = E(P, d_{r_k})$, and for $j \geq r_{k+1}$, $E(P, d_j) - E(P, d_{r_k}) \geq ((\nu+1) - 2\alpha^{2k})p_m = (1 - \alpha^{2k+1})2p_m/\alpha \geq 0$, so we have $E(P, d_j) \geq V$ for all j .

It remains to show that $E(c_i, Q) \leq V$ for all i . From conditions (8.7.1), (8.0.1), and (8.0.3) we see that $E(c_1, Q) < E(c_2, Q)$ and that the first two rows of A are

as follows:

	α^{2k}						
	d_1	d_2	$\cdots$	d_{m-1}	d_m	$\cdots$	
c_1	-1	-1	$\cdots$	-1	ν		
c_2	1	-1	$\cdots$	-1	ν		

For $2 \leq i \leq m-1$, rows i and $i+1$ are as follows:

	d_1	$\cdots$	d_{i-1}	d_i	d_{i+1}	$\cdots$	$d_{r_{i-1}-1}$	d_{r_i-1}	$\cdots$	d_{r_i-1}	d_{r_i}	$\cdots$
$\vdots$												
c_i	1	$\cdots$	1	-1	-1	$\cdots$	-1	ν	$\cdots$	ν	ν	
c_{i+1}	1	$\cdots$	1	1	-1	$\cdots$	-1	-1	$\cdots$	-1	ν	

Then $E(c_i, Q) - E(c_{i+1}, Q) = (\alpha^{2k-2i+2}(-2) + \alpha^{2k-2i+3}(\nu+1))p_m = 0$. The case $i+1 = m$ in the table gives us $E(c_m, Q) = \sum_{j=1}^{m-1} q_j - \sum_{j=1}^{m-2} q_{r_j} = V$, so we have $E(c_1, Q) < E(c_i, Q) = V$ for $2 \leq i \leq m$.

Rows m to r_2 are as follows:

	α^{2k}	$\alpha^{2k-1} \underbrace{0}_{d_{r_2-1}}$									
	d_1	d_2	$\cdots$	d_{m-1}	d_m	$\cdots$	d_{r_2-1}	d_{r_2}	$\cdots$	d_{r_k}	$\cdots$
$\vdots$											
c_m	1	1	$\cdots$	1	-1	$\cdots$	-1	-1	$\cdots$	-1	
c_{m+1}	$-\nu$	1	$\cdots$	1	1	$\cdots$	-1	-1	$\cdots$	-1	
$\vdots$											
c_{r_2}	$-\nu$	1	$\cdots$	1	1	$\cdots$	1	-1	$\cdots$	-1	
$\vdots$											

If $m+1 \leq i \leq r_2$, $E(c_m, Q) - E(c_i, Q) = (\alpha^{2k}(1+\nu) - 2\alpha^{2k-1})p_m = 0$, and thus $E(c_i, Q) = V$.

If $2 \leq i \leq k-1$, rows r_i to r_{i+1} are as follows:

	$\cdots$	d_{i-1}	d_i	d_{i+1}	$\cdots$	d_{r_i-1}	d_{r_i}	$\cdots$	$d_{r_{i+1}-1}$	$d_{r_{i+1}}$	$\cdots$	d_{r_k}	$\cdots$
$\vdots$													
c_{r_i}	$\cdots$	$-\nu$	1	1	$\cdots$	1	-1	$\cdots$	-1	-1	$\cdots$	-1	
c_{r_i+1}	$\cdots$	$-\nu$	$-\nu$	1	$\cdots$	1	1	$\cdots$	-1	-1	$\cdots$	-1	
$\vdots$													
$c_{r_{i+1}}$	$\cdots$	$-\nu$	$-\nu$	1	$\cdots$	1	1	$\cdots$	1	-1	$\cdots$	-1	
$\vdots$													

For $r_i < j \leq r_{i+1}$, $E(c_{r_i}, Q) - E(c_j, Q) = (q_i(1+\nu) - 2q_{r_i}) = 0$, so we now have $E(c_i, Q) = V$ for $2 \leq i \leq r_k$.

Rows r_k and beyond are as follows:

	$\cdots$	d_{k-1}	d_k	d_{k+1}	$\cdots$	d_{r_k-1}	d_{r_k}	$\cdots$
$\vdots$								
c_{r_k}	$\cdots$	$-\nu$	1	1	$\cdots$	1	-1	
c_{r_k+1}	$\cdots$	$-\nu$	$-\nu$	1	$\cdots$	1	1	
$\vdots$								
$c_{r_{k+1}}$	$\cdots$	$-\nu$	$-\nu$	1	$\cdots$	1	1	
$c_{r_{k+1}+1}$	$\cdots$	$-\nu$	$-\nu$	$-\nu$	$\cdots$	1 or $-\nu$	1 or $-\nu$	

As above we see that $E(c_i, Q) = E(c_{r_k}, Q)$ for $r_k < i \leq r_{k+1}$, and for $i > r_{k+1}$, $E(c_i, Q) \leq E(c_{r_{k+1}}, Q) = V$. Thus $E(c_i, Q) \leq V$ for all i . $\square$

This completes the proof that the solutions for all $m \geq 3$ are as indicated in the flow chart in Fig. 4.

9. Class 4B. In this section we give the solution for the remaining games with $\nu \geq 1$. If $T \leq T^*$, the solution is given in Theorem 3.1 or 4.4. If $T > T^*$ and T is in some $J_{m,1}$, the solution is given in Theorem 6.2 or in § 8. If T (greater than T^*) is in no $J_{m,1}$ but in some $J_{m,n}$ with $2 \leq n \leq m_1 < m$, where m_1 is the integer such that T falls between $J_{m,1}$ and $J_{m,1+1}$, the solution is given in Theorem 7.1. There remains the case where T (greater than T^*) is neither in any $J_{m,1}$ nor in any $J_{m,n}$ with $2 \leq n \leq m_1 < m$. Note that $m > m_1$ is not a restriction here. $T > d_{m_1}/c_1$, the right endpoint of $J_{m,1}$, will not allow T to lie in any $J_{m,n}$ with $2 \leq n$ and $m \leq m_1$.

For the remaining case described above, there is for each $n \leq m_1$ an integer m_n such that T falls between $J_{m_n,n}$ and $J_{m_n+1,n}$:

$$(9.0.1) \quad d_{m_n}/c_n < T \leq c_{m_n+1}/d_n, \quad 1 \leq n \leq m_1.$$

Note that $m_k \leq m_{k+1}$ for each k , since $c_{m_k}/d_{k+1} < d_{m_k}/c_k < T \leq c_{m_{k+1}+1}/d_{k+1}$ and therefore $m_k < m_{k+1} + 1$. Since m_1 is defined by the condition $d_{m_1}/c_1 < T \leq c_{m_1+1}/d_1$, and since $T > T^* \geq c_2/d_1$, we must have $m_1 + 1 > 2$. Thus we have $2 \leq m_1 \leq m_2 \leq \cdots \leq m_{m_1}$. Any subset of these m_i inequalities may in fact be equalities; examples are easy to construct. However if there is equality throughout, the game is in class 2B, as we show first.

THEOREM 9.1 (Class 2B, all $\nu > 0$). *With m_n defined by (9.0.1), suppose that $m_{m_1} = m_1$ (and thus all m_n , $1 \leq n \leq m_1$, are equal). Then the hypotheses of Theorems 4.4 or 4.5 (according as $\nu \geq 1$ or $\nu < 1$) are satisfied and the game is in class 2B.*

Proof. From (9.0.1) and the equality $m_{m_1} = m_1$ we have immediately that $d_{m_1}/c_1 < T \leq c_{m_{m_1}+1}/d_{m_1} = c_{m_1+1}/d_{m_1}$. Then the hypotheses of Theorem 4.4 or 4.5 with $S'_1 = \{c_1, \dots, c_{m_1}\}$ and $d_{m_1} \in S_2^*$ are fulfilled. $\square$

Then we need be concerned only with the case where $m_{m_1} > m_1$. Let $m_{t_1} < m_{t_2} < \cdots < m_{t_{k+1}} = m_{m_1}$ be the distinct m_n s which occur, where in each instance the subscript t_i is minimal: $1 \leq n < t_i \Rightarrow m_n < m_{t_i}$. Thus we always have $1 = t_1 < t_2 < \cdots < t_{k+1} \leq m_1$, and $1 \leq k \leq m_1 - 1$. For example, if $m_1 = 6$, $m_2 = m_3 = 8$, $m_4 = 9$ and $m_5 = m_6 = 12$, then $k + 1 = 4$, $t_1 = 1$, $t_2 = 2$, $t_3 = 4$ and $t_4 = 5$.

The following theorem then applies.

THEOREM 9.2 (Class 4B.k, $\nu \geq 1$). *Let S_1 , S_2 , T^* , and $J_{m,n}$ be as described in (6.0.1) and (6.0.2), $\nu \geq 1$, and assume that $T > T^*$ and that T falls in no $J_{m,n}$ with $n \leq m_1$ (m_i defined by (9.0.1)). Let $2 \leq m_1 = m_{t_1} < m_{t_2} < \cdots < m_{t_{k+1}} = m_{m_1}$ as described in the paragraph above. Then the game value is $-p \sum_{i=0}^{2k} (-\alpha)^i$, where $p = (\sum_{i=0}^{2k} \alpha^i)^{-1}$,*

optimal strategies P and Q are given by

$$\begin{aligned} p_{t_i} &= p\alpha^{2k-2i+2}, & 1 \leq i \leq k+1, \\ p_{1+m_i} &= p\alpha^{2k-2i+1}, & 1 \leq i \leq k, \\ q_{t_i-1} &= p\alpha^{2i-3}, & 2 \leq i \leq k+1, \\ q_{m_i} &= p\alpha^{2i-2}, & 1 \leq i \leq k+1. \end{aligned}$$

Proof. Let V be the asserted game value. We have, as usual, to show that $E(c_i, Q) \leq V \leq E(P, d_j)$ for each i and j . By definition of the t_i , $m_j = m_{t_i}$ for $t_i \leq j < t_{i+1}$. From this and (9.0.1) we see that

$$(9.2.1) \quad d_{m_i}/c_{t_i} = d_{m_j}/c_{t_i} < T \leq c_{m_{t_i+1}}/d_{t_{i+1}-1} \quad \text{for } t_i \leq j < t_{i+1}.$$

Consider first r in the range $1 \leq r \leq m_1$. For each s in the range $r \leq s \leq m_r$ we have $c_r < d_s \leq d_{m_r}$, and by (9.2.1) $d_{m_r} < c_r T$, so q_s is in the third sum in (6.1.2). Each q_i with $1 \leq i < r$ will be in one of the three sums, and $2\nu \geq \nu + 1 > \nu - 1$. Thus $E(c_r, Q) \leq \nu - (\nu - 1)(q_1 + \cdots + q_{r-1}) - (\nu + 1)(q_r + \cdots + q_{m_1} + \cdots + q_{m_r})$. If $r = 1$, we have

$$\begin{aligned} E(c_1, Q) &\leq \nu - (\nu + 1) \sum_{i=1}^{m_1} q_i = \nu - (\nu + 1) \left(\sum_{i=1}^k \alpha^{2i-1} + 1 \right) p \\ &= 2/\alpha - 1 - (2p/\alpha) \left(1 + \sum_{i=1}^k \alpha^{2i-1} \right) \\ &= (p/\alpha) \left((2-\alpha)/p - 2 \left(1 + \sum_{i=1}^k \alpha^{2i-1} \right) \right) \\ &= (p/\alpha) \left((2-\alpha) \sum_{i=0}^{2k} \alpha^i - 2 \left(1 + \sum_{i=1}^k \alpha^{2i-1} \right) \right) = V \end{aligned}$$

as we see by multiplying out and collecting terms in the second factor. If $r > 1$, we have

$$\begin{aligned} E(c_r, Q) &\leq \nu - (\nu - 1)p \sum_{2 \leq t_i \leq r} \alpha^{2i-3} - (\nu + 1)p \left(\sum_{r < t_i \leq t_{k+1}} \alpha^{2i-3} + \sum_{1 \leq t_i \leq r} \alpha^{2i-2} \right) \\ &= \nu - (\nu - 1)p \sum_{i=2}^x \alpha^{2i-3} - (\nu + 1)p \left(\sum_{i=x+1}^{k+1} \alpha^{2i-3} + \sum_{i=1}^x \alpha^{2i-2} \right) \end{aligned}$$

for some $x \leq k + 1$. Thus

$$\begin{aligned} E(c_r, Q) &\leq \nu - (\nu + 1)(1 - \alpha)p \sum_{i=2}^x \alpha^{2i-3} - (\nu + 1)p \sum_{i=2}^{k+1} \alpha^{2i-3} - (\nu + 1)p \sum_{i=1}^x \alpha^{2i-2} \\ &= 2/\alpha - 1 - (2p/\alpha) \left(\sum_{i=1}^{2x-2} (-1)^{i+1} \alpha^i + \sum_{i=x+1}^{k+1} \alpha^{2i-3} + \sum_{i=1}^x \alpha^{2i-2} \right) \\ &= (p/\alpha) \left((2-\alpha)/p + 2 \sum_{i=1}^{2x-2} (-\alpha)^i - 2 \sum_{i=x+1}^{k+1} \alpha^{2i-3} - 2 \sum_{i=1}^x \alpha^{2i-2} \right) \\ &= (p/\alpha) \left((2-\alpha) \sum_{i=0}^{2k} \alpha^i + 2 \sum_{i=1}^{2x-2} (-\alpha)^i - 2 \sum_{i=x+1}^{k+1} \alpha^{2i-3} - 2 \sum_{i=1}^x \alpha^{2i-2} \right). \end{aligned}$$

The last three sums combine to give $-2 - 2 \sum_{i=2}^{k+1} \alpha^{2i-3}$ and the first is

$$(1 + (1 - \alpha)) \sum_{i=0}^{2k} \alpha^i = \sum_{i=0}^{2k} \alpha^i + 1 - \alpha^{2k+1} = 2 + \sum_{i=1}^{2k} \alpha^i - \alpha^{2k+1}.$$

Thus $E(c_r, Q) \leq (p/\alpha) \sum_{i=1}^{2k+1} (-\alpha)^i = V$.

Next consider r in the range $m_1 < r \leq m_{l_{k+1}} = m_{m_1}$. Let $j = j(r)$ be the integer such that $m_{t_j} < r \leq m_{t_{j+1}}$. Then $r \geq m_{t_j} + 1$, and from the last inequality in (9.2.1) we have $d_{t_{j+1}-1} \leq c_{m_{t_j}+1}/T \leq c_r/T$. This implies that $q_{t_{j+1}-1}$ is in the first sum in (6.1.2). We show now that for $t_{j+1} \leq s \leq m_1$ we have q_{m_s} in the third sum. For this it suffices to show that $c_r < d_{m_s} < c_r T$. However,

$$c_r \leq c_{m_{t_{j+1}}} \leq c_{m_s} < d_{m_s} \leq d_{m_{m_1}} < c_{m_1} T < c_r T.$$

Thus

$$\begin{aligned} E(c_r, Q) &\leq \nu - 2\nu \sum_{i=1}^{t_{j+1}-1} q_i - (\nu - 1) \sum_{i=t_{j+1}}^{m_{t_{j+1}}-1} q_i - (\nu + 1) \sum_{i=m_{t_{j+1}}}^{m_{m_1}} q_i \\ &= \nu - 2\nu p \sum_{i=2}^{j+1} \alpha^{2i-3} - (\nu - 1)p \left(\sum_{i=j+2}^{k+1} \alpha^{2i-3} + \sum_{i=1}^j \alpha^{2i-2} \right) - (\nu + 1)p \sum_{i=j+1}^{k+1} \alpha^{2i-2} \\ &= 2/\alpha - 1 - 2(2/\alpha - 1)p \sum_{i=2}^{j+1} \alpha^{2i-3} - 2(1 - \alpha)(p/\alpha) \left(\sum_{i=j+2}^{k+1} \alpha^{2i-3} + \sum_{i=1}^j \alpha^{2i-2} \right) \\ &\quad - (2p/\alpha) \sum_{i=j+1}^{k+1} \alpha^{2i-2} \\ &= (p/\alpha) \left((2 - \alpha)/p - 2(2 - \alpha) \sum_{i=2}^{j+1} \alpha^{2i-3} - 2(1 - \alpha) \left(\sum_{i=j+2}^{k+1} \alpha^{2i-3} + \sum_{i=1}^j \alpha^{2i-2} \right) \right. \\ &\quad \left. - 2 \sum_{i=j+1}^{k+1} \alpha^{2i-2} \right) \\ &= (p/\alpha) \left(1 + (1 - \alpha) \sum_{i=0}^{2k} \alpha^i - 2(2 - \alpha) \sum_{i=2}^{j+1} \alpha^{2i-3} \right. \\ &\quad \left. - 2(1 - \alpha) \sum_{i=j+2}^{k+1} \alpha^{2i-3} - 2(1 - \alpha) \sum_{i=1}^j \alpha^{2i-2} - 2 \sum_{i=j+1}^{k+1} \alpha^{2i-2} \right) \\ &= (p/\alpha) \left(\sum_{i=0}^{2k} \alpha^i + (1 - \alpha^{2k+1}) - 4 \sum_{i=2}^{j+1} \alpha^{2i-3} \right. \\ &\quad \left. + 2 \sum_{i=2}^{j+1} \alpha^{2i-2} - 2 \sum_{i=j+2}^{k+1} \alpha^{2i-3} + 2 \sum_{i=j+2}^{k+1} \alpha^{2i-2} \right. \\ &\quad \left. - 2 \sum_{i=1}^j \alpha^{2i-2} + 2 \sum_{i=1}^j \alpha^{2i-1} - 2 \sum_{i=j+1}^{k+1} \alpha^{2i-2} \right) \\ &= (p/\alpha) \left(1 + \sum_{i=1}^{2k} \alpha^i + 1 - \alpha^{2k+1} - 2 \sum_{i=1}^j \alpha^{2i-1} + 2 \sum_{i=2}^{k+1} \alpha^{2i-2} \right. \\ &\quad \left. - 2 \sum_{i=1}^{k+1} \alpha^{2i-2} - 2 \sum_{i=j+1}^k \alpha^{2i-1} \right) \\ &= (p/\alpha) \left(2 + \sum_{i=1}^{2k} \alpha^i - \alpha^{2k+1} - 2 \sum_{i=1}^k \alpha^{2i-1} - 2 \right) = V. \end{aligned}$$

$\sum_{i=j+2}^{k+1}$ is of course understood to be zero if $j = k$.

If $r > m_{m_1}$ then $d_{m_1} = c_r d_{m_1} / c_r \leq c_r d_{m_1} / c_{m_{m_1+1}} \leq c_r / T$, by the definition of m_{m_1} (9.0.1). It follows that for $1 \leq s \leq m_1$, $d_s \leq c_r / T$ and therefore that q_s is in the first sum in (6.1.2). Thus $E(c_r, Q) \leq \nu - 2\nu \sum_{i=1}^{m_1} q_i = \nu - 2\nu p (\sum_{i=2}^{k+1} \alpha^{2i-3} + \sum_{i=1}^{k+1} \alpha^{2i-2}) = \nu - 2\nu p p^{-1} = -\nu \leq -1 < V$, and we have $E(c_r, Q) \leq V$ for all r .

We turn now to $E(P, d_s)$. For $1 \leq s \leq m_1$, each p_r with $1 \leq r \leq s$ is in one of the first two sums in (6.1.1) because $c_r \leq c_s < d_s$. Also, each p_r with $s+1 \leq r \leq m_s$ is in the third sum since $d_s < c_{s+1} \leq c_{m_s} < d_{m_s} < c_s T < d_s T$. Thus

$$\begin{aligned} E(P, d_s) &\geq (\nu - 1) \sum_{r=1}^s p_r + (\nu + 1) \sum_{r=s+1}^{m_s} p_r - \nu \\ &= (\nu - 1)p \sum_{i=1}^j \alpha^{2k-2i+2} + (\nu + 1)p \left(\sum_{i=j+1}^{k+1} \alpha^{2k-2i+2} + \sum_{i=1}^{j-1} \alpha^{2k-2i+1} \right) - \nu, \end{aligned}$$

where j is the largest integer such that $t_j \leq s$. Then

$$\begin{aligned} E(P, d_s) &\geq (p/\alpha) \left(2(1-\alpha) \sum_{i=1}^j \alpha^{2k-2i+2} + 2 \sum_{i=j+1}^{k+1} \alpha^{2k-2i+2} \right. \\ &\quad \left. + 2 \sum_{i=1}^{j-1} \alpha^{2k-2i+1} - (2-\alpha)p^{-1} \right) \\ &= (p/\alpha) \left(2 \sum_{i=1}^{k+1} \alpha^{2k-2i+2} - 2 \sum_{i=1}^j \alpha^{2k-2i+3} \right. \\ &\quad \left. + 2 \sum_{i=1}^{j-1} \alpha^{2k-2i+1} - (1+(1-\alpha)) \sum_{i=0}^{2k} \alpha^i \right) \\ &= (p/\alpha) \left(2 \sum_{i=0}^k \alpha^{2i} - 2\alpha^{2k+1} - \sum_{i=0}^{2k} \alpha^i - (1-\alpha^{2k+1}) \right) \\ &= (p/\alpha) \left(2 \sum_{i=1}^k \alpha^{2i} - \sum_{i=1}^{2k+1} \alpha^i \right) = V. \end{aligned}$$

Next consider s in the range $m_1 < s \leq m_{t_k}$, and let j be the integer such that $m_{t_j} < s \leq m_{t_{j+1}}$. Thus $1 \leq j \leq k-1$. For $1 \leq r \leq t_{j+1}-1$,

$$c_r \leq c_{t_{j+1}-1} < d_{t_{j+1}-1} < d_s d_{t_{j+1}-1} / c_s \leq d_s d_{t_{j+1}-1} / c_{m_{t_j+1}} \leq d_s / T$$

by (9.2.1), so p_r is in the first sum in (6.1.1). We now show that for $1 + m_{t_{j+1}} \leq r \leq m_{t_{k+1}}$ we have $d_s < c_r < d_s T$, so that p_r is in the third sum. Because $s \leq m_{t_{j+1}}$,

$$d_s < c_{m_{t_{j+1}}+1} \leq c_r \leq c_{m_{t_{k+1}}} < d_{m_{t_{k+1}}} = d_{m_{m_1}} < c_{m_1} T < d_s T.$$

Thus

$$\begin{aligned} E(P, d_s) &\geq 2\nu \sum_{r=1}^{t_{j+1}-1} p_r + (\nu - 1) \sum_{r=t_{j+1}}^{m_{t_{j+1}}} p_r + (\nu + 1) \sum_{r=1+m_{t_{j+1}}}^{m_{t_{k+1}}} p_r - \nu = 2\nu p \sum_{i=1}^j \alpha^{2k-2i+2} \\ &\quad + (\nu - 1)p \left(\sum_{i=j+1}^{k+1} \alpha^{2k-2i+2} + \sum_{i=1}^j \alpha^{2k-2i+1} \right) + (\nu + 1)p \sum_{i=j+1}^k \alpha^{2k-2i+1} - \nu \end{aligned}$$

$$\begin{aligned}
&= (p/\alpha) \left(2(2-\alpha) \sum_{i=1}^j \alpha^{2k-2i+2} + 2(1-\alpha) \left(\sum_{i=j+1}^{k+1} \alpha^{2k-2i+2} + \sum_{i=1}^j \alpha^{2k-2i+1} \right) \right. \\
&\quad \left. + 2 \sum_{i=j+1}^k \alpha^{2k-2i+1} - (1+(1-\alpha))p^{-1} \right) \\
&= (p/\alpha) \left(4 \sum_{i=1}^j \alpha^{2k-2i+2} - 2 \sum_{i=1}^j \alpha^{2k-2i+3} + 2 \sum_{i=j+1}^{k+1} \alpha^{2k-2i+2} \right. \\
&\quad - 2 \sum_{i=j+1}^{k+1} \alpha^{2k-2i+3} + 2 \sum_{i=1}^j \alpha^{2k-2i+1} - 2 \sum_{i=1}^j \alpha^{2k-2i+2} \\
&\quad \left. + 2 \sum_{i=j+1}^k \alpha^{2k-2i+1} - \sum_{i=0}^{2k} \alpha^i - (1-\alpha^{2k+1}) \right) \\
&= (p/\alpha) \left(2 \sum_{i=1}^{k+1} \alpha^{2k-2i+2} - 2 \sum_{i=1}^{k+1} \alpha^{2k-2i+3} + 2 \sum_{i=1}^k \alpha^{2k-2i+1} \right. \\
&\quad \left. - 2 \sum_{i=0}^{2k} \alpha^i - 1 + \alpha^{2k+1} \right) \\
&= (p/\alpha) \left(2 + 2 \sum_{i=1}^k \alpha^{2i} - 2\alpha^{2k+1} - 1 - \sum_{i=1}^{2k} \alpha^i - 1 + \alpha^{2k+1} \right) = V.
\end{aligned}$$

Finally, for $s > m_{t_k}$ we have $c_{t_k} < d_{t_k} < d_s d_{t_k} / c_s \leq d_s d_{t_k} / c_{m_{t_k}+1} \leq d_s / T$ by the definition of m_{t_k} , so p_{t_k} is in the first sum in (6.1.1). Also, each p_r with $c_r \leq d_s T$ is in one of the three sums, and this is true for each $r \leq m_{t_{k+1}}$, as we now show:

$$c_r \leq c_{m_{t_{k+1}}} \leq d_{m_{t_{k+1}}} < c_{t_{k+1}} T < d_{t_{k+1}} T.$$

But $t_{k+1} \leq m_1 \leq m_{t_k}$ so $d_{t_{k+1}} T \leq d_{m_{t_k}} T < d_s T$. Thus

$$\begin{aligned}
E(P, d_s) &\geq 2\nu \sum_{i=1}^{t_k} p_i + (\nu-1) \sum_{i=t_{k+1}}^{m_{t_k}} p_i - \nu \\
&= (2(2-\alpha)/\alpha)p \sum_{i=1}^k \alpha^{2k-2i+2} + (2(1-\alpha)/\alpha)p \left(1 + \sum_{i=1}^k \alpha^{2k-2i+1} \right) - (2-\alpha)/\alpha \\
&= (p/\alpha) \left(4 \sum_{i=1}^k \alpha^{2i} - 2 \sum_{i=1}^k \alpha^{2i+1} + 2 - 2\alpha + 2 \sum_{i=0}^{k-1} \alpha^{2i+1} - 2 \sum_{i=1}^k \alpha^{2i} \right. \\
&\quad \left. - (1+(1-\alpha))p^{-1} \right) \\
&= (p/\alpha) \left(2 \sum_{i=1}^k \alpha^{2i} - 2\alpha^{2k+1} + 2 - \sum_{i=0}^{2k} \alpha^i - (1-\alpha^{2k+1}) \right) \\
&= (p/\alpha) \left(2 \sum_{i=1}^k \alpha^{2i} - \sum_{i=1}^{2k+1} \alpha^i \right) = V.
\end{aligned}$$

Thus we have $E(P, d_s) \geq V$ for all s , and the proof is complete. $\square$

10. Summary of classification for $\nu \geq 1$ and examples. We begin by giving a simple scheme outlining the classification for $\nu \geq 1$. This scheme may be regarded as providing an algorithm for determining the class, and hence the solution, for any game with $\nu \geq 1$, given S_1 , S_2 , and T . It is straightforward to check the validity of this scheme on the basis of the theorems as referenced in the scheme.

(A) $T \leq T^* = \max \{d_1/c_1, c_2/d_1\}$.

(i) $1 < T \leq d_1/c_1$. Class 1A; Theorem 3.1.

(ii) $d_1/c_1 < T \leq c_2/d_1$. Class 2B; Theorem 4.4.

(B) $T > T^*$. Determine $r_1: c_{r_1} < d_1 T \leq c_{r_1+1}$.

For simplicity here we write r for r_1 .

(i) $c_r T \leq d_r$. Class 2A; Theorem 4.1.

(ii) $c_2 T \leq d_r < c_r T$. Refer to the flow chart, Fig. 4; theorems of § 8.

(iii) $c_1 T \leq d_r < c_2 T$. Class 3A; Theorem 6.2.

(iv) $d_r < c_1 T$ (i.e., T in no $J_{m,1}$)

(a) $c_m/d_n < T \leq d_m/c_n$ (i.e., T in $J_{m,n}$) for some m, n with $2 \leq n \leq r$. Class 3B; Theorem 7.1. (Note that $c_m < d_n T$ places an upper bound on m .)

(b) T in no such $J_{m,n}$.

(1) $c_{r+1} \geq d_r T$. Class 2B; Theorem 9.1.

(2) $c_{r+1} < d_r T$. Class 4B; Theorem 9.2.

Observe that for fixed S_1 and S_2 and a given $T = \bar{T}$ there is always a left neighborhood $\bar{T} - \varepsilon < T \leq \bar{T}$ of $\bar{T}$ throughout which the game class is unchanged. The T -axis is partitioned into a union of left open intervals in each of which the game class is constant. This allows a simple graphical depiction of the classification as a function of T for given S_1 and S_2 as follows. We construct a bar of height y across an interval in which the game class is yA and of height $-y$ where the class is yB . We call this graph the classification profile for $S_1 \times S_2$. Profiles for a few examples follow.

Example 1. Odds versus evens. $c_n = 2n - 1$, $d_n = 2n$.

$$1 < T \leq 2 \quad 1A$$

$$2 < T \quad 3A$$

The profile is shown in Fig. 5.

Example 2. Let $c_n = 10n - 2$, $d_n = 10n$. Then $d_1/c_1 = 1.25$, $c_2/d_1 = 1.80$, so we have class 1A for $1 < T \leq 1.25$, and class 2B for $1.25 < T \leq 1.80$, by Theorems 3.1 and 4.4. For $T > T^* = 1.80$ we begin by calculating the intervals $J_{m,1}$ of Theorem 6.2. $J_{2,1} = (1.8, 2.5]$, $J_{3,1} = (2.8, 3.75]$, $J_{4,1} = (3.8, 5.00]$ and for $m \geq 4$, $J_{m+1,1}$ overlaps $J_{m,1}$. Note that when T is in $J_{m,1}$, $T > c_m/d_1 = (10m - 2)/10 > 10m/18 = d_m/c_2$ so that (6.2.3) is satisfied. Thus when T is in any of the intervals $(1.8, 2.5]$, $(2.8, 3.75]$, $(3.8, \infty)$ the game is in class 3A. This leaves only $(2.5, 2.8]$ and $(3.75, 3.8]$ unclassified. Next we turn to Theorem 7.1. For $2.5 < T \leq 2.8$, $m_1 = 2$, and $J_{5,2} = (2.4, \frac{25}{9}]$. Then by Theorem 7.1 we have class 3B for $2.5 < T \leq \frac{25}{9}$. $J_{7,2} = (3.4, \frac{35}{9}]$ covers the gap $(3.75, 3.8]$ so for $3.75 < T \leq 3.8$ we have class 3B again. This leaves only $(\frac{25}{9}, 2.8]$, which lies in no $J_{m,2}$. Here $m_1 = 2$ and $m_2 = 5$, and Theorem 9.2 tells us we have class 4B.1. The profile is shown in Fig. 6.

The determination of the profile in the above example was not based directly on the classification scheme described at the beginning of this section. We may construct an algorithm based directly on that scheme that lends itself more readily to machine determination of the profile. For this purpose we refer to Figs. 7(a) and 7(b), where the

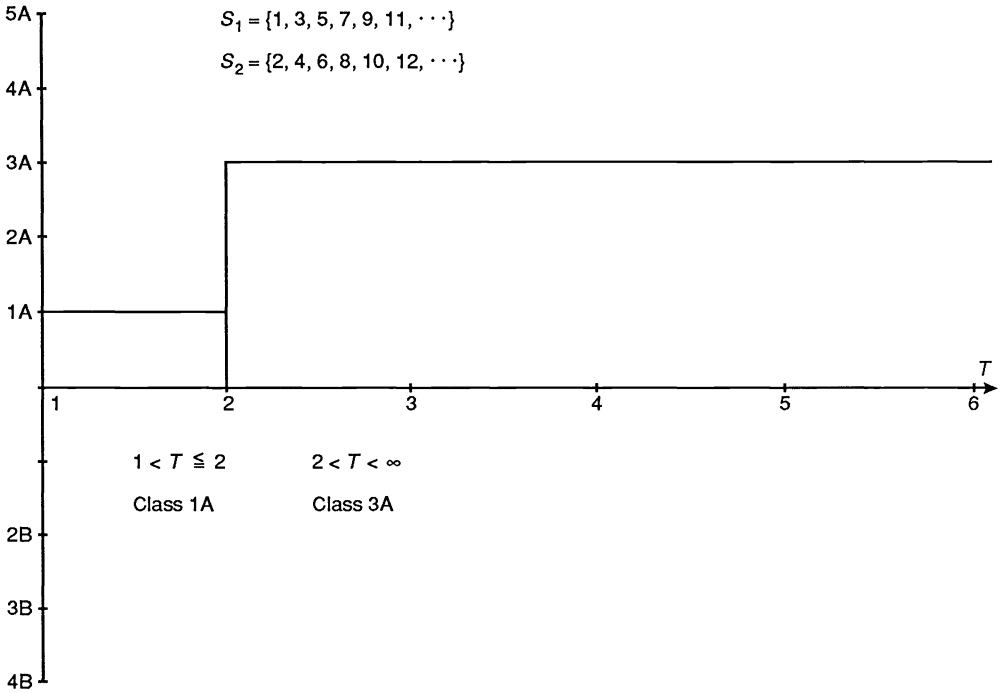


FIG. 5

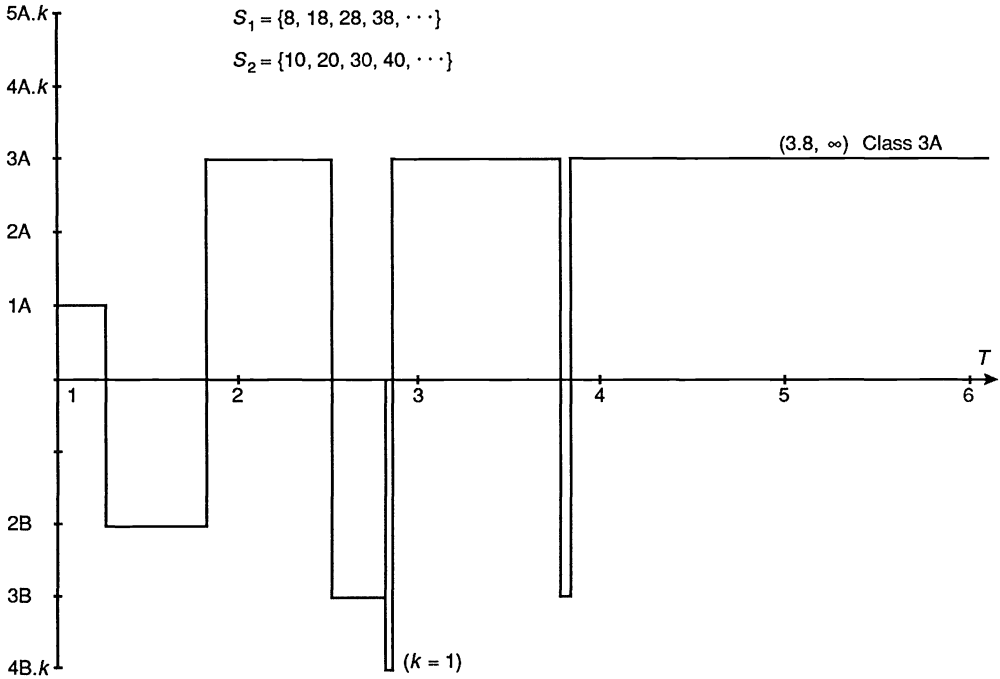


FIG. 6

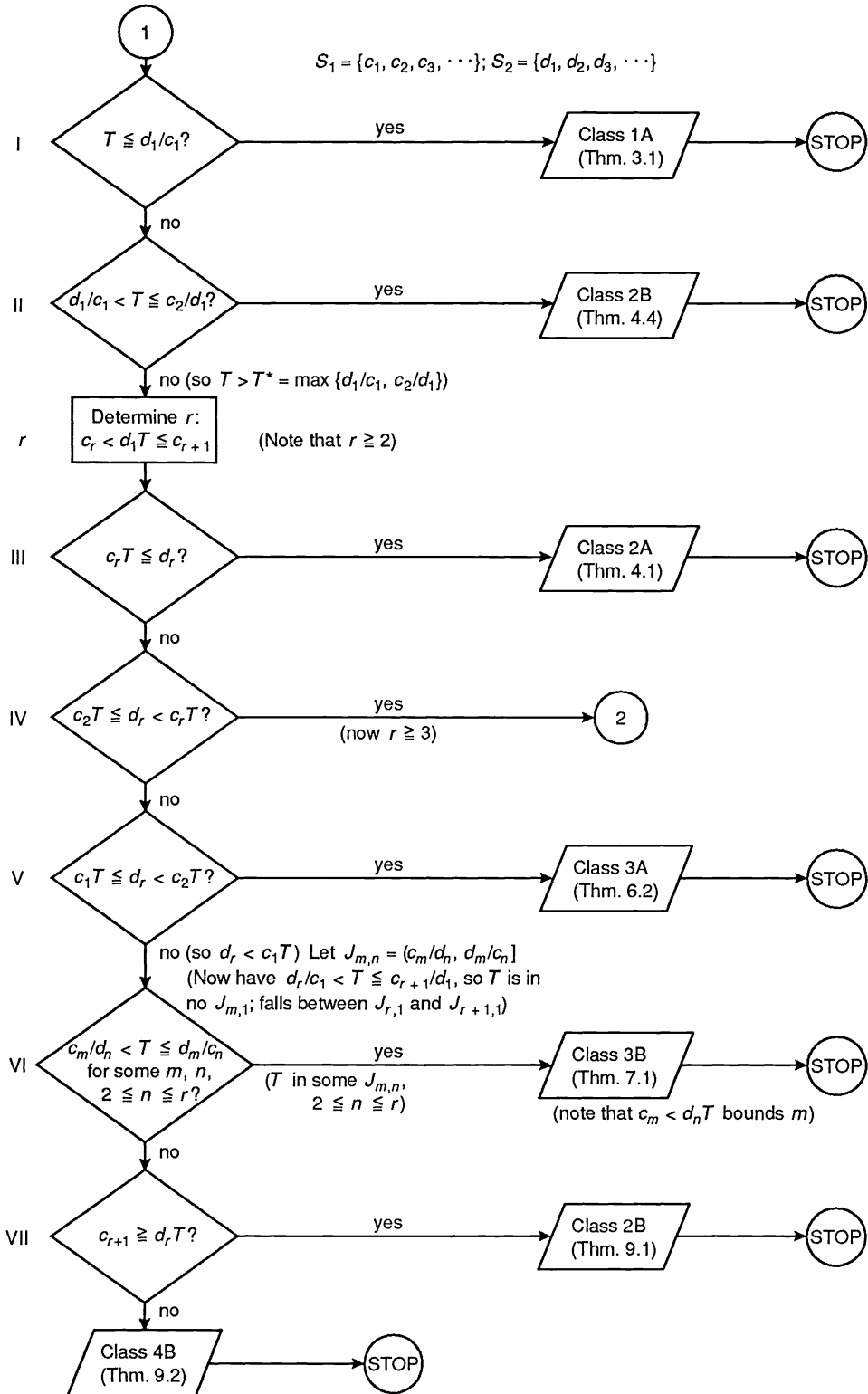


FIG. 7(a)

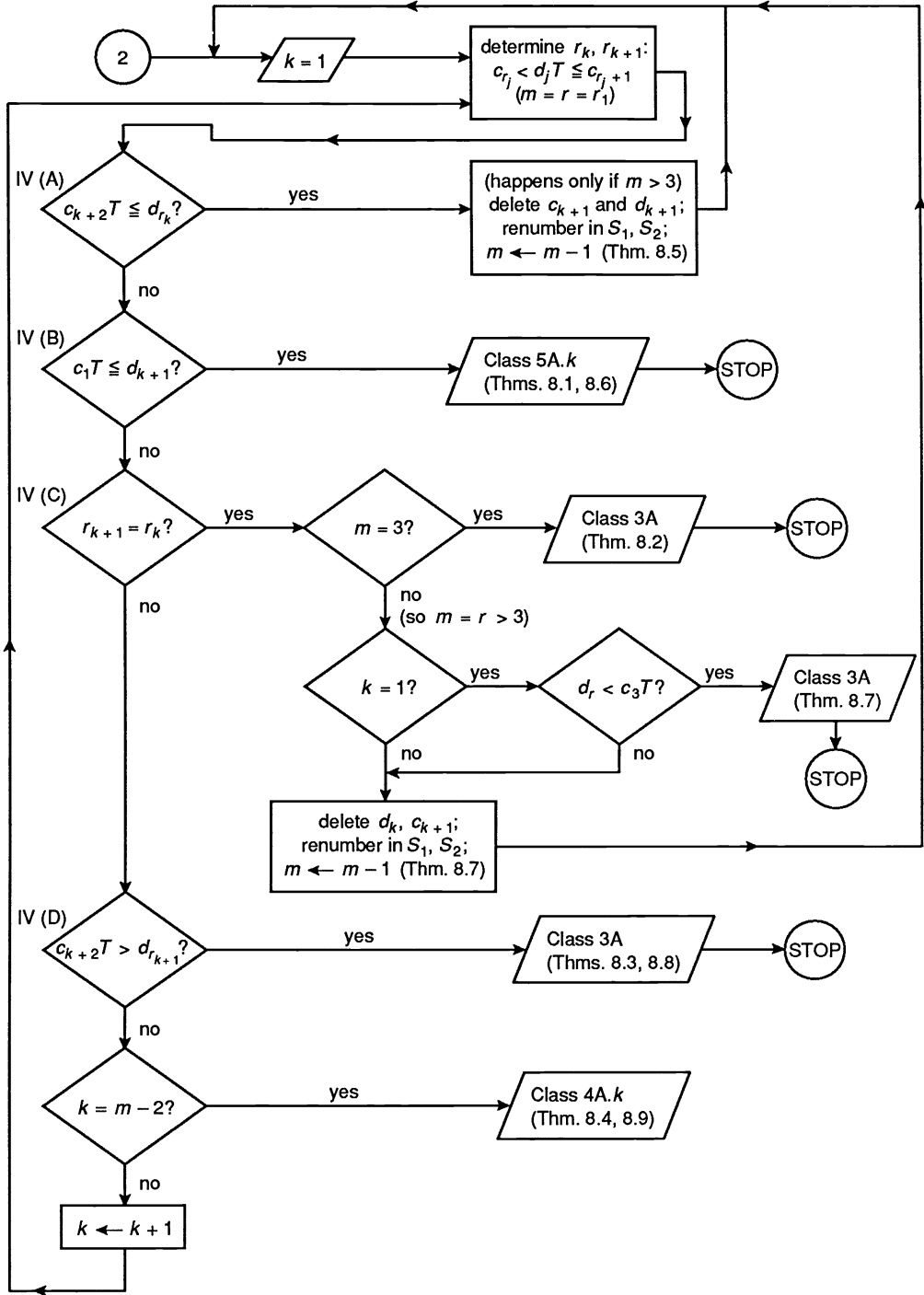


FIG. 7(b)

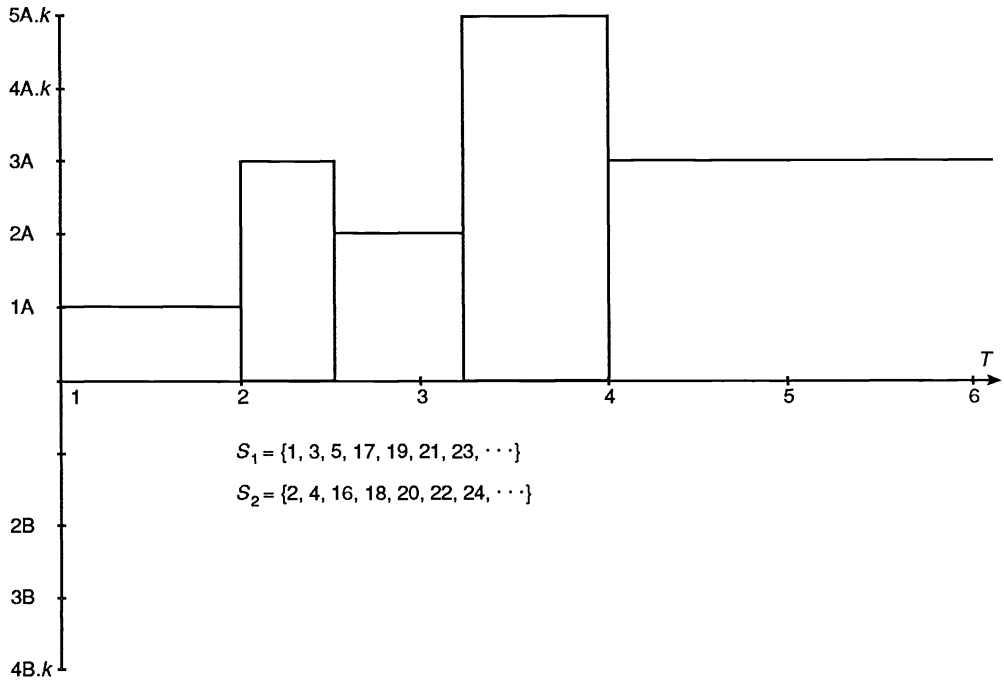


FIG. 8

scheme is presented in flow chart form. Such an algorithm is illustrated in the next example.

Example 3. Let $S_1 = \{1, 3, 5, 17, 19, 21, 23, \dots\}$ and

$$S_2 = \{2, 4, 16, 18, 20, 22, 24, \dots\}.$$

For $n \geq 4$, $c_n = 2n + 9$ and $d_n = 2n + 10$. Since c_n grows only linearly with n , the game is in class 3A for all sufficiently large T (see Theorem 6.7 and Corollary 6.8), so we begin by determining $\bar{T}$ such that we have 3A for all $T > \bar{T}$. In the notation of (6.7.1) we have $T_* = 1.5$ and $c_{n+1}/c_n \leq 1.5$ for all $n \geq 4$. Then by Theorem 6.7, the game is in class 3A for all $T > \bar{T} = \max \{c_4/d_1, d_4/c_2, d_1/c_1\} = 8.5$. This leaves us with only the interval $1 < T \leq 8.5$ to classify, and for this task we begin at box I in the flow chart of Fig. 7(a), where we find we are in class 1A for $1 < T \leq d_1/c_1 = 2$. At box II we find that no class 2B occurs because $c_2/d_1 < d_1/c_1$. At the r box we begin by determining that $r = 2$ for $2 < T \leq 2.5$, and proceed through the rest of the chart with $r = 2$. At boxes III and IV the answers are "no" and at V "yes" for all T in $(2, 2.5]$, so in this interval we have class 3A. Return to the r box and determine that $r = 3$ for $2.5 < T \leq 8.5$. With $r = 3$, the answer at III is "yes" for $T \leq 3.2$ so we have class 2A on $(2.5, 3.2]$. For $T > 3.2$ we continue to IV and find "yes" for $T \leq \frac{16}{3}$, so we enter the chart in Fig. 7(b) with $k = 1$ and $m = r_1 = r = 3$. We find that $r_2 = 3$ for $T \leq 4.25$, and proceed to IV(A), where the answer is no, and then to IV(B), where the answer is yes for $T \leq 4$. Thus for $3.2 < T \leq 4$ we have class 5A.1. For $4 < T \leq 4.25$ we move on to IV(C), where the answer is yes, and $m = 3$ so we have class 3A on this interval. This completes the range for which $r_2 = 3$; for $4.25 < T \leq \frac{16}{3}$, $r_2 = 4$, and we now get negative answers at IV(A), IV(B), IV(C). At IV(D) the answer is yes, so for $4.25 < T \leq \frac{16}{3}$ we

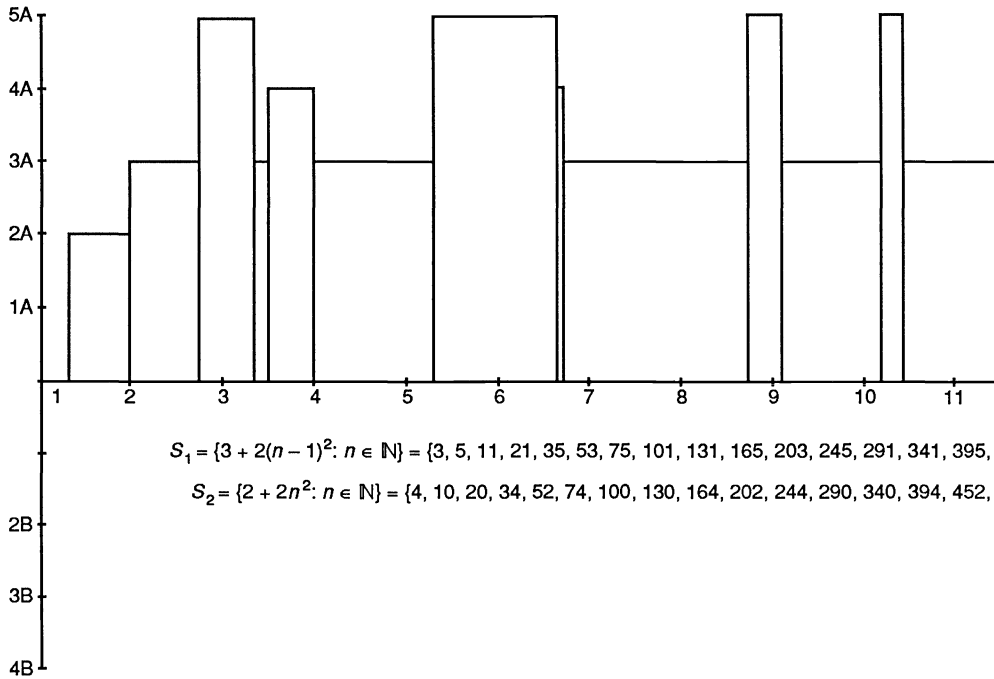


FIG. 9(a). (See Fig. 9(b).)

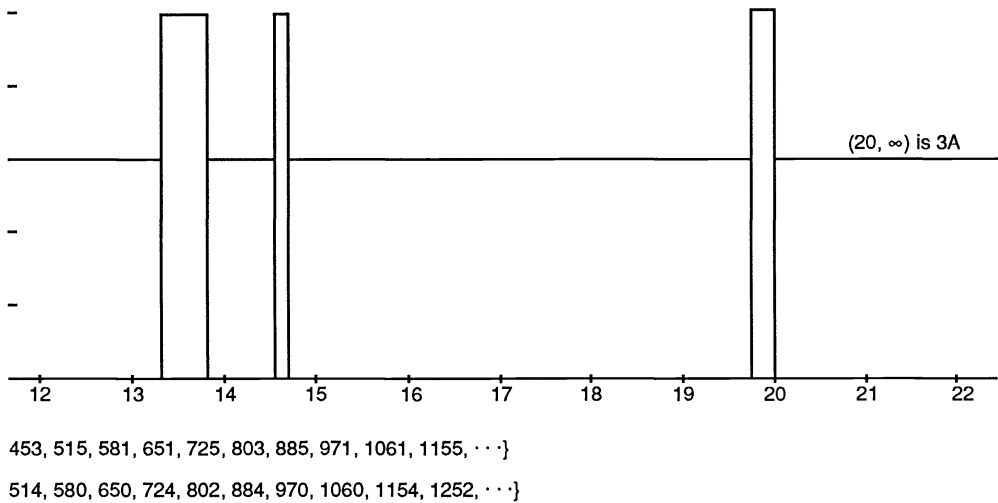


FIG. 9(b). (Fig. 9(a) continued.)

have class 3A. This completes the T range for which the answer at IV in Fig. 7(a) was yes. For $T > \frac{16}{3}$, proceed to box V, with $r = 3$ still. For $\frac{16}{3} < T \leq 8.5$ the answer at V is yes, so we have class 3A.

To summarize, we have 1A on $(1, 2]$, 3A on $(2, 2.5]$, 2A on $(2.5, 3.2]$, 5A.1 on $(3.2, 4]$, and 3A on $(4, \infty)$. The profile is shown in Fig. 8.

Finally, we show the profiles for several further examples in Figs. 9–12.

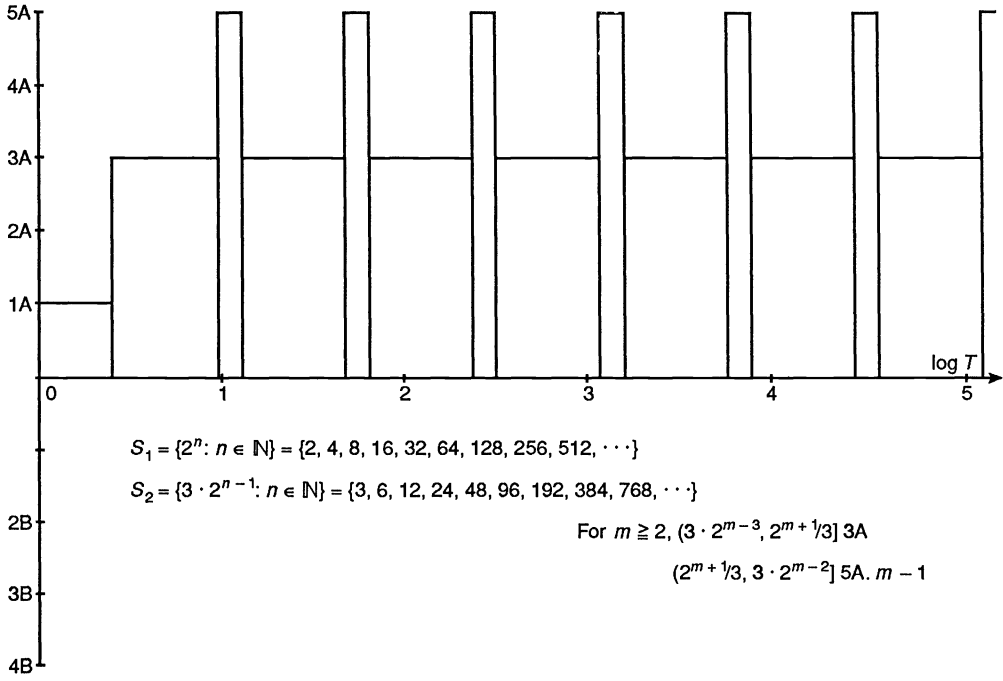


FIG. 10

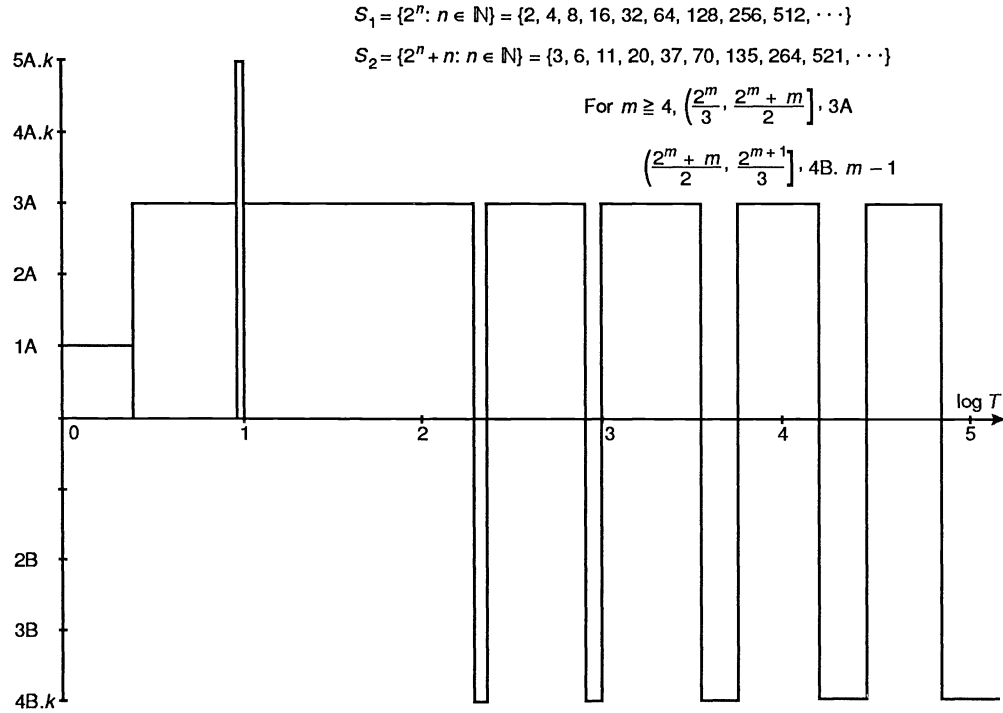


FIG. 11

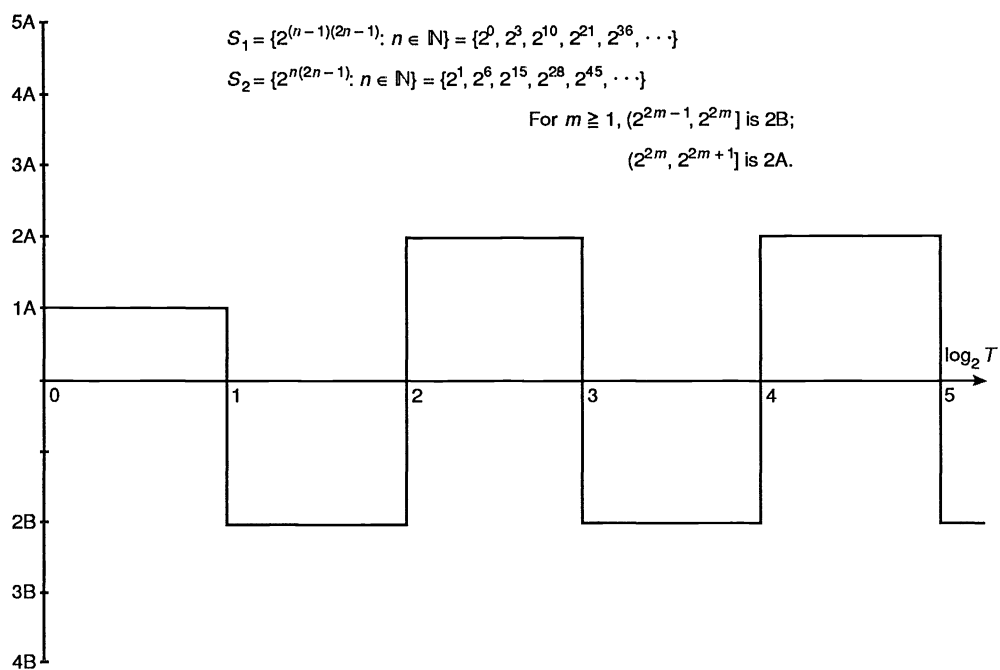


FIG. 12

REFERENCES

- [1] R. J. EVANS, *Silverman's game on discrete sets*, unpublished manuscript.
- [2] ———, *Silverman's game on intervals*, Amer. Math. Monthly, 86 (1979), pp. 277–281.
- [3] G. A. HEUER, *Odds versus evens in Silverman-type games*, Internat. J. Game Theory, 11 (1982), pp. 183–194.
- [4] ———, *A family of games on $[1, \infty)^2$ with payoff a function of y/x* , Naval Research Logistics Quarterly, 31 (1984), pp. 229–249.
- [5] N. S. MENDELSON, *A psychological game*, Amer. Math. Monthly, 53 (1946), pp. 86–88.

THE LINEARITY OF FIRST-FIT COLORING OF INTERVAL GRAPHS*

H. A. KIERSTEAD†

Abstract. It is shown that First-Fit coloring requires at most 40ω colors to color an interval graph with clique size ω . It follows that a polynomial time approximation algorithm for Dynamic Storage Allocation due to Chrobak and Slusarek has a constant performance ratio of 80.

Key words. dynamic storage allocation, First-Fit, interval graph

AMS(MOS) subject classifications. 68Q25, 68R10

1. Introduction. A graph G is said to be an interval graph if its vertex set can be represented by a collection of closed intervals V of the real numbers $\mathcal{R}$ (two intervals are allowed to have the same endpoints) so that for all $i, j \in V$, $i \cap j \neq \emptyset$ if and only if the vertex represented by i is adjacent to the vertex represented by j . In this case we say that V is a representation of G . Associated with any representation V of an interval graph is an interval order $P = (V, <)$, where the relation $i < j$ holds if and only if the right endpoint of i is less than the left endpoint of j . It is well known that interval graphs are perfect and in particular that the chromatic number $\chi(G)$ of an interval graph G is equal to its maximum clique size $\omega(G)$. In this article we study the performance of a particular on-line algorithm, the First-Fit or Greedy algorithm, for coloring interval graphs. A graph coloring algorithm is said to be on-line if it runs in the following manner. The vertices of the graph G to be colored are examined in some externally determined sequence. At the time a vertex v is examined its adjacency relations with all previously examined vertices are known as well as the colors assigned to those vertices, but nothing is known about unexamined vertices. Before examining any more vertices the algorithm irrevocably assigns a color to v . In the algorithm First-Fit this assignment is made by coloring the current vertex v the color α , where α is the least positive integer not already assigned to a neighbor of v . If the vertices of G are considered in an ideal sequence then First-Fit will use only $\chi(G)$ colors. (To construct such a sequence first find an optimal coloring of G and then put all vertices with the same color in consecutive positions in the sequence.) The following theorem establishes a linear upper bound in terms of ω for the number of colors used by First-Fit to color an interval graph when the vertices are considered in the worst possible sequence. This confirms a conjecture of Woodal [13] from 1974, which was rediscovered by Chrobak and Slusarek [2], [3] in 1984. The present formulation of the result is due to Gyarfás and Lehel [4], [5].

THEOREM 1. *Let G be an interval graph. The on-line algorithm First-Fit requires less than $40\omega(G)$ colors to color G .*

Previously, Witsenhausen [14], and independently Gyarfás and Lehel [4], had obtained an upper bound of the form $c(\epsilon)\omega(G)^{1+\epsilon}$, for all $\epsilon > 0$. The work of Gyarfás and Lehel in [4] just improved the Gyarfás–Lehel proof to obtain the upper bound $c\omega(G) \lg(\omega(G))$. Gyarfás and Lehel [5] have recently proved an upper bound of $24\omega(G)$ for a special case of this problem. Woodall [12] showed that $(4/(\sqrt{17} - 3))\omega(G)$ is a lower bound. Witsenhausen [14] and, independently, Chrobak and Slusarek [2], improved the constant to four. Chrobak and Slusarek [2] also claim, without proof, that the constant must be at least 4.4.

* Received by the editors March 1, 1988; accepted for publication March 23, 1988. This work was partially supported by Office of Naval Research contract N00014-85K-0494.

† Department of Mathematics, University of South Carolina, Columbia, South Carolina 29208.

In related work, in Kierstead and Trotter [9] the author presented an on-line algorithm for coloring an interval graph G with at most $3\omega(G) - 2$ colors and showed that no on-line algorithm could do better. Their algorithm was almost, but not quite, the First-Fit algorithm. Because of this similarity, the author [8] constructed an example to show that First-Fit does not achieve this level of performance. Bean [1] has shown that for any on-line algorithm there exist bipartite (noninterval) graphs requiring arbitrarily large numbers of colors. Recently Lovasz, Saks, and Trotter [10] presented on-line algorithms for all k , which color any graph G of chromatic number k with at most $o(n)$ colors, where n is the number of vertices of G . In another direction, the author [7] found an on-line algorithm to cover an ordered set of width w with $(5^w - 1)/4$ chains.

Our interest in First-Fit arises from the work of Chrobak and Slusarek [3] on the following problem [6].

DYNAMIC STORAGE ALLOCATION (DSA).

INSTANCE: Set A of items to be stored, each $a \in A$ having a size $s(a) \in \mathbb{Z}^+$, an arrival time $r(a) \in \mathbb{Z}_0^+$, and a departure time $d(a) \in \mathbb{Z}^+$, and a positive storage size D .

QUESTION: Is there a feasible allocation of storage for A , i.e., a function $\sigma: A \rightarrow \{1, 2, \dots, D\}$ such that for every $a \in A$ the allocated storage interval $l(a) = [\sigma(a), \sigma(a) + 1, \dots, \sigma(a) + s(a) - 1]$ is contained in $[1, D]$ and such that, for all $a, a' \in A$, if $l(a) \cap l(a')$ is nonempty then either $d(a) \leq r(a')$ or $d(a') \leq r(a)$?

An item a can be interpreted as an array in a computer program, where $s(a)$ is its dimension and the interval $[r(a), d(a)]$ is the period for which the array is active. Then the question asks whether the arrays a can be assigned subblocks of memory of size $s(a)$ from a block of memory of size D so that no two arrays active at the same time are assigned overlapping subblocks. When all the sizes are the same this problem is equivalent to coloring a representation of an interval graph and can be solved off-line in polynomial time by applying First-Fit to the intervals ordered by their left endpoints. In the general case, Stockmeyer [11] has shown that DSA is NP-complete in the strong sense. Thus it makes sense to look for a polynomial time approximation algorithm for DSA. Chrobak and Slusarek [2] proposed the following algorithm.

BUDDY DECREASING SIZE ALGORITHM (BDS).

- (1) Increase the size of each object to a power of two.
- (2) Order the objects by decreasing size and then replace each object a in the order by $s(a)$ copies of $[r(a), d(a)]$. Let G be the interval graph formed by these intervals.
- (3) Apply the First-Fit coloring algorithm to the intervals of G in the order from step (2) to produce the function σ .

Chrobak and Slusarek showed that the function σ produced by BDS satisfies the requirements of DSA if D is the number of colors used in step (3). Steps (1) and (2) assure that there is always enough space so that no subblocks are forced to be broken up by First-Fit in step (3) and using First-Fit assures that no subblocks are voluntarily broken up in step (3). It follows from Theorem 1 that BDS has a constant performance ratio of 80.

Let $G = (V, E)$ be an interval graph with representation V and let $P = (V, <)$ be the corresponding interval order. For an interval $i \in V$ and a subset L of V we write $i < L$ (respectively, $L < i$) if, for all $l \in L$, $i < l$ (respectively, $l < i$). We denote the maximum clique size in G by $\omega(G)$. Let L be a subset of V . The *neighborhood* of an interval $i \in V$

with respect to L is denoted by $N(i/L) = \{j \in L: i \sim j\}$. The *density* of a point $r \in \mathcal{R}$ with respect to L , denoted by $D(r/L)$, is $D(r/L) = |\{i \in L: r \in i\}|$. Thus, for example, $\omega(G) = \max \{D(r/V): r \in \mathcal{R}\}$. The *density* of an interval $i \in V$ with respect to L , denoted by $D(i/L)$, is $D(i/L) = \min \{D(r/L): r \in i\}$. For an interval $i \in V$, the *height* of i with respect to L , denoted by $h(i/L)$, is the length of the longest chain C in L such that $C < i$, the *depth* of i with respect to L , denoted by $d(i/L)$, is the length of the longest chain C in L such that $C > i$, and the *centrality* of i with respect to L , denoted by $c(i/L)$, is the minimum of $h(i/L)$ and $d(i/L)$. Notice that for $N(j) \supset L$, if i contains an endpoint of j then $c(i/L) = 0$ and if $c(i/L) \geq 1$, then $j \supset i$. When $L = V$ we may drop the $/L$ in the above notation. Finally, if $N = (n_1, n_2, \dots, n_t)$ is a sequence of length t we denote the initial subsequence of N of length k by $N_k = (n_1, n_2, \dots, n_k)$. The result $(n_1, n_2, \dots, n_k, n)$ of adjoining n to the end of N is denoted by $N^\wedge n$.

2. The proof. The following technical lemma is needed for the proof of Theorem 1. It was first shown to the author by Gyarfás, and appears in Witsenhausen [14] and in Gyarfás and Lehel [4], [5].

LEMMA 2. *Let V be a representation of an interval graph. If L is a clique in V , then there exists an interval $i \in L$ such that $D(i/L) \geq |L|/2$.*

Proof. We argue by induction on $|L|$. The cases $|L| = 1$ and $|L| = 2$ are trivial. For the induction step, let j be the interval of L with the smallest left endpoint and let k be the interval of L with the largest right endpoint. Using the induction hypothesis, we choose an interval $i \in L - \{j, k\}$ such that $D(i/L - \{j, k\}) \geq |L|/2 - 1$. Since $j \sim k, j \cup k \supset i$, and thus $D(i/L) \geq |L|/2$. $\square$

Proof of Theorem 1. Suppose $G = (V, E)$ is an interval graph, with interval representation V , on which First-Fit has used $x \geq 40y$ colors. Let $f(i)$ be the color assigned to interval i by First-Fit. Let $C(q) = \{i \in V: f(i) = q\}$ and $C[p, q] = \bigcup_{p \leq r \leq q} C(r)$. We shall use a recursive procedure to find an interval with density $y + 1$, thus showing that $x < 40\omega(G)$.

Roughly the idea is as follows. At a given stage there will be intervals $i_1 \supset i_2 \supset \dots \supset i_t$ and disjoint blocks $B_1, \dots, B_t$ such that $C[x - 40r_m + 1, x - 40r_{m-1}] \cap N(i_{m-1}) \supset B_m$ and $D(i_1/B_1) + \dots + D(i_t/B_t) \geq s_m$, where $r_m < s_m$. Since the intervals form a decreasing sequence, $D(i_t/C[x - 40r_m + 1, x]) \geq s_m$. We would like to continue this focusing process by finding $i_{t+1} \subset i_t$ with high density in B_{t+1} . However, there may not be any intervals with high density in B_{t+1} or the only intervals with high density in B_{t+1} may overlap i_t . In the first case we will be able to choose i_{t+1} with high centrality (cases (i) and (ii) below). In the second case we will be able to choose i with extra large density and will try to replace B_t with $B' = B_t \cup B_{t+1}$ and i_t with i (case (iv) below). In this case the centrality of i_t may decrease by one. To insure that we can choose $i \in B'$ with extra high density and contained in i_{t-1} we will require the centrality of i_m in B_m to be at least two, if $m > 1$. If $c(i_m/B_m) = 2$ and we are in the second case, we may have to replace i_{m-1} instead of i_m by i to maintain this requirement. This requirement introduces a further complication. The interval i with high density in B_{t+1} may have centrality one in B_{t+1} . In this case we replace i_t with i (case (iii) below).

The procedure discussed above requires a rather complicated induction hypothesis, which we now develop. Let $N = (n_1, n_2, \dots, n_t)$ be a sequence of positive integers and let $I = (i_1, i_2, \dots, i_t)$ be a sequence of intervals of V . The following parameters are defined on the pair (N, I) .

- (a) $s_0 = 0$, $s_{m+1} = s_m + n_{m+1}$, and $r_m = s_{m-1} + n_m/2$;
- (b) $B_m = C[x - 40r_m + 1, x - 40r_{m-1}] \cap N(i_{m-1})$, for $m \geq 1$, where we let $i_0 = \mathcal{R}$;
- (c) $c_m = c(i_m/B_m)$ and $d_m = D(i_m/B_m)$.

The pair (N, I) is *admissible* if the following conditions hold for $1 \leq m \leq t$:

- (1) $i_m \in B_m$;
- (2) $c_m \geq 2$, if $m > 1$;
- (3) $n_{m+1} \geq n_m 3^{2-c_{m+1}}$, if $m < t$;
- (4) $d_m \geq n_m$.

Notice that if (N, I) is an admissible pair then $D(i_t) \geq s_t$, since the B_m are disjoint, the i_m form a decreasing sequence by (2), and $d_m \geq n_m$ by (4). We shall prove the theorem by showing that there exists an admissible pair (N, I) , and if (N, I) is an admissible pair and $40s_t \leq x$ then there exists an admissible pair (N', I') such that $s_t < s'_{t'}$. (Here and below primes indicate elements and parameters of the new admissible pair.)

First notice that (N, I) is an admissible pair if $N = (1)$ and $I = (i)$, where i is any interval colored x . Now fix an admissible sequence (N, I) . We now begin the process of constructing (N', I') . Let $B = C[x - 40s_t + 1, x - 40r_t] \cap N(i_t)$. Since $i_t \in B_t$, $f(i_t) > x - 40r_t$, and according to First-Fit, there exists an interval $j \in B$ such that $f(j) = \alpha$ for each $\alpha \in [x - 40s_t + 1, x - 40r_t]$. Thus B contains at least $20n_t$ intervals. Let $L_k = \{i \in B: h(i/B) = k \leq d(i/B)\}$ and $R_k = \{i \in B: d(i/B) = k < h(i/B)\}$. Then $i \in L_k \cup R_k$ or $i \in B - \bigcup_{j < k} (L_j \cup R_j)$ implies that $c(i/B) \geq k$. By the pigeon hole principle one of the following cases holds:

- (i) There exists $i \in B$ such that $c(i/B) \geq 2 + \lceil \log_3(n_t) \rceil$;
- (ii) $|L_k| \geq 2n_t 3^{2-k}$ or $|R_k| \geq 2n_t 3^{2-k}$, for some k such that $2 \leq k \leq 1 + \lceil \log_3(n_t) \rceil$;
- (iii) $|L_1| \geq 2n_t$ or $|R_1| \geq 2n_t$; or
- (iv) $|L_0| \geq 5n_t$ or $|R_0| \geq 5n_t$.

Case (i). Set $t' = t + 1$, $i'_{t+1} = i$, $n'_{t+1} = 1$, $N' = N \wedge n'_{t+1}$, and $I' = I \wedge i'_{t+1}$. It is apparent that (N', I') is an admissible pair and $s'_{t'} = s_t + 1$.

Case (ii). Without loss of generality assume $|L_k| \geq 2n_t 3^{2-k}$. Note that L_k is an antichain in P since $i, j \in B$ and $i < j$ in P implies $h(i/B) < h(j/B)$. Thus L_k is a clique in G and, by Lemma 2, there exists $i \in L_k$ such that $D(i/B) \geq n_t 3^{2-k}$. Set $t' = t + 1$, $i'_{t+1} = i$, $n'_{t+1} = n_t 3^{2-k}$, $N' = N \wedge n'_{t+1}$, and $I' = I \wedge i'_{t+1}$. Again it is apparent that (N', I') is an admissible pair and $s'_{t'} = s_t + n_t 3^{2-k}$.

Case (iii). Without loss of generality assume $|L_1| \geq 2n_t$. As before $|L_1|$ is independent and there exists $i \in L_1$ such that $D(i/B) \geq n_t$. Set $t' = t$, $n'_t = 2n_t$, $i'_t = i$, $N' = N \wedge_{t-1} n'_t$, and $I' = I \wedge_{t-1} i'_t$. Then $B'_t \supset B_t \cup B$, since $r'_t = s_{t-1} + n'_t/2 = s_t$. Also $i_t \supset i'_t$. Thus $i'_t \in B'_t$, $c'_t \geq c_t$, $n'_t > n_t \geq n_{t-1} 3^{2-c_t} \geq n'_{t-1} 3^{2-c'_t}$, and $d'_t \geq D(i/B) + d_t \geq 2n_t = n'_t$. Thus the pair (N', I') is admissible. Also $s'_{t'} = s_t + n_t$.

Case (iv). Without loss of generality assume $|L_0| \geq 5n_t$. Let e be the left endpoint of i_t and $T = B_t \cup B$. Then $D(e/T) \geq D(i_t/B_t) + |L_0| \geq 6n_t$. Thus by Lemma 2 there exists an interval $j_1 \in T$ such that $D(j_1/T) \geq 3n_t$. Choose an interval $j_2 \in B_t$ such that $c(j_2/B_t) = c_t - 1$ and $j_2 < i_t$. Then either $c(j_1/T) \geq c_t - 1$ or $j_1 \supset j_2$ and thus $D(j_2/T) \geq 3n_t$. In either case there exists $i \in T$ such that $D(i/T) \geq 3n_t$ and $c(i/T) \geq c_t - 1$.

If $c(i/T) \geq 2$ or $t = 1$, set $t' = t$, $n'_t = 3n_t$, $i'_t = i$, $N' = N \wedge_{t-1} n'_t$, and $I' = I \wedge_{t-1} i'_t$. Then $B'_t \supset T$, since $r'_t = s_{t-1} + n'_t/2 > s_t$. If $t > 1$, then $i_{t-1} \supset i'_t$. Thus $i'_t \in B'_t$, $c'_t \geq c_t - 1$, $n'_t = 3n_t \geq 3n_{t-1} 3^{2-c_t} \geq n'_{t-1} 3^{2-c'_t}$, and $d'_t \geq 3n_t = n'_t$. Thus the pair (N', I') is admissible. Also $s'_{t'} = s_t + 2n_t$.

Otherwise $c(i/T) = 1$ and $t > 1$. Thus $c_t = 2$ and $n_t \geq n_{t-1}$. Set $t' = t - 1$, $n'_{t-1} = 3n_t + n_{t-1}$, $i'_{t-1} = i$, $N' = N \wedge_{t-2} n'_{t-1}$, and $I' = I \wedge_{t-2} i'_{t-1}$. Then $B'_{t-1} \supset B_{t-1} \cup T$, since $r'_{t-1} = s_{t-2} + n'_{t-1}/2 \geq s_{t-2} + n_{t-1} + n_t = s_t$. Also $i_{t-1} \supset i'_{t-1}$. Thus $i'_{t-1} \in B'_{t-1}$, $c'_{t-1} \geq c_{t-1}$, $n'_{t-1} \geq n_{t-1} \geq n_{t-2} 3^{2-c_{t-1}} \geq n'_{t-2} 3^{2-c'_{t-1}}$, and $d'_{t-1} \geq D(i/T) + d_{t-1} \geq n'_{t-1}$. Thus the pair (N', I') is admissible. Also $s'_{t'} = s_t + 2n_t$. $\square$

The constant $c = 40$ we used to establish the linearity of First-Fit for coloring interval orders is not optimal. For example in the above argument we only used the number of colors appearing on intervals in B to estimate the size of B . However if B really only has one interval of each color then B has no chain of length four and we can choose i_t with considerably higher density. Using this idea it is possible to reduce c to 32 at the cost of additional complexity. Other tricks can be used to further reduce c , but at this time we have no ideas as to how to show that $c \approx 5$. It is still of interest to know the optimal value of c .

Acknowledgments. The author thanks W. T. Trotter and A. Gyárfas for stimulating discussions on this topic.

REFERENCES

- [1] D. BEAN, *Effective coloration*, J. Symbolic Logic, 41 (1976), pp. 469–480.
- [2] M. CHROBAK AND M. SLUSAREK, *On some packing problems related to dynamic storage allocation*, preprint, 1984.
- [3] ———, Problem 84-23, J. of Algorithms, 5 (1984), p. 588.
- [4] A. GYÁRFAS AND J. LEHEL, *On-line and First-fit colorings of graphs*, J. Graph Theory, 12 (1988), pp. 217–227.
- [5] ———, *On a special case of the wall problem*, preprint, 1988.
- [6] M. GAREY AND D. JOHNSON, *Computers and Intractability, a Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, 1979.
- [7] H. A. KIERSTEAD, *An effective version of Dilworth's theorem*, Trans. Amer. Math. Soc., 268 (1981), pp. 63–77.
- [8] ———, unpublished, 1981.
- [9] H. A. KIERSTEAD AND W. T. TROTTER, *An extremal problem in recursive combinatorics*, Congress. Numer., 33 (1981), pp. 143–153.
- [10] L. LOVASZ, M. SAKS, AND W. T. TROTTER, preprint, 1988.
- [11] I. J. STOCKMEYER, private communication, 1976.
- [12] D. R. WOODALL, private communication, 1973.
- [13] ———, Problem No. 4, *Combinatorics* (Proc. British Combinatorial Conference 1973), London Mathematical Society Lecture Note Series 13, T. P. McDonough and V. C. Marvon, eds., Cambridge University Press, London, 1974, p. 202.
- [14] H. S. WITSENHAUSEN, *On Wodall's interval problem*, J. Combin. Theory Ser. A, 21 (1976), pp. 222–229.

BROADCAST NETWORKS OF BOUNDED DEGREE*

ARTHUR L. LIESTMAN† AND JOSEPH G. PETERS†

Abstract. Broadcasting is an information dissemination process in which a message is to be sent from a single originator to all members of a network by placing calls over the communication lines of the network. Several previous papers have investigated ways to construct sparse graphs (networks) in which this process can be completed in minimum time from any originator. The graphs produced by these methods contain high degree vertices. This paper describes graphs with fixed maximum degree in which broadcasting can be completed in near minimum time.

Key words. broadcasting, graphs, networks, bounded degree, shuffle-exchange networks, cube-connected cycles

AMS(MOS) subject classifications. 94C15, 68M10

1. Definitions. Broadcasting refers to the process of message dissemination in a communication network whereby a message, originated by one member, is transmitted to all members of the network. Broadcasting is accomplished by placing a series of calls over the communication lines of the network. This is to be completed as quickly as possible subject to the constraints that each call involves only two vertices, each call requires one unit of time, a vertex can participate in only one call per unit of time, and a vertex can only call a vertex to which it is adjacent.

Given a connected graph G and a message originator, vertex u , we define the *broadcast time of vertex u* , $b(u)$, to be the minimum number of time units required to complete broadcasting from vertex u . It is easy to see that for any vertex u in a connected graph G with n vertices, $b(u) \geq \lceil \log_2 n \rceil$, since during each time unit the number of informed vertices can at most double.

We define the *broadcast time of a graph G* , $b(G)$, to be the maximum broadcast time of any vertex u in G , i.e., $b(G) = \max \{b(u) \mid u \in V(G)\}$. For the complete graph K_n with $n \geq 2$ vertices, $b(K_n) = \lceil \log_2 n \rceil$, yet K_n is not minimal with respect to this property. That is, we can remove edges from K_n and still have a graph G with n vertices such that $b(G) = \lceil \log_2 n \rceil$. We define a *minimal broadcast graph* to be a graph G with n vertices such that $b(G) = \lceil \log_2 n \rceil$, but for every proper spanning subgraph $G' \subset G$, $b(G') > \lceil \log_2 n \rceil$.

We define the *broadcast function $B(n)$* to be the minimum number of edges in any minimal broadcast graph on n vertices. A *minimum broadcast graph (mbg)* is a minimal broadcast graph on n vertices having $B(n)$ edges. From an applications perspective, minimum broadcast graphs represent the cheapest possible communication networks (having the fewest communication lines) in which broadcasting can be accomplished, from any vertex, as fast as theoretically possible.

2. Previous results. Johnson and Garey (see [6]) showed that the problem of determining whether $b(v) \geq k$, for a vertex v in an arbitrary graph G for fixed $k \geq 4$ is NP-complete.

In [5], Farley, Hedetniemi, Mitchell, and Proskurowski studied the broadcast function $B(n)$, i.e., the number of edges in a minimum broadcast graph with n vertices. In

* Received by the editors October 5, 1987; accepted for publication (in revised form) May 18, 1988. This research was supported by the Natural Sciences and Engineering Research Council of Canada under grants A-1734 and A-0322.

† School of Computing Science, Simon Fraser University, Burnaby, British Columbia, Canada V5A 1S6.

TABLE 2.1
The broadcast function $B(n)$.

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$B(n)$	0	1	2	4	5	6	8	12	10	12	13	15	18	21	24	32	22	23

particular, they determined the values of $B(n)$ for $n \leq 15$ and showed that $B(2^k) = k \times 2^{k-1}$. Mitchell and Hedetniemi [9] determined the value for $B(17)$ and Wang [14] found the value of $B(18)$. For $1 \leq n \leq 18$, the values of the broadcast function $B(n)$ are shown in Table 2.1.

Farley, Hedetniemi, Mitchell, and Proskurowski [5] also found examples of minimum broadcast graphs for each value of n , $1 \leq n \leq 15$. This work was extended by Mitchell and Hedetniemi [9] who found all mbgs with $n \leq 11$ vertices and presented the only known examples of mbgs with $12 \leq n \leq 17$ vertices. More recently, Bermond [2] found a second mbg with 14 vertices and Wang [14] found three additional mbgs on 17 vertices and the only known mbg on 18 vertices. For $1 \leq n \leq 18$, the number of mbgs with n vertices is shown in Table 2.2 where * indicates that the value given is a lower bound on the number of mbgs.

The results of these studies suggest that mbgs are extremely difficult to find; in fact, no mbg with n vertices is known for any value of $n \geq 19$, except for the values $n = 2^k$, where mbgs are easy to construct.

Since mbgs seem to be difficult to find for arbitrary n , several authors have devised methods to construct sparse graphs that allow minimum time broadcasting from each vertex. We call such a graph a *minimal broadcast network*.

In [4], Farley designed several techniques for constructing minimal broadcast networks with n vertices and approximately $(n/2) \log_2 n$ edges, for arbitrary values of n . Chau and Liestman [3] presented constructions based on Farley's techniques that yield somewhat sparser minimal broadcast networks for most values of n . In [10], Peleg showed that $B(n) \in \Theta(L(n-1)n)$ for $n \geq 1$ where $L(k)$ denotes the exact number of consecutive leading 1's in the binary representation of k . Asymptotically, Peleg's construction (which establishes an upper bound) produces the best known minimal broadcast networks for most values of n .

So far, the emphasis in this research has been on obtaining sparse graphs in which each vertex can broadcast in minimum time. (Questions related to broadcasting in slightly more than minimum time have previously been addressed by Liestman [8] and by Peleg [10].) If these graphs are to be used in the design of actual networks, other considerations may override the need for minimum time broadcasting. In particular, the constructions of Farley and of Chau and Liestman result in networks with average degree $O(\log_2 n)$ while the construction of Peleg yields networks with some vertices of degree $O(n)$. It may be more realistic to use a graph with fixed maximum degree in which every vertex can broadcast "quickly." We will use the term *broadcast network* to describe such a graph.

TABLE 2.2
Number of mbg's.

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Number	1	1	1	1	4	1	1	4	6	21	2	1*	1*	2*	3*	8*	5*	1*

* Indicates number known.

It is easily seen that a broadcast can be completed in $O(\sqrt{n})$ time for any vertex in an n vertex chordal ring [1] that has $O(n)$ edges and maximum degree 3. In this paper we describe how to construct broadcast networks on n vertices with $O(n)$ edges, maximum degree k , and broadcast time $O(\log_2 n)$ for $k = 3, 4$.

3. Lower bounds. As noted above, $\lceil \log_2 n \rceil$ is a lower bound on the broadcast time of any graph on n vertices. This bound can be improved for graphs with bounded degree.

THEOREM 1. *If G is a graph on n vertices with maximum degree 3, then $b(G) \geq 1.4404 \log_2 n - 1.769$.*

Proof. Let a_k be the maximum number of nodes that could be informed after k time units in any graph with maximum degree 3. If each vertex informs its neighbors as quickly as possible after it has received the message, the only vertices that can have uninformed neighbors at time k are the vertices that first became informed at times $k - 1$ and $k - 2$. This gives the recurrence relation $a_0 = 1$, $a_1 = 2$, $a_2 = 4$, $a_3 = 8$, and $a_t = a_{t-1} + (a_{t-1} - a_{t-3}) = 2a_{t-1} - a_{t-3}$ for $t > 3$. Using generating functions and some identities from [12] we obtain $a_0 = 1$ and $a_t = F_{t+2} - 2$ for $t \geq 1$, where F_k is the k th Fibonacci number. We need to solve for t to determine a lower bound on the time required to inform $n = a_t$ vertices. If we let $\phi = (1 + \sqrt{5})/2$ and $\hat{\phi} = (1 - \sqrt{5})/2$, we obtain $n \leq 2/\sqrt{5}(\phi^{t+2} - \hat{\phi}^{t+2}) - 2$. Since $|\hat{\phi}| < 1$, the expression can be simplified to $n \leq (2/\sqrt{5})\phi^{t+2}$. Solving for t , we obtain $t \geq 1.4404 \log_2 n - 2.23$. Using iterated numerical techniques, we obtain a slightly more accurate additive constant of -1.769 . $\square$

THEOREM 2. *If G is a graph on n vertices with maximum degree 4, then $b(G) \geq 1.1374 \log_2 n - 0.637$.*

Proof. Using arguments similar to those in the proof of the previous theorem, we obtain the recurrence relation for graphs with maximum degree 4: $a_0 = 1$, $a_1 = 2$, $a_2 = 4$, $a_3 = 8$, $a_4 = 16$, and $a_t = 2a_{t-1} - a_{t-4}$ for $t > 4$. This recurrence relation is somewhat more difficult to solve analytically than the recurrence for degree 3 graphs, but the coefficients are easily estimated by iterated numerical techniques to get the above result. $\square$

It is clear that similar bounds can be obtained for other values of k .

4. Cube-connected cycles. The first broadcast networks we describe are based on the *cube-connected cycles* of Preparata and Vuillemin [11]. The cube-connected cycles on $n = k2^k$ vertices is a cubic graph (degree 3 at every vertex) with diameter $5/2 \log_2 n + O(1)$ that can be used to emulate the k -dimensional cube. Before considering the cube-connected cycles, we give a broadcasting scheme for the k -dimensional cube that can then be generalized to the cube-connected cycles on $n = k2^k$ vertices.

A cube on 2^k vertices can be described as follows. Label each vertex with a unique k bit string $x = x_{k-1}x_{k-2} \cdots x_1x_0$. Each vertex x is adjacent to vertices $x^0 = x_{k-1} \cdots x_2x_1\bar{x}_0$, $x^1 = x_{k-1} \cdots x_2\bar{x}_1x_0$, $\dots$, and $x^{k-1} = \bar{x}_{k-1}x_{k-2} \cdots x_1x_0$.

To broadcast on the cube, we say that the originator learns the message at time 0. Each vertex x that learns the message at time $i \leq k - 1$ calls its neighbors $x^i, x^{i+1}, \dots$, and x^{k-1} , informing them of the message, at times $i + 1, i + 2, \dots$, and k , respectively. When we use this scheme, all vertices are informed by time k , which is optimal.

The cube-connected cycles on $n = k2^k$ vertices can be formed by replacing every vertex of the k -dimensional cube with a cycle of length k and distributing the edges of the cube around the cycle so that each vertex in the resulting graph has degree 3. In particular, we label each of the k vertices in the cycle, replacing the vertex labeled x with a pair (x, i) where i is an integer in the range $0 \leq i \leq k - 1$. Vertex (x, i) is adjacent to vertices $(x, (i + 1) \bmod k)$, $(x, (i - 1) \bmod k)$ and (y, i) where $y = x^i = x_{k-1} \cdots x_{i+1}\bar{x}_ix_{i-1} \cdots x_0$.

To broadcast on the cube-connected cycles, we generalize the scheme described for the cube by replacing each call in the cube scheme by three calls in the cube-connected cycles scheme. This will ensure that at least one vertex in each cycle has been informed. Additional calls are then used to finish the scheme.

Without loss of generality, assume that the originator is labeled $(x, 0)$. At time 0, the originator learns the message. For every vertex (x, i) that learns the message at time $2i$ with $i \leq k-1$, (x, i) calls (y, i) at time $2i+1$, and at time $2i+2$, (x, i) calls $(x, (i+1) \bmod k)$, while (y, i) calls $(y, (i+1) \bmod k)$. Note that using these calls, at least one vertex of each cycle has learned the message by time $2k-1$ (the calls at time $2k$ are not necessary for this condition). More specifically, at time $2k-1$ each cycle contains informed members (x, i) , $(x, i+1)$, $\dots$, and $(x, k-1)$ for some $i \leq k-1$, leaving $(x, 0)$, $(x, 1)$, $\dots$, and $(x, i-1)$ uninformed. These vertices are informed by the following calls: At time $2k$, $(x, k-1)$ calls $(x, 0)$. At time $2k+l$ where $1 \leq l \leq \lfloor i/2 \rfloor - 1$, $(x, l-1)$ calls (x, l) while $(x, i-l+1)$ calls $(x, i-l)$. Finally, at time $2k + \lfloor i/2 \rfloor$, $(x, \lfloor i/2 \rfloor - 1)$ calls $(x, \lfloor i/2 \rfloor)$ and if i is odd $(x, \lceil i/2 \rceil + 1)$ calls $(x, \lceil i/2 \rceil)$. Since each cycle has at most $k-1$ uninformed members at time $2k-1$, every vertex is informed by time $2k + \lceil (k-2)/2 \rceil = \lceil 5k/2 \rceil - 1$. Thus, we have the following theorem.

THEOREM 3. *If G is the cube-connected cycles on $n = k2^k$ vertices, $b(G) \leq \lceil 5k/2 \rceil - 1$.*

Since it is desirable to be able to build networks for any size n , we now describe a simple way to construct modified cube-connected cycles. For any n where $k2^k < n < (k+1)2^{k+1}$, we construct the augmented cube-connected cycles on n vertices by adding $n - k2^k$ vertices to the cube-connected cycles on $k2^k$ vertices. If the vertices are added carefully, a result similar to Theorem 3 will hold for the augmented cube-connected cycles.

We construct the augmented cube-connected cycles on n vertices where $n = k2^k + j$ and $0 < j < (k+2)2^k$ by adding vertices to the cube-connected cycles on $k2^k$ vertices as follows. Let $i = \lfloor j/2^k \rfloor$. We distribute the j extra vertices among the 2^k cycles, giving $i+1$ additional vertices to $j - i2^k$ cycles and i vertices to the remaining cycles. Within each cycle, the additional vertices will be divided into groups as evenly as possible and each group will be inserted between adjacent cycle elements. If $i \leq k$ vertices are to be added to the cycle corresponding to cycle label x , one vertex is inserted between each pair (x, l) and $(x, (l+1) \bmod k)$ for $0 \leq l \leq i-1$. That is, the edge between (x, l) and $(x, (l+1) \bmod k)$ is replaced by an edge from (x, l) to a new vertex (x, l') and an edge from (x, l') to $(x, (l+1) \bmod k)$. If $k+1$ vertices are to be added to the cycle x , a path containing two new vertices is inserted between $(x, 0)$ and $(x, 1)$, and one new vertex is inserted between every other adjacent pair (x, l) and $(x, (l+1) \bmod k)$. Similarly, if $k+2$ vertices are to be added to the cycle x , paths containing two new vertices are inserted between $(x, 0)$ and $(x, 1)$ and between $(x, 1)$ and $(x, 2)$, while one new vertex is inserted between every other adjacent pair (x, l) and $(x, (l+1) \bmod k)$.

If we use essentially the same calling scheme to broadcast on this augmented cube-connected cycles, it is easy to see that at most $i+1$ delays (additional time units) will be added to each path used to inform the first informed member of each cycle. Thus, by time $2k+i$, at least one member of each cycle is informed. Each of the cycles may contain as many as $k+i+1$ vertices. The remaining uninformed elements of the cycle may be informed by calls analogous to those made in the scheme used for the cube-connected cycles and the broadcast can be completed by time $2k+i + \lceil (k+i+1)/2 \rceil$. Thus, we have Theorem 4.

THEOREM 4. *If G is the augmented cube-connected cycles on n vertices where $n = k2^k + j$ and $0 < j < (k+2)2^k$ and $i = \lfloor j/2^k \rfloor$, then $b(G) \leq 2k+i + \lceil (k+i+1)/2 \rceil$.*

5. Generalized shuffle-exchange graphs. In this section, we will discuss the use of *shuffle-exchange graphs* and several variants for broadcasting. The shuffle-exchange graph is a cubic graph on $n = 2^k$ vertices. Each vertex is labeled with a unique k bit string $x = x_{k-1} \cdots x_1 x_0$. The neighbors of x are $f(x) = x_{k-2} \cdots x_0 x_{k-1}$, $f^{-1}(x) = x_0 x_{k-1} \cdots x_1$, and $g(x) = x_{k-1} \cdots x_1 \bar{x}_0$. f and f^{-1} are the *shuffle* and *inverse shuffle* operations, respectively, and g is the *exchange* operation (see [13]). Notice that $g = g^{-1}$. It is not hard to show that the diameter of this graph is $2 \log_2 n - 1$. Consider two vertices $x = x_{k-1} \cdots x_1 x_0$ and $y = y_{k-1} \cdots y_1 y_0$. In the worst case, $y_i = \bar{x}_i$ for $0 \leq i \leq k-1$, so each bit of x must be complemented to get to y . Since the only way to complement a bit is with a g operation, at least $k-1$ of the bits must be “rotated” into the rightmost position. This gives a total of $\log_2 n - 1$ f operations and $\log_2 n$ g operations. While it appears that one more f operation is needed to rotate the low order bit back into the rightmost position, it is possible to avoid this rotation by comparing x_i with y_{i+1} instead of with y_i at each stage. (That is, compare x_0 to y_1 and complement if necessary to get $x_{k-1} \cdots x_1 y_1$, rotate, compare x_1 to y_2 , and complement if necessary to get $y_1 x_{k-1} \cdots x_2 y_2$, etc.)

Broadcasting in a shuffle-exchange graph is straightforward. Upon receipt of a message, a vertex first informs its f neighbor, and then its g neighbor. Of course, one or both of these operations may be unnecessary. If a vertex is informed by its g neighbor, then it only needs to inform its f neighbor, and, especially near the end of a broadcast, the f neighbor of a vertex may already be informed. It is clear that each vertex will receive the message after some sequence of at most $\log_2 n - 1$ f operations and $\log_2 n$ g operations. Since each vertex transmits to its f neighbor without delay, the f operations contribute at most $\log_2 n - 1$ to the broadcast time for a vertex. Each g operation may incur a delay of one time unit, so the g operations contribute at most $2 \log_2 n$ to the broadcast time.

THEOREM 5. *If G is a shuffle-exchange graph on $n = 2^k$ vertices, then $b(G) \leq 3 \log_2 n - 1$.*

A simple “folding” operation due to Fan Chung can be used to reduce both the diameter and the broadcast time of shuffle-exchange graphs. A *folded-shuffle-exchange graph* is obtained from a shuffle-exchange graph by identifying each vertex with the vertex in which all of its bits have been complemented. In other words, vertex $x = x_{k-1} \cdots x_1 x_0$ is identified with $\bar{x} = \bar{x}_{k-1} \cdots \bar{x}_1 \bar{x}_0$. Now, to get from a vertex $x = x_{k-1} \cdots x_1 x_0$ to a vertex $y = y_{k-1} \cdots y_1 y_0$ requires at most $\lfloor k/2 \rfloor$ g operations because at least half of the bits of y must be the same as the corresponding bits in either x or $\bar{x}$. $k-1$ f operations are still needed to rotate the bits, so the diameter of the folded-shuffle-exchange graph is $\lfloor 3k/2 \rfloor - 1$. Since $n = 2^{k-1}$ in a folded-shuffle-exchange graph, the diameter is $\lfloor 3/2 (\log_2 n + 1) \rfloor - 1$. The broadcast time for folded-shuffle-exchange graphs follows by the same argument as was used for shuffle-exchange graphs.

THEOREM 6. *If G is a folded-shuffle-exchange graph on $n = 2^{k-1}$ vertices, then $b(G) \leq 2 \log_2 n + 1$.*

Shuffle-exchange graphs can be generalized by letting the labels be strings of base α digits instead of bits. We will call these graphs α -*shuffle-exchange graphs*. Each vertex in an α -shuffle-exchange graph on $n = \alpha^k$ vertices is labeled with a unique k digit string $x = x_{k-1} \cdots x_1 x_0$ where $x_i \in [0, 1, \dots, \alpha - 1]$ for $0 \leq i \leq k-1$. The neighbors of x are $f(x) = x_{k-2} \cdots x_0 x_{k-1}$, $f^{-1}(x) = x_0 x_{k-1} \cdots x_1$, $g(x) = x_{k-1} \cdots x_1 (x_0 + 1) \bmod \alpha$, and $g^{-1}(x) = x_{k-1} \cdots x_1 (x_0 - 1) \bmod \alpha$. f and f^{-1} are the shuffle and inverse shuffle operations, respectively, and g and g^{-1} are the exchange and inverse exchange operations, respectively. In general, $g \neq g^{-1}$ and each vertex has degree 4. However, 2-shuffle-exchange graphs are ordinary shuffle-exchange graphs and each vertex has degree 3.

The diameter of α -shuffle-exchange graphs can be determined by generalizing the arguments for 2-shuffle-exchange graphs. Each digit of a vertex label is at most $\lfloor \alpha/2 \rfloor$ g

or g^{-1} operations away from the corresponding digit in any other label, so there are at most $\lfloor \alpha/2 \rfloor \log_\alpha n$ g and g^{-1} operations on the shortest path between any pair of vertices. In addition, at most $\log_\alpha n - 1$ f operations are needed, so the diameter is $(\lfloor \alpha/2 \rfloor + 1) \log_\alpha n - 1$.

To broadcast on an α -shuffle-exchange graph, each vertex, upon receipt of a message, informs its f , g , and g^{-1} neighbors in that order. The f operations incur no delays, so they contribute at most $\log_\alpha n - 1$ time units to the broadcast time. The g and g^{-1} edges of the graph can be grouped into cycles of length α since α consecutive g operations applied to a label x gives x . The shortest path between any pair of vertices includes parts of at most $\log_\alpha n$ such cycles. After the first vertex on such a cycle has been notified, it sends the message to its f neighbor if necessary, and then around its cycle in both directions to inform the other vertices on the cycle. After this, each vertex on the cycle that receives the message from its g neighbor sends it to its f and g^{-1} neighbors and each vertex that receives the message from its g^{-1} neighbor sends it to its f and g neighbors. The two paths will meet approximately halfway around the cycle. The g and g^{-1} operations are delayed one and two time units, respectively, at the first informed vertex on the cycle. Subsequent g and g^{-1} operations on the cycle are all delayed one time unit. It is easy to show that, after the first vertex on a cycle is informed, at most α time units are used to inform all other vertices on the cycle.

THEOREM 7. *If G is an α -shuffle-exchange graph on $n = \alpha^k$ vertices, then $b(G) \leq (\alpha + 1) \log_\alpha n - 1$.*

After converting the logarithms to base 2, the multiplicative constant is minimized when $\alpha = 4$.

COROLLARY 8. *If G is a 4-shuffle-exchange graph on $n = 4^k$ vertices, then $b(G) \leq 2.5 \log_2 n - 1$.*

The folding operation can be generalized to α -shuffle-exchange graphs by identifying each vertex $x = x^0 = x_{k-1} \cdots x_1 x_0$ with vertices

$$x^1 = (x_{k-1} + 1) \bmod \alpha \cdots (x_0 + 1) \bmod \alpha,$$

$$x^2 = (x_{k-1} + 2) \bmod \alpha \cdots (x_0 + 2) \bmod \alpha, \dots,$$

$$x^{\alpha-1} = (x_{k-1} + \alpha - 1) \bmod \alpha \cdots (x_0 + \alpha - 1) \bmod \alpha.$$

Thus, an α -shuffle-exchange graph with $n = \alpha^k$ vertices is folded $\alpha - 1$ times to obtain a *folded- α -shuffle-exchange graph* with $n = \alpha^{k-1}$ vertices. The broadcasting algorithm on a folded- α -shuffle-exchange graph is the same as the algorithm for α -shuffle-exchange graphs; each vertex informs its f , g , and g^{-1} neighbors in that order. As before, there are at most $k - 1 = \log_\alpha n$ f operations on the shortest path between any pair of vertices, and, since f operations incur no delays, they contribute at most $\log_\alpha n$ time units to the broadcast time. Also, as in the α -shuffle-exchange graphs, the g and g^{-1} edges of the graph can be grouped into cycles of length α and the shortest path between any pair of vertices includes parts of at most $\log_\alpha n$ such cycles. Ignoring, for the moment, that each vertex has α labels, the total number of g and g^{-1} operations on the shortest path between two vertices (or, more correctly, between two labels) $x = x^0 = x_{k-1} \cdots x_1 x_0$ and $y = y^0 = y_{k-1} \cdots y_1 y_0$ is the sum of the “distances” between pairs of corresponding digits. For example, if $x_i = 2$ and $y_i = 4$, then two g operations are needed to “fix” the i th digit, and if $x_i = 1$ and $y_i = \alpha - 1$, then two g^{-1} operations are needed. In both cases, the “distance” is two. Since each vertex has α labels, the total number of g and g^{-1} operations on the shortest path between two vertices x and y is the minimum over all pairs of labels x' and y' of the sum of the distances between all pairs of corresponding digits. (Note that x_i will actually be compared with y_{i+1} to eliminate

one of the rotations as explained at the beginning of this section. However, to simplify the following exposition we will assume that y_i is the digit corresponding to x_i .)

The shortest path between any two vertex labels $x = x^0 = x_{k-1} \cdots x_1 x_0$ and $y = y^0 = y_{k-1} \cdots y_1 y_0$ is an alternating sequence of g^* and f operations where a g^* operation is a sequence of zero or more g operations or zero or more g^{-1} operations. The result of a g^* operation is that the path travels part way around one of the cycles of length α changing one of the x_i 's into the corresponding y_i . Then an f operation is used to rotate the next digit into place (i.e., go to another cycle of length α), and a g^* operation is used to fix that digit, and so on. The shortest path between the two labels x^0 and $y^1 = (y_{k-1} + 1) \bmod \alpha \cdots (y_0 + 1) \bmod \alpha$ is similar to the path from x^0 to y^0 except that all of the g^* operations involving one or more g^{-1} operations will use one fewer g^{-1} operation, and most of the g^* operations involving zero or more g operations will use one more g operation. (The exception occurs halfway around the cycle where adding one more g operation would make the path longer than replacing the entire g^* operation with a sequence of g^{-1} operations.) It should be clear that the shortest path from x^r to y^s has the same length as the shortest path from x^0 to $y^{(s-r) \bmod \alpha}$, so it is only necessary to look at k pairs of labels (instead of k^2 pairs) to determine the shortest path between vertices x and y .

There are α different types of g^* operations possible (i.e., zero g or g^{-1} operations, one g operation, two g operations, $\cdots$, two g^{-1} operations, one g^{-1} operation). Each type of g^* operation has an associated weight d_i which is the number of its g or g^{-1} operations. Using a g operation of type i in a path contributes d_i to the length of the path. Figure 5.1 (a) shows a cycle of length α with the weights of the different g^* operations. The path enters the cycle at the vertex labeled 0, and the label on each vertex is the "cost" of leaving the cycle at that vertex.

To calculate the diameter of a folded- α -shuffle-exchange graph, we need the following lemma to determine how many g^* operations of each type are on the shortest path between the two vertices that are farthest apart.

LEMMA 9. *Let $\{p_0, p_1, \cdots, p_{\alpha-1}\}$ and $\{d_0, d_1, \cdots, d_{\alpha-1}\}$ be ordered sets of non-negative real numbers and let $\sum_{i=0}^{\alpha-1} p_i = k$. Then $\min_{0 \leq i \leq \alpha-1} \sum_{j=0}^{\alpha-1} p_j d_{(j+i) \bmod \alpha}$ is maximized when $p_i = k/\alpha$, $0 \leq i \leq \alpha-1$.*

Proof. Let D be an $\alpha \times \alpha$ array in which each row is a different cyclic permutation of the set $\{d_0, d_1, \cdots, d_{\alpha-1}\}$ and let

$$P = \{p_0, p_1, \cdots, p_{\alpha-1}\}^T \quad \text{and} \quad Q = \{q_0, q_1, \cdots, q_{\alpha-1}\}^T$$

be column vectors such that $Q = D \times P$ as shown below:

$$\begin{vmatrix} q_0 \\ \vdots \\ q_{\alpha-1} \end{vmatrix} = \begin{vmatrix} d_0 & d_1 & \cdots & d_{\alpha-1} \\ d_1 & d_2 & \cdots & d_0 \\ \vdots & \vdots & \ddots & \vdots \\ d_{\alpha-1} & d_0 & \cdots & d_{\alpha-2} \end{vmatrix} \times \begin{vmatrix} p_0 \\ \vdots \\ p_{\alpha-1} \end{vmatrix}.$$

So, $q_i = \sum_{j=0}^{\alpha-1} p_j d_{(j+i) \bmod \alpha}$ and we want to show that $\min \{q_i | 0 \leq i \leq \alpha-1\}$ is maximized when the p_i 's are all the same. Now

$$\sum_{i=0}^{\alpha-1} q_i = \sum_{i=0}^{\alpha-1} \sum_{j=0}^{\alpha-1} p_j d_{(j+i) \bmod \alpha} = \left[\sum_{j=0}^{\alpha-1} p_j \right] \times \left[\sum_{i=0}^{\alpha-1} d_i \right] = k \sum_{i=0}^{\alpha-1} d_i.$$

The d_i 's are fixed and k is a fixed constant independent of the relative values of the p_i 's so $\sum_{i=0}^{\alpha-1} q_i$ must also be a constant independent of the relative values of the p_i 's. It follows that $\min \{q_i | 0 \leq i \leq \alpha-1\}$ is maximized when the q_i 's are all the same. Since the q_i 's are all the same when the p_i 's are all the same, the result follows. $\square$

The d_i 's in the lemma are the weights associated with the different types of g^* operations. Each row of the matrix D corresponds to choosing a different label for one of the endpoints of a path. (If the label y^1 is used instead of y^0 , then all of the g^* operations of type i become type $(i + 1) \bmod \alpha$. If the label y^2 is used, then they all become type $(i + 2) \bmod \alpha$, etc.) The p_i 's are the numbers of each type of g^* operation that can be on a path between two vertices. The lemma shows that a maximum length shortest path between any pair of vertices occurs when the numbers of the different types of g^* operations on the path are the same. Using the weights (d_i 's) from Fig. 5.1(a), it follows that the diameter is at most

$$k - 1 + \left\lceil \frac{k}{\alpha} \right\rceil \sum_{i=1}^{(\alpha-1)/2} 2i = \log_{\alpha} n + \left\lceil \frac{\log_{\alpha} n + 1}{\alpha} \right\rceil \sum_{i=1}^{(\alpha-1)/2} 2i$$

when α is odd, and

$$k - 1 + \left\lceil \frac{k}{\alpha} \right\rceil \sum_{i=1}^{\alpha/2} (2i - 1) = \log_{\alpha} n + \left\lceil \frac{\log_{\alpha} n + 1}{\alpha} \right\rceil \sum_{i=1}^{\alpha/2} (2i - 1)$$

when α is even. The calculation of the broadcast time is similar to the calculation of the diameter, but the weights shown in Fig. 5.1(b) must be used instead to account for the delays.

THEOREM 10. *If G is a folded- α -shuffle-exchange graph on $n = \alpha^{k-1}$ vertices, then*

$$b(G) \leq \log_{\alpha} n + \left\lceil \frac{\log_{\alpha} n + 1}{\alpha} \right\rceil \sum_{i=1}^{(\alpha-1)/2} (4i + 1)$$

when α is odd, and

$$b(G) \leq \log_{\alpha} n + \left\lceil \frac{\log_{\alpha} n + 1}{\alpha} \right\rceil \left(\sum_{i=1}^{\alpha/2} (4i + 1) - \alpha - 1 \right)$$

when α is even.

After converting the logarithms in the expressions in Theorem 10 to base 2, the multiplicative constant is minimized when $\alpha = 4$.

COROLLARY 11. *If G is a folded-4-shuffle-exchange graph on $n = 4^{k-1}$ vertices, then $b(G) \leq 1.625 \log_2 n + 2.25$.*

For $\alpha \geq 3$, the α -shuffle-exchange and folded- α -shuffle-exchange graphs have degree 4. These graphs can be transformed into degree 3 graphs by replacing each vertex with

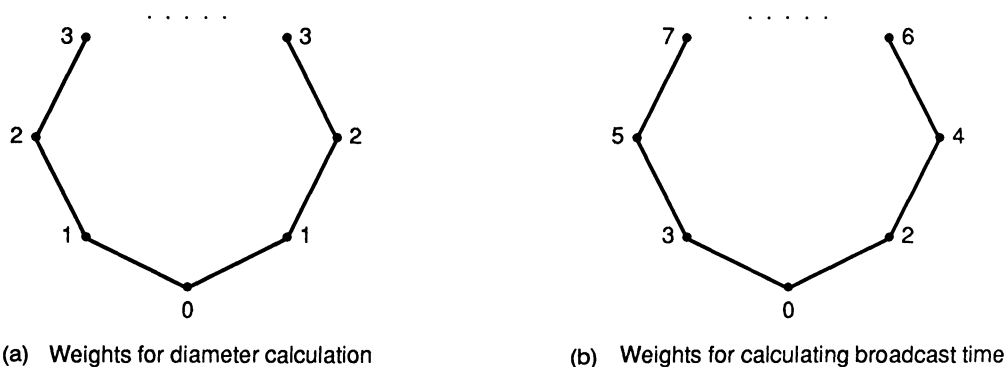
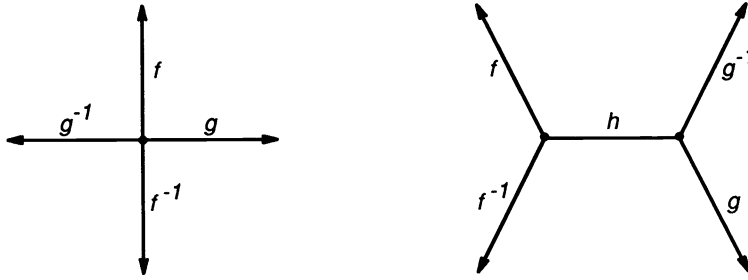


FIG. 5.1. Weights for use with Lemma 9.

FIG. 5.2. *H-extension operation.*

a pair of adjacent vertices as shown in Fig. 5.2. We call this operation an *H-extension*, and the resulting graphs are called *H-extended- α -shuffle-exchange graphs* and *H-extended-folded- α -shuffle-exchange graphs*. The broadcasting algorithm for these graphs is to first inform the h neighbor and then inform the other neighbor. It should be clear that the delays incurred by g and g^{-1} operations are the same as in the unextended graphs. The only difference is that the delays for g and g^{-1} operations are caused by h operations instead of f operations. It is also easy to see that each f operation incurs a delay of one, whereas f operations incurred no delays in the unextended graphs.

THEOREM 12. *If G is an H-extended-folded- α -shuffle-exchange graph on $n = 2\alpha^{k-1}$ vertices, then*

$$b(G) \leq 2(k-1) + \lceil k/\alpha \rceil \sum_{i=1}^{(\alpha-1)/2} (4i+1)$$

when α is odd, and

$$b(G) \leq 2(k-1) + \lceil k/\alpha \rceil \left(\sum_{i=1}^{\alpha/2} (4i+1) - \alpha - 1 \right)$$

when α is even.

After converting the logarithms in the expressions in Theorem 12 to base 2, the multiplicative constant is minimized when $\alpha = 6$.

COROLLARY 13. *If G is a H-extended-folded-6-shuffle-exchange graph on $n = 2 \times 6^{k-1}$ vertices, then $b(G) \leq 2.0632 \log_2 n + 1.2701$.*

6. Summary. We have investigated graphs with maximum degrees 3 and 4 in which broadcasting can be completed quickly from any vertex. We have given lower bounds on the time required to broadcast in such graphs and have given several constructions that produce good broadcast networks. In particular, we showed that

$$b(G) \geq 1.4404 \log_2 n - 1.769$$

for any graph G on n vertices with maximum degree 3. Our best upper bound on the broadcast time of such graphs is $b(G) \leq 2 \log_2 n + 1$ which is achieved by the folded-shuffle-exchange graphs on $n = 2^{k-1}$ vertices. In the case of maximum degree 4, we showed that $b(G) \geq 1.1374 \log_2 n - 0.637$. The best upper bound in the degree 4 case is $b(G) \leq 1.625 \log_2 n + 2.25$ obtained from folded-4-shuffle-exchange graphs on $n = 4^{k-1}$ vertices.

7. Acknowledgments. We thank Peter Winkler for suggesting this problem, Jean-Claude Bermond for telling us about Fan Chung's folding technique, and Romas Aleliunas for help with the proof of Lemma 9.

Note added in proof. The multiplicative constants in the upper bounds derived in this paper have been improved by Bermond, Hell, Liestman, and Peters [*Broadcasting in bounded degree graphs*, Technical Report CMPT TR 88-5, Simon Fraser University, Burnaby, British Columbia, Canada] to 1.8750 and 1.4167 for degrees 3 and 4, respectively. Upper and lower bounds for larger degrees are also presented there.

REFERENCES

- [1] B. W. ARDEN AND H. LEE, *Analysis of chordal ring network*, IEEE Trans. Comput., 30 (1981), pp. 291–295.
- [2] J. C. BERMOND, private communication, January 1981.
- [3] S. C. CHAU AND A. L. LIESTMAN, *Constructing minimal broadcast networks*, J. Combin., Inform. System Sci., 10 (1985), pp. 110–122.
- [4] A. FARLEY, *Minimal broadcast networks*, Networks, 9 (1979), pp. 313–332.
- [5] A. FARLEY, S. HEDETNIEMI, S. MITCHELL, AND A. PROSKUROWSKI, *Minimum broadcast graphs*, Discrete Math., 25 (1979), pp. 189–193.
- [6] M. GAREY AND D. JOHNSON, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, 1979.
- [7] A. KUSALIK, private communication, August 1982.
- [8] A. L. LIESTMAN, *Fault-tolerant broadcast graphs*, Networks, 15 (1985), pp. 159–171.
- [9] S. MITCHELL AND S. HEDETNIEMI, *A census of minimum broadcast graphs*, J. Combin., Inform. Systems Sci., 5 (1980), pp. 141–151.
- [10] D. PELEG, *Tight bounds on minimum broadcast graphs*, July 1987, manuscript.
- [11] F. P. PREPARATA AND J. VUILLEMIN, *The cube-connected cycles: a versatile network for parallel computation*, Comm. ACM, 24 (1981), pp. 300–309.
- [12] P. W. PURDOM, JR. AND C. A. BROWN, *The Analysis of Algorithms*, Holt, Rinehart, and Winston, New York, 1985, § 5.3.
- [13] H. S. STONE, *Parallel processing with the perfect shuffle*, IEEE Trans. Comput., 20 (1971), pp. 153–161.
- [14] X. WANG, private communication, April 1986.

DISTRIBUTION OF THE SYMMETRIC DIFFERENCE METRIC ON PHYLOGENETIC TREES*

M. A. STEEL†

Abstract. The symmetric difference metric has been useful in comparing phylogenetic trees derived from DNA sequence data. The main result shown here is that the frequency of pairs of binary trees a given distance apart is described by a limiting Poisson distribution, with $e^{-1/8} \approx 88$ percent of all pairs maximally distant. Asymptotic bounds on the distribution are derived, and the asymptotic mean and variance of the normalized metric on the class of all phylogenetic trees is also calculated. The results rely on simple combinatorial constructions and analytic properties of appropriate generating functions.

Key words. trees, binary trees, phylogenetic trees, probability, symmetric difference metric, asymptotic distribution

AMS(MOS) subject classifications. 05C05, 60C05, 92A12

Introduction. The symmetric difference metric defined on phylogenetic trees is a special case of symmetric difference metrics on sets studied in [9], [12]. The tree metric has been useful in testing evolutionary hypotheses and in examining the methods used to build evolutionary trees [11]. An optimally efficient algorithm has been developed by Day [4] to compute the metric, and its distribution among pairs of small trees is described in [3] and [8].

This paper extends those results to obtain bounds on the distribution of pairs of arbitrarily-large binary trees a given distance apart. As a result, the asymptotic distribution is shown to be Poisson, which answers a conjecture in [8]. The distribution of the normalized metric on the full class of phylogenetic trees is also examined. In particular, the asymptotic mean of the normalized metric is derived, and confirms a second conjecture in [8].

A description of the distribution in the binary case is also given in terms of a generating polynomial, the coefficients of which can be computed directly from the tree. Some consequences of this representation are given. The properties of the metric on binary trees make it useful for hypothesis testing involving trees derived from homologous DNA sequences, as in [10]. The resulting trees may be expected to be similar and it is useful to have a metric for which most trees are far apart.

DEFINITIONS. Let L be a set of $n \geq 2$ labels. A *phylogenetic tree on L* is a tree with n vertices of degree one, each labeled with a distinct element from L , and with the remaining (internal) vertices of degree at least three, and unlabeled. For such a tree, the n edges incident with a pendant vertex are called *pendant edges*, and the remaining edges are *internal*.

Two phylogenetic trees on L are considered equivalent if there is a graph isomorphism between them that preserves the labeling on the pendant vertices. More generally, if two phylogenetic trees are graph isomorphic with their labelings suppressed, we say they are *topologically equivalent*.

As in [8], let $PT(n, f)$ denote the set of *phylogenetic trees* with n pendant vertices and f internal edges on the label set $\{1, \dots, n\}$. For $n \geq 3$, let $PT(n) = \bigcup \{PT(n, f); 0 \leq f \leq n-3\}$, and $BPT(n) = PT(n, n-3)$, the set of *binary phylo-*

* Received by the editors July 13, 1987; accepted for publication (in revised form) May 5, 1988.

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

genetic trees, for which each internal vertex has degree three. The following result is from [5, p. 29].

LEMMA 1. *The size of $PT(n, f)$ is determined recursively as follows:*

$$|PT(n, f)| = (n + f - 2) |PT(n - 1, f - 1)| + (f + 1) |PT(n - 1, f)|, \quad n \geq 4,$$

$$|PT(3, 0)| = 1, \quad |PT(3, f)| = 0, \quad f > 0.$$

For $n \geq 3$, Lemma 1 gives $|BPT(n)| = (2n - 5)!! = 1.3.5 \cdots (2n - 5)$. For convenience, we let $b(n) = (2n - 5)!!$ and $p(n) = |PT(n)|$.

The symmetric difference metric d , which Bourque [2], and Robinson and Foulds [13] applied to phylogenetic trees, is defined on $PT(n)$, and so on $BPT(n)$, as follows: For $T \in PT(n)$, deletion of an internal edge e induces a two-set partition $\pi(T, e)$ of $\{1, \dots, n\}$ corresponding to the labels on the two connected components of T with e deleted. For $T_1 \in PT(n_1, f_1)$, $T_2 \in PT(n_2, f_2)$, and $\pi(T_1, e_1) = \pi(T_2, e_2)$ we call e_1, e_2 an *equivalent pair of edges*. If T_1, T_2 have exactly m equivalent pairs of edges then $d(T_1, T_2) = f_1 + f_2 - 2m$. In particular for $T_1, T_2 \in BPT(n)$,

$$d(T_1, T_2) = 2(n - 3 - m).$$

For $T \in PT(n)$ we recall from [8] the generating polynomials

$$P(T) = P(T, x) = \sum_{m \geq 0} p_m(T) x^m, \quad Q(T) = Q(T, x) = \sum_{m \geq 0} q_m(T) x^m$$

where $p_m(T)$ (respectively, $q_m(T)$) is the number of trees in $PT(n)$ (respectively, $BPT(n)$) at distance m from T .

Thus, for $T \in PT(n, f)$, $Q(T, x)$ has degree $n + f - 3$ and is an even or odd polynomial of parity equal to the numerical parity of its degree. For $T \in PT(n)$, $s \geq 0$ let $q(s, T)$ denote the number of binary trees having s equivalent edge pairs with T , and let $q(s, n)$ be the average value of $b(n)^{-1} q(s, T)$ over $BPT(n)$. Thus, $q(s, n) = b(n)^{-2} \sum_{T \in BPT(n)} q(s, T)$ is the probability that two trees randomly chosen from $BPT(n)$ have exactly s equivalent pairs of edges.

The main result (Theorem 3) shows that $q(s) = \lim_{n \rightarrow \infty} q(s, n)$ has a Poisson distribution in s with mean $\frac{1}{8}$. Consequently, the probability that two randomly chosen trees are a maximal distance apart tends asymptotically to $e^{-1/8}$, answering a question raised in [8].

We begin by noting that for $T \in PT(n)$, $P(T)$ and $Q(T)$ do not depend on the labeling of T , but only on its topology. We shall frequently write these and other tree-valued functions that are invariant under topological equivalence without specifying the labeling of the tree. With this in mind from [8], we now state the following theorem.

THEOREM 1. *Let e be an internal edge of $T \in PT(n)$. Let T/e be the tree formed by contracting e , and let T_1, T_2 be the maximal subtrees of T with e as a pendant edge. Then*

$$P(T) = xP(T/e) + (1 - x^2)P(T_1)P(T_2),$$

$$Q(T) = xQ(T/e) + (1 - x^2)Q(T_1)Q(T_2).$$

We now give a constructive description of $Q(T, x)$. Let $T \in PT(n, f)$ and let E be a set of internal edges of T . For each edge $e \in E$ cut e in half and place new pendant vertices on each of the two "ends" of e . In this way E defines a collection of trees T_i , having n_i pendant vertices, for $i = 1, \dots, |E| + 1$ (with $T_1 = T$, if $E = \emptyset$). Clearly, $\sum_{1 \leq i \leq |E| + 1} n_i = n + 2|E|$. Let $\Phi(E)$ be the sequence $(n_1, \dots, n_{|E| + 1})$, taken in some

order, and let $\langle \Phi(E) \rangle = \prod_{1 \leq i \leq |E|+1} b(n_i)$. Define $r(s, T)$ to be the sum of $\langle \Phi(E) \rangle$ over all sets of internal edges E , with $|E| = s$. Finally, let

$$R(T) = R(T, x) = \sum_{s \geq 0} r(s, T) x^s, \quad q(T) = q(T, x) = \sum_{s \geq 0} q(s, T) x^s$$

(so that $q(T, x) = x^{(n-3+f)/2} Q(T, x^{-1/2})$).

LEMMA 2. For $T \in \text{PT}(n)$, $T_n \in \text{PT}(n, 0)$, we have the following:

- (a) $q(T, x) = R(T, x - 1)$;
- (b) In the notation of Theorem 1,
 - (i) $R(T) = R(T/e) + xR(T_1)R(T_2)$,
 - (ii) $q(T) = q(T/e) + (x - 1)q(T_1)q(T_2)$;
- (c) $R(T, 0) = R(T_n, x) = b(n)$.

Proof. (a) Let $T \in \text{BPT}(n)$ and let E be a set of s internal edges of T . Under the above construction, for each edge $e \in E$, new pendant vertices v_1, v_2 are attached to the ends of a bisection of e .

For $i = 1, 2$, label v_i with the set of labels of those pendant vertices of T that are no longer joined by a path to v_i when e is cut. Each tree T_i ($i = 1, \dots, s+1$) defined by E , thus has a natural label set L_i for its pendant vertices, so that $T_i \in \text{BPT}(L_i)$. This process is illustrated for $s = 2$ in Fig. 1 by the tree J_6 with two distinguished edges.

Now, let $B(T, E)$ be the set of trees in $\text{BPT}(n)$ equivalent to T on E and possibly other internal edges. We construct a bijection F from $B(T, E)$ to $\prod_i \text{BPT}(L_i)$. Given $T' \in B(T, E)$, performing the above edge splitting and labeling procedure on T' produces the label sets $L_1, \dots, L_{s+1}$, and hence an element of $F(T') \in \prod_i \text{BPT}(L_i)$.

The inverse of F , takes $(T_1, \dots, T_{s+1}) \in \prod_i \text{BPT}(L_i)$ and identifies all pairs of pendant vertices v_1, v_2 labeled with sets A_1, A_2 such that $A_1 \cup A_2 = L$, the identified vertices then being suppressed to give a tree in $B(T, E)$. Now, $|L_i| = n_i$ implies $|\text{BPT}(L_i)| = b(n_i)$, so that the bijection gives $|B(T, E)| = \langle \Phi(E) \rangle$. By the principle of inclusion and exclusion [6] $r(T, x - 1)$ is then the (ordinary) generating function for the number of binary trees equivalent to T on an exact number of internal edges, thus we establish (a).

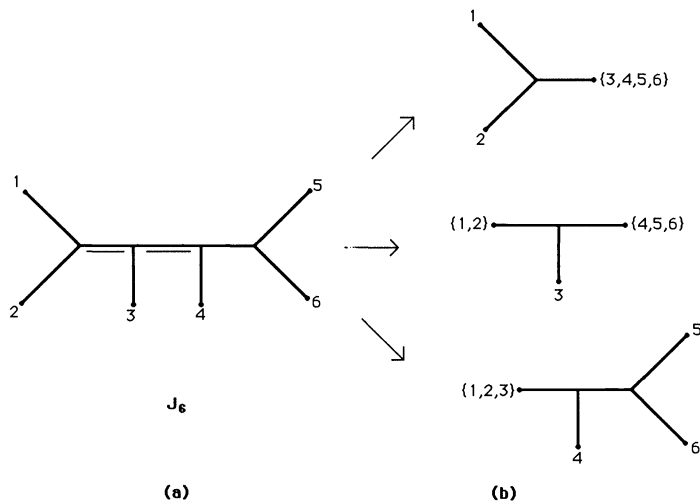


FIG. 1

Part (b)(i) can be proved directly, or from Theorem 1 by noting $Q(T, x) = x^{n-3+f} R(T, x^{-2} - 1)$. Part (b)(ii) follows from (a), while part (c) follows from the definition of $R(T, x)$.

Example 1. For J_6 in Fig. 1(a), we have

$$\begin{aligned} R(J_6, x) &= b(6) + (2b(3)b(5) + b(4)^2)x + 3b(3)^2b(4)x^2 + b(3)^2x^3 \\ &= 105 + 39x + 9x^2 + x^3, \\ q(J_6, x) &= R(T, x - 1) = 74 + 24x + 6x^2 + x^3, \\ Q(J_6, x) &= 1 + 6x^2 + 24x^4 + 74x^6. \end{aligned}$$

Example 2. Let $T \in PT(n, f)$. Then $q(s, T) = 0$, for $s > f$, and $q(f, T) = \prod_i b(\partial_i)$, where $(\partial_1, \dots, \partial_{f+1})$ is the degree sequence of the internal vertices of T .

DEFINITION. For $n \geq 2a > 4$, let $T_n(a) \in PT(n)$ be a tree obtained by attaching pairs of pendant vertices to “ a ” pendant vertices of a (star) tree $T \in PT(n - a, 0)$. The resulting tree is unique up to topology so that $q(s, T_n(a))$ is well defined. We call such trees *binary semistars*. The tree J_6 , in Fig. 1(a), with its central edge contracted, is a $T_6(2)$.

To calculate $q(s, T_n(a))$, let F be the set of internal edges of $T_n(a)$, so that $|F| = a$, and for $E \subseteq F$, $\Phi(E)$ consists of $|E|$ copies of three and one copy of $n - |E|$, giving $\langle \Phi(E) \rangle = b(n - |E|)$. Thus, $r(s + i, T) = {}^a C_{(s+i)} b(n - s - i)$, and so by Lemma 2,

$$q(s, T_n(a)) = \sum_{i \geq 0} (-1)^i \cdot {}^{(s+i)} C_s \cdot {}^a C_{(s+i)} \cdot b(n - s - i).$$

Rearranging terms, we obtain Lemma 3.

LEMMA 3. $q(s, T_n(a)) = {}^a C_s \sum_{i \geq 0} (-1)^i \cdot {}^{(a-s)} C_i \cdot b(n - s - i)$.

DEFINITIONS. For $T \in PT(n)$, $n > 4$, e is a *binary edge* of T if e is an internal edge adjacent to a pair of adjacent pendant edges. For $T \in PT(n)$, let $a(T)$ be the number of binary edges of T .

For $T \in PT(n)$, $n > 4$, if e is a nonbinary internal edge of T , we say the contracted tree T/e is a σ -reduction of T . Equivalently it is a contraction for which the maximal subtrees T_1, T_2 both have at least four pendant vertices.

LEMMA 4. For $T \in PT(n, f)$, $n > 4$, $a(T) = a$, T can be successively σ -reduced to a $T_n(a)$ in $f - a$ steps.

Proof. If $T' \in PT(n)$ is not a binary semistar, T' has a subtree topologically equivalent to either J_6 in Fig. 1(a), or to J_6 with a contracted noncentral internal edge. Since these trees are σ -reducible, T' is also. By induction, T can thus be reduced to a binary semistar $T_n(a)$. Since the internal edges that survive under all possible σ -reductions are precisely the binary edges, we have $a(T) = a$, the number of binary edges of $T_n(a)$. Finally since each σ -reduction eliminates one internal edge, $f - a$ steps are required.

Now for positive integers t, s , let $f(x)$ be a positive, real-valued function defined on integers greater than or equal to s , with the property that $t < x \leq y$ implies $f(x)f(y) \leq f(x-1)f(y+1)$. Thus, for positive integers N, k , with $N \geq kt$, induction on N for each k gives

$$\max \left\{ \prod_{1 \leq i \leq k} f(x_i) : \sum_i x_i = N, x_i \geq t \right\} = f(t)^{k-1} f(N - (k-1)t).$$

Taking $t \geq 3$, $f(x) = b(x)$, we have, for $x \leq y$, $b(x)b(y) = (2x-5)b(x-1)b(y) \leq (2y-5)b(x-1)b(y) = b(x-1)b(y+1)$, so that $b(x)$ satisfies the above property.

This gives the following lemma.

LEMMA 5. For positive integers $t \geq 3$, N , k ;

$$\max \left\{ \prod_{1 \leq i \leq k} b(x_i) : \sum_i x_i = N, x_i \geq t \right\} = b(t)^{k-1} b(N - (k-1)t).$$

Example 3. For $T \in PT(n, f)$, $n \geq 3$, $s \geq 0$, $q(s, T) \leq {}^f C_s b(n-s)$, so that

$$\lim_{n \rightarrow \infty} b(n)^{-1} q(s, T) \leq 2^{-s}/s!.$$

Proof of Example 3. By Lemma 2(a), $q(s, T) \leq \sum_{i \geq 0} {}^{(s+i)} C_s q(s+i, T) = r(s, T)$. From Example 2 we may assume $s \leq f \leq n-3$, so that for a set E of s internal edges of T , if $\Phi(E) = (n_1, \dots, n_{s+1})$, then $\sum_i n_i = n + 2s \geq 3(s+1) \geq \min \{n_i\} \cdot s$. Thus, we can apply Lemma 5 with $t = 3$, $N = n + 2s$, $k = s + 1$, to obtain $\langle \Phi(E) \rangle \leq b(3)^s b(n-s) = b(n-s)$, and since there are ${}^f C_s$ possible choices for E , $r(s, T) \leq {}^f C_s \cdot b(n-s)$, which completes the proof.

THEOREM 2. For $T \in PT(n)$, $s \geq 0$, $n > 4$,

$$b(n)^{-1} q(s, T) = b(n)^{-1} q(s, T_n(a)) + \delta(s, T),$$

where $a = a(T)$, $|\delta(s, T)| < 3(s+1)/2(2n-7)$.

Proof. If $n = 5$, T is a binary semistar and the theorem holds, then suppose $n \geq 6$. By Lemmas 2(a) and 2(b),

$$(1) \quad q(s, T) = q(s, T/e) + \sum_{\alpha + \beta = s-1} q(\alpha, T_1) q(\beta, T_2) - \sum_{\alpha + \beta = s} q(\alpha, T_1) q(\beta, T_2).$$

Now if T/e is a σ -reduction of T , each of the two summation terms in (1) does not exceed $(s+1)b(4)b(n-2)$ (since for $T \in PT(m)$, $q(s, T) \leq b(m)$, and for $n \geq 6$, Lemma 5 applies with $k = 2$, $t = 4$, $N = n + 2$). Thus, by Lemma 4, $b(n)^{-1} q(s, T) = b(n)^{-1} q(s, T_n(a)) + \delta(s, T)$; $a = a(T)$, where

$$\begin{aligned} |\delta(s, T)| &\leq (s+1)(n-3-a)b(4)b(n-2)/b(n) \\ &= 3(s+1)(n-3-a)/(2n-5)(2n-7) \\ &< 3(s+1)/2(2n-7). \end{aligned}$$

LEMMA 6. Let

$$\eta_n(a) = |\{T \in BPT(n) : a(T) = a\}|,$$

$n > 4$, and $T_n(x) = \sum_a \eta_n(a) x^a$, $t_n(s) = D^s(T_n(x))|_{x=1}$, where D denotes differentiation with respect to x . Then, $t_n(s) = 2^{-s} b(n-s) n! / (n-2s)!$.

Proof. Given a set of labels L , let $\pi_2(L, k)$ be the collection of all sets of k disjoint sets of size two drawn from L , and let $\pi_2(n, k) = \pi_2(\{1, \dots, n\}, k)$. For $S \in \pi_2(n, k)$, let $A(S) \subseteq BPT(n)$ be the set of binary trees for which any pair of pendant vertices labeled by a pair from S have adjacent edges. If $A(n, k) = \sum_{S \in \pi_2(n, k)} |A(S)|$, and $A_n(x) = \sum_j A(n, k) x^j$, then by the principle of inclusion and exclusion [6] $T_n(x) = A_n(x-1)$, giving $t_n(s) = s! A(n, s)$.

Let $V(n, s) = \{(T, S) : T \in A(S), S \in \pi_2(n, s)\}$, and let

$$W(n, k) = \{(T, G, H) : T \in BPT(n-s), G \in \pi_2(H, s), H \subseteq \{1, \dots, n\}, |H| = 2s\}.$$

Then $A(n, s) = |V(n, s)|$, and we construct a bijection F from $V(n, s)$ to $W(n, s)$ as follows. Given $(T, S) \in V(n, s)$, let $H(S) = \cup \{X : X \in S\}$, so that $|H(S)| = 2s$, and $S \in \pi_2(H(S), s)$.

Let $T(S)$ be the tree obtained by deleting each pair of pendant vertices (and their pendant edges) with labels x_1, x_2 in S , and relabeling the exposed internal vertex by $\min \{x_1, x_2\}$. Relabeling $T(S)$ again by $\{1, \dots, n-s\}$ so as to preserve the order of labels gives a tree $T'(S) \in BPT(n-s)$. Let $F(T, S) = (T'(S), S, H(S))$. Then F has the following inverse (and hence is a bijection). Given $(T, G, H) \in W(n, s)$, if $G = \{\{x_1, y_1\}, \dots, \{x_s, y_s\}\}$, let $L(G) = \{1, \dots, n\} - \cup_i \{\max \{x_i, y_i\}\}$. Given $T \in BPT(n-s)$, relabel the pendant vertices of T with $L(G)$ so as to preserve the order of the labels. Then, for $i = 1, \dots, s$, join new pendant vertices labeled x_i , and y_i to the pendant vertex of T labeled $\min \{x_i, y_i\}$, and we obtain a tree $T' \in A(G)$. Thus, $F^{-1}((T, G, H)) = (T', G)$.

Summarizing, we have $t_n(s) = s!A(n, s) = s!|V(n, s)| = s!|W(n, s)|$. But $|W(n, s)| = b(n-s)|\pi_2(2s, s)|^nC_{2s}$, and $|\pi_2(2s, s)| = (2s)!/s!2^s[1]$, which proves the lemma.

Remark 1. An argument in [7] shows that

$$\eta_n(a) = \begin{cases} n!(n-4)!/(n-2a)!a!(a-2)!2^{2a-2} & \text{for } 2 \leq a \leq [n/2], \\ 0 & \text{otherwise.} \end{cases}$$

The proof relies on a recurrence for $\eta_n(a)$ which can be written

$$T_n(x) = (n-4+nx-x)T_{n-1}(x) + 2(x-x^2)d/dxT_{n-1}(x).$$

An inductive argument based on this result gives an alternative proof of Lemma 6.

THEOREM 3. For $n > 4$, $s \geq 0$, $q(s, n) = \psi(s, n) + \delta_n(s)$, where $\psi(s, n) = (n!/b(n)^22^s s!) \sum_i b(n-s-i)^2/(-2)^i i!(n-2s-2i)!$, $|\delta_n(s)| < 3(s+1)/2(2n-7)$.

Proof.

$$\begin{aligned} q(s, n) &= b(n)^{-2} \sum_{T \in BPT(n)} q(s, T) \\ &= b(n)^{-2} \left\{ \sum_a q(s, T_n(a)) \eta_n(a) \right\} + \delta_n(s) \quad \text{by Theorem 2.} \end{aligned}$$

By Lemma 3,

$$\begin{aligned} \sum_{a \geq 0} q(s, T_n(a)) \eta_n(a) &= \sum_{a \geq 0} \left\{ {}^a C_s \sum_{i \geq 0} (-1)^i {}^{(a-s)} C_i b(n-s-i) \right\} \eta_n(a) \\ &= \sum_{i \geq 0} \left\{ \sum_{a \geq 0} {}^a C_s {}^{(a-s)} C_i \eta_n(a) \right\} (-1)^i b(n-s-i). \end{aligned}$$

Now, $\sum_{a \geq 0} {}^a C_s {}^{(a-s)} C_i \eta_n(a) = t_n(s+i)/s!i!$, so that

$$b(n)^{-2} \sum q(s, T_n(a)) \eta_n(a) = (s!)^{-1} \sum_{i \geq 0} (-1)^i t_n(s+i) b(n-s-i)/b(n)^2 i!$$

$$= \psi(s, n) \quad \text{by Lemma 6, as required.}$$

COROLLARY 1. $q(s) = \lim_{n \rightarrow \infty} q(s, n) = e^{-1/8}/8^s s!$.

Proof. The proof is obtained by Theorem 3, observing that

$$\lim_{n \rightarrow \infty} b(n-x)^2 n! / b(n)^2 (n-2x)! = 4^{-x} \quad \text{and} \quad \lim_{n \rightarrow \infty} \delta_n(s) = 0.$$

COROLLARY 2. If $v(n)$ is the expected distance between two trees in $BPT(n)$, and $\sigma^2(n)$ is the variance, then we have the following:

(a) $\lim_{n \rightarrow \infty} (2n-6) - v(n) = 0.25$;

(b) $\lim_{n \rightarrow \infty} \sigma^2(n) = 0.5$.

Proof. (a) $v(n) = \sum_s q(s, n)(2n - 6 - 2s)$; thus,

$$(2n - 6) - v(n) = (2n - 6) \left(1 - \sum_s q(s, n) \right) + 2 \sum_s sq(s, n) = 2 \sum_s sq(s, n),$$

since $\sum_s q(s, n) = 1$, and $\lim_{n \rightarrow \infty} \sum_s sq(s, n) = \sum_s sq(s) = \frac{1}{8}$, by Corollary 1.

(b)

$$\begin{aligned} \sigma^2(n) &= \sum_s q(s, n)((2n - 6 - 2s) - v(n))^2 \\ &= ((2n - 6) - v(n))^2 \sum_s q(s, n) - 4(2n - 6 - v(n)) \sum_s sq(s, n) \\ &\quad + 4 \sum_s s^2 q(s, n). \end{aligned}$$

Letting $n \rightarrow \infty$, using (a) and noting that $\sum_s s^2 q(s) = 0.125 + (0.125)^2$, we obtain the result.

Remark 2. Using a different type of counting argument it can be shown that $q(0, n)$ is monotone increasing in n , for $n \geq 3$. Table 4 of [8], which gives $q(s, n)$ for $4 \leq n \leq 16$, further suggests that for each $s > 0$, $q(s, n)$ is monotone decreasing in n .

Having found the asymptotic average value over $BPT(n)$ of $b(n)^{-1}q(s, T)$, we now calculate its asymptotic range.

THEOREM 4.

$$\begin{aligned} \lim_{n \rightarrow \infty} \sup_{m \geq n, T \in BPT(m)} \{ b(m)^{-1}q(s, T) \} &= \begin{cases} 1, & s = 0, \\ e^{-1/4} / 4^s s!, & s > 0, \end{cases} \\ \lim_{n \rightarrow \infty} \inf_{m \geq n, T \in BPT(m)} \{ b(m)^{-1}q(s, T) \} &= \begin{cases} e^{-1/4}, & s = 0, \\ 0, & s > 0. \end{cases} \end{aligned}$$

Proof. For $s = 0$, (1) gives $q(0, T) \geq q(0, T_n(a))$, which together with Lemma 3 and Theorem 2 gives $q(0, T) \geq 1 - a(T)/(2n - 5) + \delta(0, T)$.

For each positive integer n , choose $J_n \in BPT(n)$ with $a(J_n) = 2$. For a given n , any two such trees are topologically equivalent and are often called caterpillar trees. Then $b(n)^{-1}q(0, J_n) \geq 1 - 2/(2n - 5) + \delta(0, J_n) \rightarrow 1$, as $n \rightarrow \infty$, and since

$$b(m)^{-1}q(0, T) \leq 1$$

for all $T \in BPT(m)$, $\limsup \{ b(m)^{-1}q(0, T) \} = 1$. A similar argument using J_n gives $\liminf \{ b(m)^{-1}q(0, T) \} = 0$ for $s > 0$. To obtain the other two results, consider the family of binary trees, $K_n \in BPT(2n)$, obtained by attaching pairs of pendant edges to every pendant vertex of a caterpillar tree $T \in J_n$. For fixed n the trees obtained are again topologically equivalent. An example of a K_5 is given in Fig. 2. Clearly $a(K_n) = n$, so that by Lemma 3 and Theorem 2,

$$b(2n)^{-1}q(s, K_n) = {}^nC_s \sum_{i \geq 0} (-1)^i \cdot {}^{(n-s)}C_i \cdot b(2n - i - s) / b(2n) + \delta(s, K_n).$$

Rearranging, we have

$$\begin{aligned} (s!)^{-1} \sum_{i \geq 0} (-1)^i (i!)^{-1} \{ n(n-1) \cdots (n-s-i+1) \\ / (2(2n-5) \cdots (2(2n)-2i-2s-3)) \} + \delta(s, K_n). \end{aligned}$$

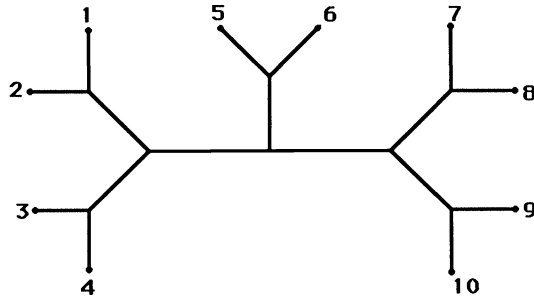


FIG. 2

Now the bracketed term has $s + i$ factors in the numerator and denominator so that as $n \rightarrow \infty$, the bracketed term approaches 4^{-s-i} . Hence,

$$\begin{aligned} \lim_{n \rightarrow \infty} b(2n)^{-1} q(s, K_n) &= (4^s s!)^{-1} \sum_{i \geq 0} \left(-\frac{1}{4}\right)^i / i! \\ &= e^{-1/4} / 4^s s!. \end{aligned}$$

Thus,

$$\begin{aligned} \limsup \{ b(m)^{-1} q(s, T) \} &\geq e^{-1/4} / 4^s s!, \\ \liminf \{ b(m)^{-1} q(0, T) \} &\leq e^{-1/4}. \end{aligned}$$

We now show $\alpha = \liminf \{ b(m)^{-1} q(0, T) \} \geq e^{-1/4}$ and by a similar argument, the other inequality for $\limsup$ can be derived, thus establishing the theorem. Let $T_i \in BPT(n_i)$ be a sequence with $\lim_{i \rightarrow \infty} b(n_i)^{-1} q(0, T_i) = \alpha$. Then since $a(T_i)/n_i$ is bounded, it has a convergent subsequence $a(T_{i(j)})/n_{i(j)}$. Let $\gamma = \lim_{j \rightarrow \infty} a(T_{i(j)})/n_{i(j)}$. Since $T_{i(j)}$ is a subsequence of T_i , $\lim_{j \rightarrow \infty} b(n_{i(j)})^{-1} q(0, T_{i(j)}) = \alpha$. A calculation similar to the above result on $b(2n)^{-1} q(s, K_n)$ shows $\alpha = e^{-\gamma/2}$. But for any $T \in BPT(n)$, $a(T) \leq n/2$, so that $\gamma \leq \frac{1}{2}$. Thus $\alpha \geq e^{-1/4}$, as required.

We now consider the distribution of the symmetric difference metric on $PT(n)$. The normalized distance between two trees $T, T' \in PT(n)$ is $d(T, T')$ divided by the maximum possible distance, $2n - 6$. Theorem 5 shows that, asymptotically, the normalized distance becomes increasingly peaked about its mean $\mu(n)$, which is shown to be less than one, confirming a conjecture in [8]. The symbols O and $\sim$ have their usual meaning: $f(n) \sim g(n)$ means $\lim_{n \rightarrow \infty} f(n)/g(n) = 1$, and $f(n) = O(g(n))$ means $f(n)/g(n)$ is bounded as $n \rightarrow \infty$. Let $\rho = 2 \ln 2 - 1$. Lemma 7 follows from similar results in [14].

LEMMA 7.

- (a) $p(n)/n! \sim \rho^{1-n} n^{-5/2} \sqrt{(\rho/4\pi)}$.
- (b) $p(n)^{-1} (\sum_{f \geq 0} f |PT(n, f)|) - n\rho^{-1}(1 - \ln 2) = O(1)$.
- (c) $p(n)^{-1} (\sum_{f \geq 0} f(f-1) |PT(n, f)|) - n^2 \rho^{-2}(1 - \ln 2)^2 = O(n)$.

THEOREM 5. Let $\mu(n)$ and $\sigma^2(n)$ denote, respectively, the mean and variance of the normalized distance between two trees in $PT(n)$. Then, we have the following:

- (a) $\mu(n) \sim (1 - \ln 2)/\rho \approx .7943$;
- (b) $\sigma^2(n) = O(n^{-1})$.

Proof. A straightforward argument using Lemma 7(a) gives a constant C_1 such that:

$$(2) \quad \text{if } n_1, n_2 \geq 3, \quad n_1 + n_2 = n + 2, \quad p(n_1)p(n_2)/p(n) < C_1/n.$$

Let $T \in PT(n, f)$ and $\mu(T) = p(n)^{-1} DP(T, x)|_{x=1}$, ($D = d/dx$) be the expected distance between T and trees in $PT(n)$.

By Theorem 1,

$$DP(T, x)|_{x=1} = P(T/e, 1) + DP(T/e, x)|_{x=1} - 2P(T_1, 1)P(T_2, 1).$$

Hence, $\mu(T) = 1 + \mu(T/e) - 2p(n_1)p(n_2)/p(n)$, where $T_1 \in PT(n_1)$, $T_2 \in PT(n_2)$. By contracting T to $T_n \in PT(n, 0)$, (2) gives

$$(3) \quad \mu(T) = f + \mu(T_n) - E(T) \quad \text{where } 0 < E(T) < 2C_1 f/n.$$

Averaging $\mu(T)$ over $PT(n)$, and dividing by $(2n - 6)$ to obtain $\mu(n)$, we obtain

$$(4) \quad \mu(n) = (p(n)^{-1} \sum_f f |PT(n, f)|) + (\mu(T_n) - \varepsilon(n))/(2n - 6)$$

with $0 < \varepsilon(n) < 2C_1$.

Now the trees at distance f from any $T_n \in PT(n, 0)$ are precisely those with f internal edges. Thus,

$$(5) \quad \mu(T_n) = p(n)^{-1} \sum_{f \geq 0} f |PT(n, f)|, \quad \mu(n) = (2\mu(T_n) - \varepsilon(n))/(2n - 6).$$

Result (a) now follows by Lemma 7.

The variance $\sigma^2(n)$ of the normalized distance is the average value of

$$((d(T, T'))/(2n - 6) - \mu(n))^2$$

over all pairs $T, T' \in PT(n)$. Thus,

$$(6) \quad \sigma^2(n) = \left(\sum_{T \in PT(n)} \sum_k k^2 P_k(T) \right) / p(n)(2n - 6)^2 - \mu^2(n) \\ = \left(\sum_{T \in PT(n)} D^2 P(T, x)|_{x=1} + \sum_{T \in PT(n)} DP(T, x)|_{x=1} \right) / p(n)(2n - 6)^2 - \mu^2(n).$$

From Theorem 1,

$$(7) \quad D^2 P(T, x)|_{x=1} = D^2 P(T/e, x)|_{x=1} + 2DP(T/e, x)|_{x=1} \\ - 4D((P(T_1, x))P(T_2, x))|_{x=1} - 2P(T_1, 1)P(T_2, 1).$$

Now, $p(n)^{-1} D((P(T_1, x))P(T_2, x))|_{x=1} \leq 2(2n - 8)p(n_1)p(n_2)/p(n)$ (where $T_1 \in PT(n_1)$, $T_2 \in PT(n_2)$) since for $T' \in PT(m, f)$, $DP(T', x)|_{x=1}$ is clearly bounded above by $(m + f - 3)p(m) \leq (2m - 6)p(m)$.

Let $T \in PT(n, f)$. Dividing (7) by $p(n)$, we obtain $p(n)^{-1} D^2 P(T, x)|_{x=1} = p(n)^{-1} D^2 P(T/e, x)|_{x=1} + 2\mu(T/e) - \varepsilon(T)$, with $0 < \varepsilon(T) < \varepsilon(n) = O(1)$ by (2). Reducing T to $T_n \in PT(n, 0)$, and using (3), we have

$$(8) \quad p(n)^{-1} D^2 P(T, x)|_{x=1} = p(n)^{-1} D^2 P(T_n, x)|_{x=1} \\ + 2 \left(\sum_{0 \leq f' \leq f-1} (f' + \mu(T_n)) - \varepsilon_1(T) \right)$$

with $0 < \varepsilon_1(T) < \varepsilon_1(n) = O(n)$.

Now, $D^2 P(T_n, x)|_{x=1} = \sum_f f(f - 1) |PT(n, f)|$, while the second term in (8) is $f(f - 1) + 2f\mu(T_n)$. Averaging (8) over $PT(n)$, we obtain

$$2p(n)^{-1} \sum_f f(f - 1) |PT(n, f)| + 2\mu^2(T_n) - \varepsilon_2(n) \quad \text{with } \varepsilon_2(n) = O(n).$$

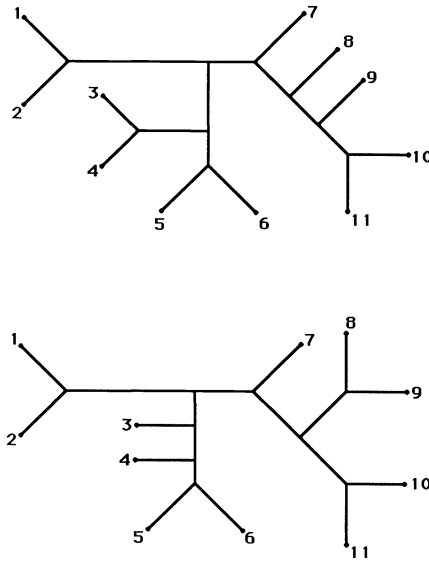


FIG. 3

Substituting this into (6), and noting that $\sum_{T \in PT(n)} DP(T, x)|_{x=1}/p(n)(2n-6)^2 = \mu(n)/(2n-6)$, and $2\mu^2(T_n)/(2n-6)^2 - 0.5\mu(n) = O(n^{-1})$ (from (5)), we have $\sigma^2(n) = 2(\sum_f f(f-1)|PT(n, f)|)/(2n-6)^2 - 0.5\mu^2(n) + \varepsilon_3(n)$, with $\varepsilon_3(n) = O(n^{-1})$. The result now follows from part (a) and Lemma 7.

Remark 3. By Chebyshev's inequality, Theorem 5 shows that for any number k , the probability two trees in $PT(n)$ have k or less equivalent edges tends to zero as n becomes large. This is in contrast to the binary case for which most trees are a maximal distance apart.

For $T \in BPT(n)$ an analysis similar to the first part of Theorem 5 can be applied to $v(T)$ —the expected distance between T and trees in $BPT(n)$. Differentiating $Q(T, x)$ and setting $x = 1$, we obtain $v(T) = v(T/e) + 1 - 2b(n_1)b(n_2)/b(n)$. By induction, $v(T) = v(T_n) + (n-3) - 2r(1, T)/b(n)$, for $T_n \in PT(n, 0)$.

Now $Q(T_n, x) = b(n)x^{n-3}$ so that $v(T_n) = n-3$. Thus, $v(T)/(2n-6) = 1 - 2r(1, T)/b(n)(2n-6)$. As in Example 3 $r(1, T) \leq (n-3)b(n-1)$, giving $v(T)/(2n-6) \geq 1 - 1/(2n-5)$. In particular, $v(T)/(2n-6) \rightarrow 1$ as $n \rightarrow \infty$.

Remark 4. The expected distance of binary trees from a given binary tree $T \in BPT(n)$ does not characterize T (up to topological equivalence) in $BPT(n)$. Indeed a counting argument shows that for all integers $k \geq 1$, there exists a positive integer n and a set $S \subset BPT(n)$ of k topologically distinct trees, on which $v(T)$ is constant. For $k = 2$, this is realized for $n = 11$, with the two trees given in Fig. 3. Although $v(T)$ does not characterize the topology of T , it is not known whether or not $Q(T, x)$ (equivalently, $R(T, x)$ by Lemma 2) does.

REFERENCES

- [1] I. ANDERSON, *A First Course in Combinatorial Mathematics*, Clarendon Press, Oxford, 1974.
- [2] M. BOURQUE, *Arbres de Steiner et reseaux dont varie l'emplacement de certains sommets*, Ph.D. thesis, Department d'Informatique et de Recherche Operationnelle, Universite de Montreal, Montreal, Quebec, Canada, 1978.

- [3] W. H. E. DAY, *Distribution of distances between pairs of classifications*, in Numerical Taxonomy, J. Felsenstein, ed., Springer-Verlag, Berlin, Heidelberg, 1983, pp. 127–131.
- [4] ———, *Optimal algorithms for comparing trees with labeled leaves*, J. Classification, 2 (1985), pp. 7–28.
- [5] J. FELSENSTEIN, *The number of evolutionary trees*, Syst. Zool., 27 (1978), pp. 27–33.
- [6] I. P. GOULDEN AND D. M. JACKSON, *Combinatorial Enumeration*, John Wiley, New York, 1983.
- [7] M. D. HENDY AND D. PENNY, *Branch and bound algorithms to determine minimal evolutionary trees*, Math. Biosci., 59 (1982), pp. 277–290.
- [8] M. D. HENDY, C. H. C. LITTLE, AND D. PENNY, *Comparing trees with pendant vertices labelled*, SIAM J. Appl. Math., 44 (1984), pp. 1054–1065.
- [9] E. MARCZEWSKI AND H. STEINHAUS, *On a certain distance of sets and the corresponding distance of functions*, Colloq. Math., 6 (1958), pp. 319–327.
- [10] D. PENNY, L. R. FOULDS, AND M. D. HENDY, *Testing the theory of evolution by comparing phylogenetic trees constructed from five different protein sequences*, Nature, 297 (1982), pp. 197–200.
- [11] D. PENNY AND M. D. HENDY, *The use of tree comparison metrics*, Syst. Zool., 34 (1985), pp. 75–82.
- [12] F. RESTLE, *A metric and an ordering on sets*, Psychometrika, 24 (1959), pp. 207–220.
- [13] D. ROBINSON AND L. R. FOULDS, *Comparison of phylogenetic trees*, Math. Biosci., 53 (1981), pp. 131–147.
- [14] ———, *Enumeration of phylogenetic trees without points of degree two*, Ars Combin., 17A (1984), pp. 169–183.

THE COMPLEXITY OF METRIC REALIZATION*

PETER WINKLER†

Abstract. It is shown that the problem of realizing a metric by a graph or network with minimum total edge-length is, depending on the version, NP-hard or NP-complete.

In particular, Discrete Metric Realization (DMR) is NP-complete "in the strong sense," where DMR is defined as follows:

INSTANCE. An n -by- n integer-entry distance matrix $D = (d_{ij})$ and a positive integer k ;

QUESTION. Is there a graph $G = \langle V, E \rangle$ with distinguished vertices $v_1, v_2, \dots, v_n$ that realize D , i.e., the number of edges in a shortest path between v_i and v_j is exactly d_{ij} for each $1 \leq i \leq j \leq n$, and such that the total number of edges of G is at most k ?

Key words. metric, graph, realization, complexity

AMS(MOS) subject classifications. 68R10, 54E35

1. Introduction. The problem of realizing a metric by a graph or network of minimal total length seems to have been around since the early 1960s. Basically, a distance matrix is given and the object is to construct a graph (hopefully, in many cases, a tree) with edge-lengths, containing distinguished vertices that realize the distance matrix via the ordinary path metric.

In many of the applications the distance matrix arises from a similarity measure. Here we hope that the realizing graph or tree will suggest an underlying structure; auxiliary vertices might even predict the existence of additional entities. In [6] and [17], for example, phylogenetic trees are used to realize genetic similarities in the study of evolution. In [4], weighted trees are obtained in the study of psychological similarities between animal terms and in the analysis of memory for sentences.

In [3] and [5] the object is quite different: distance matrices are used to encode trees, based on the observation in [20] that trees are uniquely determined by distances between endpoints, and the realization problem arises of course in recovering the tree from the code. Realization by graphs arises in the synthesis of certain electrical circuits [10], and, lastly, in operations research [22].

Study of the metric realization problem itself began about the same time in Europe and the United States, and has resulted in the establishment of the following facts, due to Buneman, Hakimi, Yau, Imrich, Simoes-Pereira, Zamfirescu, Stotskii, and Patrinos, distributed with some duplication among [2], [11]–[13], [16], [18], [19], and [21]. (For precise definitions see § 2 below.)

(1) Optimal realizations (minimum total edge-length) always exist, even when non-integer edge-lengths are permitted.

(2) A distance matrix $D = (d_{ij})$ is realizable by a *tree* (with edge-lengths) if and only if every four-tuple (p, q, r, s) of indices satisfies the "four-point condition": namely, the two largest of the three sums $d_{p,q} + d_{r,s}$, $d_{p,r} + d_{q,s}$, and $d_{p,s} + d_{q,r}$ must be equal.

(3) If a tree realization exists then there is only one such realization; it is optimal among all realizations and obtainable in polynomial time.

(4) Several reduction methods are available for trying to construct an optimal realization in the general case, but none works all the time.

* Received by the editors January 11, 1988; accepted for publication (in revised form) April 1, 1988. This research was supported by Office of Naval Research grant N00014-85-K-0769.

† Department of Mathematics and Computer Science, Emory University, Atlanta, Georgia 30322.

In light of the last of these facts it seems likely that the general metric realization problem is intractable, that is, unsolvable in polynomial time (assuming $P \neq NP$); and we will show that this is indeed the case. The discrete form of the metric realization problem does, as it turns out, lie in NP and thus is added to the list of NP-complete problems involving metrics, which includes minimal connection of points in the plane [8] and in a given graph [14].

2. Notation and preliminaries. By a *graph* $G = \langle V, E \rangle$ we mean a finite set V of vertices together with a collection E of two-element subsets (edges) of V . Loops and multiple edges are irrelevant for our purposes and will not be considered. G induces a metric (sometimes called the “path metric” or “geodesic metric”) on V by defining $d_G(x, y)$ to be the least number of edges in a path from vertex x to vertex y , with $d_G(x, y) = \infty$ if no path exists.

We will reserve the term *network* for a graph with edge-lengths (sometimes called “weights” or “costs”), i.e., a triple $H = \langle V, E, \lambda \rangle$ consisting of a graph and a function $\lambda: E \rightarrow \mathbb{R}^+$. The metric in H is then given by $d_H(x, y)$ = the minimum over all paths P from x to y of the sum of the lengths of the edges in P . A *discrete* network is one in which all edge-lengths are integers, and of course a graph can be regarded as a special case with all edges of unit length. The *total length* $L(H)$ of a network is the sum of the lengths of all the edges.

A *distance matrix* is a square symmetric matrix $D = (d_{i,j})$ with zero diagonal and positive reals elsewhere, satisfying $d_{i,j} + d_{j,k} \leq d_{i,k}$ for all indices i, j, k ; in other words, D displays the distance function of some finite metric space. D is said to be *realized* by a network $H = \langle V, E, \lambda \rangle$ if V contains distinguished vertices v_i corresponding to the indices of D such that $d_H(v_i, v_j) = d_{i,j}$ for each i and j . H is an *optimum* realization of D if in addition, it has minimum total length among all realizations and has no auxiliary vertices of degree two (these can always be suppressed).

An example of a distance matrix and realization is given in Fig. 1; the realization is optimal both among discrete networks and general networks, but, as can be seen by rotation, is not unique.

It is trivial that every distance matrix has a realization since each pair of vertices v_i, v_j can be connected by an edge of length $d_{i,j}$. Further, repeatedly deleting all redundant edges, that is, edges $\{x, z\}$ for which there is a vertex y satisfying $d(x, z) = d(x, y) + d(y, z)$, results in a realization that is optimal among those having no auxiliary vertices. This easy polynomial-time algorithm works even in the case of quasi-metrics and directed networks [1]. However, auxiliary vertices can reduce the total length of a realization by an arbitrarily large factor.

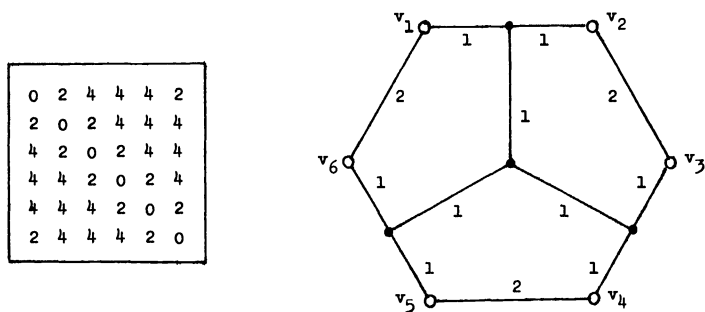


FIG. 1. $\circ$ indicates distinguished vertices; $\bullet$ indicates auxiliary vertices.

In general our object is to determine the complexity of finding an optimal realization for a given metric; as usual the problem is best phrased in terms of finding a realization whose total length is not more than some given number. The following is the first version of the problem, in the format of Garey and Johnson [9].

METRIC REALIZATION (MR):

INSTANCE. An n -by- n real distance matrix D and a positive real number r .

QUESTION. Is there a network $H = \langle V, E, \lambda \rangle$ containing n distinguished vertices realizing D , with $L(H) \leq r$?

Since MR involves input and output of real numbers it makes sense (as in [8]) to give instead an integer version, which, if necessary, could be used as an approximating tool.

DISCRETE METRIC REALIZATION (DMR):

INSTANCE. An n -by- n integer distance matrix D and a positive integer k .

QUESTION. Is there a graph $G = \langle V, E \rangle$ such that V contains n distinguished vertices realizing D , with $L(G) \leq k$?

We could equally well have asked for a discrete network in the above question (a point that has some later significance) since auxiliary vertices of degree two can be added or deleted to pass between a discrete network and its equivalent graph.

3. The complexity results. A problem is “in NP” if it can be solved by a nondeterministic Turing machine in time bounded by a polynomial in the length of the input; it is “NP-hard” (usage varies somewhat here) if it is at least as hard as any problem in NP. A problem is “NP-complete” if it is in NP, but the existence of a *deterministic* polynomial-time solution to it would imply that $P = NP$, i.e., *all* problems in NP have deterministic polynomial solutions—which seems unlikely at this time. Thus, being NP-hard or NP-complete is generally taken as substantial evidence of intractability.

If a problem requires integer input then the “length” of the input is measured according to the *logarithms* of the integers; thus, the length of an instance of DMR might be taken to be $n^2 + \sum \{ \log d_{i,j} : i \neq j \} + \log k$. Sometimes a problem is NP-complete using this treatment for the length of integers, but if the magnitude of the integer is taken as its length, or if the integers appearing in an instance are bounded in advance by a polynomial in the input length, then the problem becomes tractable. An example (PARTITION) and discussion of this point can be found in Garey and Johnson [9]. If the problem remains NP-complete even when a polynomial bound is imposed on the integer inputs, then the problem is said to be NP-complete “in the strong sense.”

Our object is to prove the following theorem.

THEOREM. METRIC REALIZATION is NP-hard and DISCRETE METRIC REALIZATION is NP-complete in the strong sense.

Proof. The strategy is first to show that DMR is in NP. Then we assume integer matrix input with entries bounded by a polynomial in n (namely the constant “4”) and show that the problem of finding a suitable realizing network is NP-hard, whether or not noninteger edge-lengths are permitted.

To put DMR in NP we need to show that if the answer to an instance is YES then there is a “certificate” that proves it. The certificate must be polynomially describable (that is, its length is bounded by a polynomial in the input length) and polynomially verifiable (it can be checked in time bounded by a polynomial in the input length).

The certificate for an instance of DMR will *not* be a graph but a discrete network $H = \langle V, E, \lambda \rangle$. The reason for avoiding the graph will be apparent from the following

example: let the instance be a two-by-two matrix D with $d_{1,2} = d_{2,1} = k =$ some huge integer q . The network certificate will then have a single edge of length q , but the graph realization is a path with $q + 1$ vertices—an object that is exponential in size relative to $\log q$.

On the other hand, we have not established that there is necessarily a *network* certificate whose number of vertices is polynomial in the input length. Imrich, Simões-Pereira, and Zamfirescu [12] derive a bound of $2^{n(n-1)/2+1}$ for the number of vertices in an optimum realization of an n -by- n distance matrix; fortunately, their remark that this bound is “by no means best possible” turns out to be verifiable.

LEMMA 1. *Let D be an n -by- n distance matrix (real entries) with a realization $H = \langle V, E, \lambda \rangle$ that is optimal, either among all networks, or among all discrete networks. Then $|V| \leq n^4/2$.*

Proof of Lemma 1. For every pair x, y of distinguished vertices of H , let $P(x, y)$ be a fixed shortest path from x to y . Then every edge (and thus every vertex) of H lies on $P(x, y)$ for at least one pair x, y , else that edge could be deleted, contradicting the optimality of H . Since there are only $n(n-1)/2$ (unordered) pairs of distinguished vertices, it will suffice to show that no $P(x, y)$ can have more than, say, n^2 vertices.

To see this fix x and y and for each pair u, v of distinguished vertices such that $P(u, v)$ intersects $P(x, y)$, let $x(u, v)$ be the closest vertex to x among those common to $P(u, v)$ and $P(x, y)$, and $y(u, v)$ the closest to y . Since $P(x, y)$ has more than $2[(n(n-1)/2)]$ vertices there is one, say z , that is not $x(u, v)$ or $y(u, v)$ for any u and v . Since z is an auxiliary vertex (else $z = x(z, y)$) it has degree greater than two; let e be an edge incident to z that does not lie on the path $P(x, y)$. We claim e can be deleted without disturbing distances among distinguished vertices, again contradicting the optimality of H .

For, if e lies on $P(u, v)$, then the vertex z lies strictly between $x(u, v)$ and $y(u, v)$ on the path $P(x, y)$ (see Fig. 2). The segment of $P(u, v)$ between $x(u, v)$ and $y(u, v)$ can be replaced by the corresponding segment of $P(x, y)$, which is of the same length (since both $P(u, v)$ and $P(x, y)$ are shortest paths) and does not contain the edge e . This completes the proof of the lemma. $\square$

Of course, $n^4/2$ is not best possible either (we do not know what is), but at least we can conclude that if there is any network realizing the input matrix whose total length satisfies the required bound, then there is one whose vertex and edge sets are describable polynomially in n . Since none of the edge-lengths will exceed the largest matrix entry, the edge-length function λ will be describable polynomially in n and $\sum \log d_{i,j}$. We note also that if the given bound k is greater than half the sum of the matrix entries, then the required network always exists and no certificate is needed; hence a large value for k causes no difficulties.

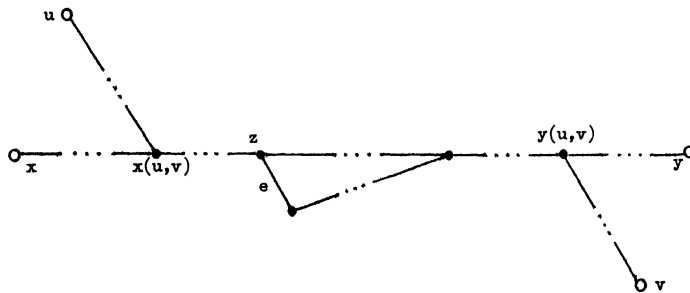


FIG. 2

To verify the certificate we need only sum the edge-lengths and compare with k —this is easy enough—and compute the distance matrix for the distinguished vertices of H . Fortunately it is not necessary to compute the length of every path to do this; any good shortest-path algorithm, e.g., Floyd's [7], will find the distance matrix in polynomial time, and it can then be matched against the input matrix D . We conclude that DMR is indeed in NP.

To show that MR and DMR are NP-hard we employ the usual technique of reducing some known NP-complete problem to each of them. The problem of choice is PARTITION INTO CLIQUES (PIC) and is described as follows in [9]:

INSTANCE. A graph G with vertex set U , and a positive integer $q \leq |U|$.

QUESTION. Can the vertices of G be partitioned into $p \leq q$ disjoint sets $U_1, U_2, \dots, U_p$ such that, for $1 \leq i \leq p$, the subgraph induced by U_i is a complete graph?

PARTITION INTO CLIQUES is one of the original problems shown by Karp [15] to be NP-complete in 1972. Note that it is the complement of the familiar graph coloring problem. We will demonstrate that if MR (with integer input) or DMR can be solved in polynomial time when the matrix entries are bounded by n^2 , then PIC is polynomial.

Accordingly, let an instance (G, q) of PIC be given, with, say, vertex set $U = \{u_1, u_2, \dots, u_m\}$. Set $n = m + m^2$ and construct the n -by- n distance matrix $D = (d_{i,j})$ as follows:

$d_{i,i} = 0$;
 $d_{i,j} = 2$ if $\{u_i, u_j\}$ is an edge of G ,
 or if exactly one of $\{i, j\}$ is $> m$; and
 $d_{i,j} = 4$ in all other cases.

In any realization of D , we identify U with the set of distinguished vertices corresponding to indices $1, \dots, m$ of the matrix and W with the set of m^2 distinguished vertices corresponding to indices $m + 1, \dots, n$. One realization of D can be obtained by beginning with G , adding W to the vertex set together with an edge from each member of W to each member of U , and setting the length of every edge equal to two.

The reason for adding m^2 new indices, representing "cones" over the graph G , is that we have almost no control over optimal realization of distances within U . Hence, our approach is to overwhelm this issue with the question of realizing distances from U to the cone vertices.

Let $k = m^2(m + q + 1) - 1$. We claim the following:

G has a partition into $p \leq q$ cliques **iff**
 D can be realized by a network of total length $\leq k$ **iff**
 D can be realized by a *discrete* network of total length $\leq k$.

We begin by letting $H = \langle V, E, \lambda \rangle$ be any realization of D , with λ real valued, and assuming that G does *not* have a partition into at most q cliques; we will show $L(H) \geq m^2(m + q + 1)$.

In any network, if X and Y are subsets of the vertices, let us denote by $L(X, Y)$ the sum of the lengths of all edges that lie on shortest paths from any vertex in X to any vertex in Y . Thus $L(V, V) = L(H)$ for the above network, but in fact we will show that $L(U, W)$ is already at least $m^2(m + q + 1)$.

Let w and w' be distinct elements of W , and let u_i and u_j be points in U which are *not* necessarily distinct. We claim that *no* edge e can be simultaneously on a shortest path from w to u_i and on a shortest path from w' to u_j . For, otherwise, let z be the

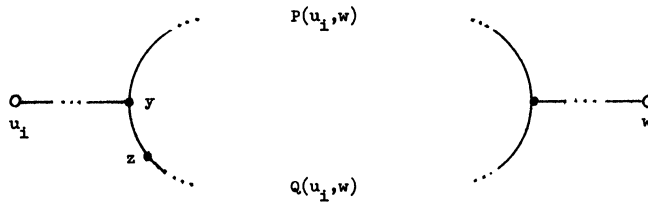


FIG. 3

endpoint of e nearest to w ; then $d_H(w, z) < 2$ and $d_H(z, w') \leq 2$, contradicting the fact that $d_H(w, w') = 4$.

It follows that each edge of H is counted in $L(U, \{w\})$ for at most one w ; hence,

$$L(U, W) = \sum_{w \in W} L(U, \{w\}).$$

Thus, we now need only show that $L(U, \{w\}) \geq m + q + 1$ for arbitrary $w \in W$.

To that end let w be an arbitrary member of W and let $H' = \langle V', E', \lambda' \rangle$ be a realization of D that minimizes $L(U, \{w\})$ (this exists via the argument of Theorem 2.2 in [12]). We need the following observation.

LEMMA 2. Every $u_i \in U$ has a unique shortest path $P(u_i, w)$ to w in H' .

Proof of Lemma 2. If not let $Q(u_i, w)$ be a second shortest path; let z be the closest vertex to u and $e = \{y, z\}$ the closest edge to u_i lying on $Q(u_i, w)$ but not on $P(u_i, w)$. (See Fig. 3.) We claim that removal of the edge e cannot change the distance from any $u_j \in U$ to w .

To see this, note if the edge e lies on a shortest path from some $u_j \in U$, then the segment of that path from y to w can be replaced by the segment from y to w in $P(u_i, w)$ to achieve a path of equal length not containing e . Thus $d_{H'}(u_j, w)$ is undisturbed, as claimed.

Now any increase the removal of e might cause among other distances can be repaired by direct addition of edges to H' ; thus the minimality of $L(U, \{w\})$ is contradicted, and the lemma is proved. $\square$

It follows from Lemma 2 that if $1 \leq i \leq j \leq m$ and $\alpha_{i,j} = L(P(u_i, w) \cap P(u_j, w))$ then the union of $P(u_i, w)$ and $P(u_j, w)$ is as pictured in Fig. 4.

If $\alpha_{i,j}$ and $\alpha_{j,h}$ are both nonzero then $P(u_i, w)$, $P(u_j, w)$ and $P(u_h, w)$ all contain a common edge incident to w ; hence the binary relation θ on U given by

$$u_i \theta u_j \text{ iff } \alpha_{i,j} > 0$$

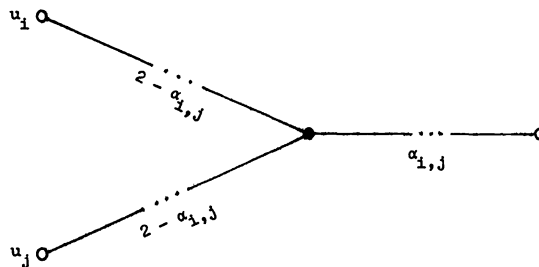


FIG. 4

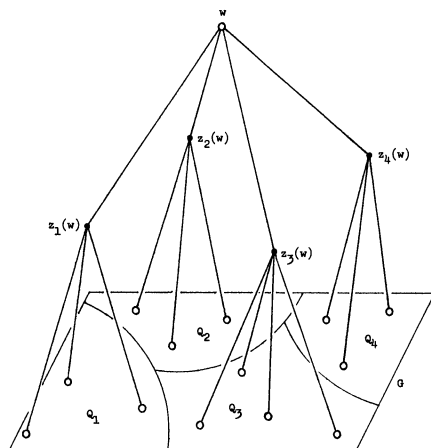


FIG. 5

is transitive, as well as trivially reflexive and symmetric. Let $U_1, U_2, \dots, U_s$ be the equivalence classes of θ .

If $\{u_i, u_j\}$ is not an edge of G then $d_{i,j} = 4$, and hence $\alpha_{i,j} = 0$; thus each U_i induces a clique in G , forcing $s > q$ by assumption. Even when $\{u_i, u_j\}$ is an edge, we have $\alpha_{i,j} \leq 1$ since $d_{i,j} = 2 \leq 2(2 - \alpha_{i,j})$ via Fig. 4. Thus $L(U_i, \{w\}) \geq 1 + |U_i|$ and

$$L(U, \{w\}) = \sum_{i=1}^s L(U_i, \{w\}) \geq s + m > q + m$$

as promised.

It remains only to show that if G does have a partition $U_1, U_2, \dots, U_p$ into $p \leq q$ cliques, then there is a (discrete) network $H = \langle V, E, \lambda \rangle$ realizing D and satisfying $L(H) \leq m^2(m + q + 1) - 1$; H is constructed as follows. For each $w \in W$ and each clique U_i , introduce an auxiliary vertex $z_i(w)$ connected by unit-length edges to w and to each vertex in U_i . The total length of these edges will then be $m^2(m + p)$. To these we add all of the edges of the original graph G , each with length two. The network H is illustrated in Fig. 5, but only one vertex $w \in W$, with its associated edges, is shown.

The resulting network will not generally be optimal, but it is easily seen to realize D and

$$L(H) \leq m^2(m + p) + 2[m(m - 1)/2] < m^2(m + q + 1),$$

completing the proof of the Theorem. $\square$

Acknowledgment. This paper is dedicated to the memory of Irene Mallison, librarian, without whom the literature research for this paper (and many others) would have been far more difficult and less complete.

REFERENCES

- [1] K. V. S. BHAT AND S. SAADAT, *Optimization realization of a distance matrix*, Internat. J. Comput. Math. Section B, 8 (1980), pp. 77–83.
- [2] P. BUNEMAN, *A note on the metric properties of trees*, J. Comb. Theory Ser. B, 17 (1974), pp. 48–50.

- [3] S. CHAIKEN, A. K. DEWDNEY, AND P. S. SLATER, *An optimal diagonal tree code*, SIAM J. Algebraic Discrete Methods, 4 (1983), pp. 42–49.
- [4] J. P. CUNNINGHAM, *Free trees and bidirectional trees as representations of psychological distance*, J. Math. Psych., 17 (1978), pp. 165–188.
- [5] A. K. DEWDNEY, *Diagonal tree codes*, Inform. and Control, 40 (1979), pp. 234–239.
- [6] M. EIGEN, W. GARDINER, P. SCHUSTER, AND R. WINKLER-OSWATITSCH, *The origin of genetic information*, Sci. Amer., April 1981, pp. 88–118.
- [7] R. W. FLOYD, *Algorithm 97: shortest path*, Comm. ACM, 5 (1962), p. 345.
- [8] M. R. GAREY, R. L. GRAHAM, AND D. S. JOHNSON, *The complexity of computing Steiner minimal trees*, SIAM J. Appl. Math., 32 (1977), pp. 835–859.
- [9] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, New York 1979.
- [10] S. L. HAKIMI AND S. S. YAU, *Distance matrix and the synthesis of N-port resistance networks*, Network Theory Group, Technical Report #5, Northwestern University, Evanston, IL, 1963.
- [11] ———, *Distance matrix of a graph and its realizability*, Quart. Appl. Math., 22 (1964–65), pp. 305–317.
- [12] W. IMRICH, J. M. S. SIMÕES-PEREIRA, AND C. M. ZAMFIRESCU, *On optimal embeddings of metrics in graphs*, J. Comb. Theory Ser. B, 36 (1984), pp. 1–15.
- [13] W. IMRICH AND E. STOTZKIL, *Optimal imbeddings of metrics in graphs*, Sibirsk. Mat. Zh., 13 (1972), pp. 558–565.
- [14] D. S. JOHNSON, J. K. LENSTRA, AND A. H. G. RINNOOY KAN, *The complexity of the network design problem*, Networks, 8 (1978), pp. 279–285.
- [15] R. M. KARP, *Reducibility among combinatorial problems*, in Complexity of Computer Computations, R. E. Miller and J. W. Thatcher, eds., Plenum Press, New York (1972), pp. 85–103.
- [16] 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.
- [17] D. PENNY, L. R. FOULDS, AND M. D. HENDY, *Testing the theory of evolution by comparing phylogenetic trees constructed from five different protein sequences*, Nature, 297 (1982), pp. 197–200.
- [18] J. M. S. SIMÕES-PEREIRA, *A note on the tree realizability of a distance matrix*, J. Comb. Theory, 6 (1969), pp. 303–310.
- [19] J. M. S. SIMÕES-PEREIRA AND C. M. ZAMFIRESCU, *Submatrices of nontree-realizable distance matrices*, Linear Algebra Appl., 44 (1982), pp. 1–17.
- [20] E. A. SMOLENSKII, *A method for linear recoding of graphs*, USSR Comput. Math. and Math. Phys., 2 (1963), pp. 396–397. In Russian Zh. Vychisl. Mat. i Mat. Fiz., 2 (1962), pp. 371–372.
- [21] E. D. STOTZKIL, *Imbedding of finite metrics in graphs*, Sibirsk. Mat. Zh., 5 (1964), pp. 1203–1206.
- [22] B. C. TANSEL, R. L. FRANCIS, T. J. LOWE, AND M. L. CHEN, *Duality and distance constraints for the non-linear p-center problem and covering problem on a tree network*, Oper. Res., 30 (1982), pp. 725–744.