COMBINATORICA 1 (1981) 49—54

FORESTS AND SCORE VECTORS

by

Daniel J. KLEITMAN*, Kenneth J. WINSTON

Department of Mathematics
Massachusetts Institute of Technology
Cambridge, Mass., 02139, USA

Received 11 June 1979

The *score vector* of a labeled digraph is the vector of out-degrees of its vertices. Let G be a finite labeled undirected graph without loops, and let $\sigma(G)$ be the set of distinct score vectors arising from all possible orientations of G. Let $\varphi(G)$ be the set of subgraphs of G which are forests of labeled trees. We display a bijection between $\sigma(G)$ and $\varphi(G)$.

1. Introduction

All graphs treated herein will be finite, labeled, and without loops. The set $[n]=\{1, 2, \dots, n\}$ will be used as the vertex set of all graphs with n vertices. We will use the term *edges* for undirected graphs and the term *arcs* for digraphs. We allow a finite number of multiple arcs and edges.

If G is an undirected graph and i and j are vertices of G, then the edges connecting i and j will be labelled $(i,j)^1$, $(i,j)^2$, ... We will write $(i,j)\in G$ to mean that there is at least one edge between i and j in G, and $(i,j)^0\in G$ to mean that i and j are non-adjacent. An undirected graph F is a *forest* if there is no set of distinct vertices $p_1, \dots, p_m$ $(m\geqq 3)$ of F with $(p_1, p_2)\in F$, $(p_2, p_3)\in F$, ..., $(p_{m-1}, p_m)\in F$, $(p_m, p_1)\in F$, and F contains no multiple edges. F is a *subforest* of an undirected graph G if F is a forest on the vertices of G and if the edge set of F is a subset of the edge set of G. Let $\varphi(G)$ be the set of distinct subforests of the undirected graph G.

We will refer to the vertices of a digraph as *players*. If i and j are two players of a digraph D, we write $\langle i,j\rangle^k\in D$ if there are at least k arcs from i to j in D. When there is only a single arc between i and j, we say *i beats j* if the arc is directed from i to j. A vector $V=\langle u_1, \dots, u_n\rangle$ is the *score vector* of D if the out-degree of i in D is v_i for i from 1 to n. Conversely, any digraph whose score vector is V is called a *realization* of V. An *orientation* of an undirected graph G is a digraph on the vertices of G in which the edges of G have each been assigned directions. Let $\sigma(G)$ be the set of distinct score vectors that arise from all possible orientations of G.

---

* Supported in part by ONR Contract N00014-76-C-0366.
AMS subject classification (1980): 05 C 20; 05 C 05, 05 C 30

When $G=K_n$, the complete graph (without multiple edges), an orientation of G is called a *tournament*. Let V_n be the number of tournament score n-vectors; that is, V_n is the cardinality of $\sigma(K_n)$. Similarly, let F_n be the number of forests of labeled trees on $[n]$; that is, F_n is the cardinality of $\varphi(K_n)$.

R. Stanley computed the sequence $V_1=1$, $V_2=2$, $V_3=7$, $V_4=38$, $V_5=291, \ldots$ and looked it up in [5]. Hence noticed that this sequence seemed to be the same as sequence 714 of [5], which was $F_1, F_2, \ldots$. He (and Zaslavsky, [6]) showed that the cardinality of $\varphi(G)$ equaled the cardinality of $\sigma(G)$ for undirected graphs G without multiple edges. However, his proof did not produce an explicit bijection from $\varphi(G)$ to $\sigma(G)$.

In the next section, we display such a bijection.

2. The Bijection

We first display the map I: $\varphi(G)\mapsto\sigma(G)$ in a form which is computationally simple. We do this for the case in which G has no multiple edges.

We construct from a given subforest F of G an orientation $D(F)$ of G. The score vector of $D(F)$ will be the image $I(F)$ under our map. Each tree of F will be identified by its lowest numbered vertex, which we will call its *index*. Let x and y be two different players such that the edge (x, y) is in G. We determine which player beats the other in $D(F)$ via the rules in the following paragraph:

If x and y are in different trees of F, then if the index of x's tree is i and the index of y's tree is j, x beats y if and only if $i>j$. Otherwise, x and y are in the same tree of index i of F. Suppose that x lies on the path between i and y (this includes $i=x$). Then if z is the vertex which is in this path and adjacent to x on the y side, x beats y if and only if $z\leqq y$ (this includes $z=y$). If y lies on the path between i and x (including $i=y$), then the obvious analog holds. If neither x nor y lies on the path from i to the other, there must be a vertex w (possibly with $w=i$) which lies on both the path from i to x and the path from i to y, and which is the vertex furthest from i with this property. Let p be the vertex adjacent to w on the x side, and let q be the vertex adjacent to w on the y side. Then x beats y if and only if $p>q$.

For example, suppose $G=K_6$ and F has edges $(1, 2)$, $(1, 6)$, and $(3, 5)$ (4 isolated). Then in $D(F)$: 1 beats 2, 6; 2 beats no one; 3 beats 1, 2, 5, 6; 4 beats everyone; 5 beats 1, 2, 6; 6 beats 2. Hence, the score vector constructed by these rules is $I(F)=\langle 2, 0, 4, 5, 3, 1\rangle$.

Clearly these rules allow us to create a well-defined orientation $D(F)$ of G and thus a score vector $I(F)$ when G has no multiple edges. Thus I is a map from $\varphi(G)$ into $\sigma(G)$ in this case. It remains for us to generalize the map to allow multiple edges and to show that I is one-to-one and onto.

Theorem. *The map I: $\varphi(G)\mapsto\sigma(G)$ is a bijection.*

Proof. We will proceed by reformulating I as a kind of depth-first search (cf. [1] for a similar use of depth-first search). The form in which we give I below applies as well when G has multiple edges. This will allow us to display an inverse map M: $\sigma(G)\mapsto\varphi(G)$ such that given a subforest F of G, $M(I(F))=F$; and given a score vector $V\in\sigma(G)$, $I(M(V))=V$. These properties show that these maps are one-to-one and onto.

ALGORITHM 1: MAP I FROM $\varphi(G)$ TO $\sigma(G)$

Let $F \in \varphi(G)$ be given. Let U be the edge set of G, and let X be the set of all orientations of G. Perform the following steps:

STEP 1: *If U is empty, stop. Otherwise, set both p and i to the lowest numbered index of F for which there exists a vertex r with $(p, r) \in U$. Go to step 2.*

STEP 2: *Is there a vertex r with $(p, r) \in U$? If so, let q be the lowest numbered vertex of F with $(p, q) \in U$. Go to step 3.*

Otherwise, if $p=i$, go to step 1.

Otherwise, let x be the vertex adjacent to p in the path from i to p. Reset p to x and repeat this step 2.

STEP 3: *Remove all (p, q) edges from U. Let k be such that $(p, q)^k \in F$. If $k=0$, remove from X all orientations containing a $\langle p, q\rangle$ arc. Go to step 2.*

Otherwise, remove from X all orientations D in which k is not the largest number such that $\langle p, q\rangle^k \in D$. If $p=i$, reset p to q and go to step 2.

Otherwise, for all vertices x on the path from i to p in F and for which there is an (x, p) edge in U, remove all (x, p) edges from U, and remove all orientations containing a $\langle p, x\rangle$ arc from X. Then reset p to q and go to step 2.

It is not hard to see that at the termination of Algorithm 1, there is exactly one orientation remaining in X. Note that this sole remaining orientation $D(F)$ is the one produced by the algorithm given at the beginning of this section when G has no multiple edges.

We now give the algorithm that takes a score vector V and produces a forest $M(V)$. We will use the properties of the set of realizations of V to construct the forest. Roughly speaking, we look at realizations of the vector which have certain arcs directed in prescribed ways. We put edges in the forest when these realizations agree on the directions of certain other arcs.

ALGORITHM 2: MAP M FROM $\sigma(G)$ TO $\varphi(G)$

Let $V \in \sigma(G)$ be given. Let R be the set of all realizations of V which are orientations of G, and let U be the edge set of G. We build up the edge set of a forest F by performing the following steps:

STEP 1: *If U is empty, stop. Otherwise, set both p and i to the lowest numbered vertex p such that p lies on some edge of U. Go to step 2.*

STEP 2: *Is there a vertex r with $(p, r) \in U$? If so, let q be the lowest numbered vertex of F with $(p, q) \in U$. Go to step 3.*

Otherwise, if $p=i$, go to step 1.

Otherwise, let x be the vertex adjacent to p in the path from i to p along the edges of F. Reset p to x and repeat this step 2.

STEP 3: *Remove all (p, q) edges from U. Let k be the largest number such that $\langle p, q\rangle^k \in D$ for all orientations $D \in R$. If $k=0$, remove from R all orientations containing a $\langle p, q\rangle$ arc. Go to step 2.*

Otherwise, put $\langle p, q\rangle^k \in F$. Remove from R all orientations having more than k copies of $\langle p, q\rangle$. For each edge $e \in U$ such that e is adjacent to q and such that the addition of e to F would complete a cycle in F, remove e from U. Then reset p to q and go to step 2.

The removal of the cycle-completing edges in step 3 insures that the set of edges in F when Algorithm 2 terminates form a forest.

Suppose e is an edge removed from U in step 3 of Algorithm 2 because it would complete a cycle. Number the vertices of the prospective cycle $x_1, x_2, \ldots, x_m$ (where $e=(x_1, x_m)$) so that edges were placed in F in the order: first $\langle x_1, x_2\rangle^{k_1}$; then $(x_2, x_3)^{k_2}$; and so forth up to $(x_{m-1}, x_m)^{k_{m-1}}$ last. Then at the time e is removed from U, all realizations of V remaining in R must contain arcs: $\langle x_1, x_2\rangle^{k_1}$, $\langle x_2, x_3\rangle^{k_2}$, and so forth up to $\langle x_{m-1}, x_m\rangle^{k_{m-1}}$.

Suppose that when e is removed there remains in R a realization D of V with $\langle x_m, x_1\rangle \in D$. Let D^* be the orientation identical to D except for the reversal of exactly one each of the arcs $\langle x_1, x_2\rangle$, $\langle x_2, x_3\rangle$, up to $\langle x_{m-1}, x_m\rangle$, as well as $\langle x_m, x_1\rangle$. Clearly D^* is also a realization of V. But since $\langle x_1, x_2\rangle^{k_1} \notin D^*$, the existence of such a D^* violates the conditions that must have been satisfied for $(x_1, x_2)^{k_1}$ to have been placed in F. Thus there is no such D. Hence, at the time e is removed from U, there is no $\langle x_m, x_1\rangle$ arc in any orientation then in R.

The argument in the previous paragraph indicates that when cycle-completing edges are removed from U, there is no choice in direction of the corresponding arcs in the remaining realizations of V. This matches precisely the way in which cycle-completing edges were handled in Algorithm 1. The claims that $I(M(V))=V$ and $M(I(F))=F$ then follow. This completes the proof of the theorem.

3. Remarks

A byproduct of the correspondence given in the theorem is the creation of a canonical realization (with respect to a given base graph G) of a score vector. The orientation $D(F)$ remaining in X at the completion of Algorithm 1 will be the same as the single realization remaining in R at the completion of Algorithm 2 applied to $I(F)$.

Computationally, the most difficult part of Algorithm 2 will in general be finding the set R of all orientations of G which are realizations of the argument vector V. We can avoid this as follows: Instead of using R, we begin the algorithm by forming any single orientation S of G which is a realization of V (there are standard procedures for finding such realizations, cf. [3]). We proceed as before in steps 1 and 2, but we use the following modified step 3:

STEP 3: *Let k be the largest number such that $\langle p, q\rangle^k \in S$. If $k=0$, remove all (p, q) edges from U and go to step 2.*

Otherwise, are there distinct players $q=p_1, p_2, \ldots, p_m=p$ such that $\langle p_i, p_{i+1}\rangle \in S$ and $(p_i, p_{i+1}) \in U$ for $1 \leq i \leq m-1$? If so, modify S by reversing the directions of each of the arcs in this cycle. Repeat this step 3.

Otherwise, put $(p, q)^k$ in F. For each edge $e \in U$ such that e is adjacent to q and such that the addition of e to F would complete a cycle in F, remove e from U. Then remove all copies of (p, q) from U, reset p to q, and go to step 2.

The equivalence of this version of step 3 and the previous version follows from the easily proved fact that any two realizations of the same score vector differ only by at most some series of reversals of cycles.

A (1-fold) tournament T is *reducible* if there is a proper subset P of its vertex set such that whenever $p \in P$ and $q \notin P$, $\langle p, q \rangle \in T$; the tournament is *irreducible* otherwise. Clearly the vertex set of any tournament T can be broken into disjoint sets, called *strong components,* in which the subtournament induced by T is irreducible. It is not hard to see that all realizations of a given tournament score vector V must have the same disjoint sets of vertices in their breakdown to irreducible subtournaments. It therefore makes sense to speak of a vector as reducible or irreducible.

It is clear from the above that the forest corresponding to a tournament score n-vector V will not be an n-tree if and only if V is reducible with the first entry not in the highest irreducible component. Thus there is a one-to-one correspondence between trees and tournament score vectors in which a given entry (say the first) is in the highest irreducible component.

4. Asymptotic Enumeration

Our interest in forests and score vectors was motivated by our desire to evaluate the number V_n of tournament score vectors. The bijection now allows us to use results on enumeration of forests on this problem.

We treat a generalized version of the problem: A graph G is *p-fold* if there are either exactly zero or exactly p edges between every two points. Let $K_{n;p}$ denote the complete p-fold graph. Then an orientation of $K_{n;p}$ represents a generalized tournament in which, rather than simply awarding a win to one of the players in each match, we divide up p points between them. Let $V_{n;p}$ be the cardinality of $\sigma(K_{n;p})$.

Let $G(n, p, k)$ be the number of forests in $\varphi(K_{n;p})$ in which there are exactly k connected components (trees). A 1-fold tree with n vertices has $n-1$ edges. Thus, a forest counted by $G(n, 1, k)$ has $n-k$ edges. Hence, $G(n, p, k) = G(n, 1, k)p^{n-k}$.

Rényi ([4], p. 74) showed that

$$G(n, 1, k) = \frac{n^{n-k}}{k!} \sum_{j=0}^{k} \binom{k}{j} (k+j) \left(-\frac{1}{2n}\right)^j \cdot \prod_{i=1}^{k+j-1} (n-i).$$

Note that the summands are zero when $k+j-1 \geqq n$. It is not hard to see that when $k < (n+1)/2$ this expression can be written as a polynomial of degree $n-2$ in n. Following Rényi, we collect terms in n to find that when $k < (n+1)/2$,

$$G(n, 1, k) = \frac{2^{-(k-1)}}{(k-1)!} (n^{n-2} + 5(k-1)n^{n-3} + \ldots).$$

Note that when $k \geqq (n+1)/2$, there are less than $(n-1)/2$ edges in a 1-fold forest. There are (compared to, say, n^{n-2}) a negligible number of labeled graphs with this few edges. We therefore take only the highest power of n in the previous equation, obtaining

$$V_{n;p} \approx p^{n-1} n^{n-2} \sum_{k=1}^{(n+1)/2} \frac{(2p)^{-(k-1)}}{(k-1)!} \approx \exp\left(\frac{1}{2p}\right) p^{n-1} n^{n-2}.$$

Here $A \approx B$ means the limit as n goes to infinity of A/B is 1. In particular, we have $V_n \approx \sqrt{e}n^{n-2}$.

A score vector $V = \langle v_1, \dots, v_n \rangle$ is a *score sequence* if $v_1 \geqq v_2 \geqq \dots \geqq v_n$. Let $S_{n;p}$ be the number of distinct p-fold tournament score sequences. The asymptotic evaluation of $S_{n;p}$ (and in particular of $S_n = S_{n;1}$) is quite difficult; see for example [2]. Note however that for fixed n, as p gets large the probability that a vector $V = \langle v_1, \dots, v_n \rangle \in \sigma(K_{n;p})$ has distinct components (that is, $v_i = v_j$ implies $i = j$) gets larger. Using $\approx$ with respect to p here, it therefore follows that

$$S_{n;p} \approx \frac{1}{n!} V_{n;p} \approx \frac{p^{n-1}n^{n-2}}{n!}.$$

Note that the remarks in the last section imply that if V is a 1-fold tournament score sequence then $M(V)$ is a tree: since player 1 has the highest score it must be in the highest irreducible component. In addition, note that if the sequence V is reducible, then the components below 1's component are on the "youngest branch" of $M(V)$. That is, if P is the component containing 1 and Q is the rest of the vertex set, then there is a path $1 = p_1, p_2, \dots, p_m$ such that p_i is the smallest vertex adjacent to p_{i-1} (except possibly for p_{i-2}) and such that all vertices of Q lie below p_m in $M(V)$. Further observations along these lines may produce a workable set of necessary and sufficient conditions for a tree to be in the image set of M applied to tournament score sequences and perhaps allow an asymptotic determination of S_n.

References

[1] I. Gessel and D. L. Wang, Depth-First Search as a Combinatorial Correspondence, *Journal of Combinatorial Theory (A)*, **26** (1979), 308—313.
[2] D. J. Kleitman, The Number of Tournament Score Sequences for a Large Number of Players, *Combinatorial Structures and Their Applications*, University of Calgary, Alberta, (1969), 209—213.
[3] D. J. Kleitman and D. L. Wang, Algorithms for Constructing Graphs and Digraphs with Given Valences and Factors, *Discrete Math.* **6** (1973), 79—88.
[4] A. Rényi, Some Remarks on the Theory of Trees, *MTA Matematikai Kutató Intézetének Közleményei (Publ. of Math. Inst. Hung.)* **4** (1959), 73—85.
[5] N. J. A. Sloane, *A Handbook of Integer Sequences*, Academic Press, 1973.
[6] R. Stanley, Decompositions of Rational Convex Polytopes, *Proceedings of Symposium on Combinatorial Mathematics and Optimal Design*, Colorado State University (June, 1978), to appear.