A combinatorial ranking problem

R. N. Burns and C. E. Haff

Abstract

An algorithm is presented for finding an nth-best spanning tree of an edge-weighted graph G. In sharp contrast to related ranking algorithms, the number of steps is a linear function of the parameter n. The results apply as well to ranking the bases of an abstract matroid.

§1. Introduction

All graphs dealt with in this paper are assumed to be connected, finite and undirected. (The reader is referred to [4] for any graph theoretical terms not defined in this paper.) Given a graph G with vertex set $V=V(G)$ and edge set $E=E(G)$, let $c=\langle c(e)\colon e\in E\rangle$ be a real vector which assigns *weights* $c(e)$ to the edges of G. A *spanning tree* T of G is a connected subgraph of G which contains no circuit and such that $V(T)=V(G)$. Let $\mathscr{T}=\mathscr{T}(G)$ denote the family of all spanning trees of graph G. The *shortest spanning tree problem* [4] is to find a spanning tree $T_1\in\mathscr{T}$ of minimum edge-weight sum; that is, such that $c(T_1)\leqslant c(T)$, for all $T\in\mathscr{T}$, where $c(T)=\sum_{e\in T} c(e)$. We shall call such a T_1 a *best* spanning tree of graph G. We define an nth-best spanning tree inductively. For $n\geqslant 2$, T_n is an *nth-best spanning tree* of G if (1) T_n is distinct from elements of $\{T_1, T_2,\ldots, T_{n-1}\}$; (2) $c(T_n)\geqslant c(T_{n-1})$, and there exists no $T\in\mathscr{T}$ with $c(T_n)>c(T)\geqslant c(T_{n-1})$.

Kruskal [6] gives a very efficient algorithm for finding a T_1 in a graph G. The problem of finding an nth-best spanning tree T_n in graph G is the natural generalization of Kruskal's problem.

In the following section we show how the properties of spanning trees of a graph G and properties of the 'tree graph $T(G)$', associated with G, relate to the problem of finding an *efficient* T_n-algorithm. Section 3 contains an explicit linear algorithm for finding a T_n based upon these results. Algorithms for related ranking problems [1], [7], are exponential rather than linear. Section 4 extends these results to the problem of ranking the bases of an abstract matroid.

AMS (1970) subject classification: Primary O5B35.

Received October 11, 1973 and, in revised form, November 7, 1974.

§2. The Nth best spanning tree of a graph

We require the following properties of spanning trees. The proofs are routine and will not be reproduced here.

LEMMA 1. *Let $\mathcal{T}$ be the family of spanning trees of graph G. Then*
(1.1) *all spanning trees of G have the same cardinality*; $T \in \mathcal{T}$ and $T' \in \mathcal{T}$ $|T| = |T'| = k$.
(1.2) *for any $T \in \mathcal{T}$ and $y \in E - T$, $T \cup \{y\}$ contains a unique circuit C_y.*
(1.3) *for any two spanning trees T and T^* of $\mathcal{T}$ and any $y \in T^* - T$, there exists an $x \in T$, $x \neq y$, such that $T' = T - \{x\} \cup \{y\} \in \mathcal{T}$. T and T' are said to be* adjacent; *that is $|T \cap T'| = k - 1$.*
(1.4) *for any $J \subseteq E$ which contains no circuits, there exists a spanning tree $T \in \mathcal{T}$ such that $J \subseteq T$; that is, any independent set of edges can be extended to a spanning tree.*

Given edge set E of graph G and family $\mathcal{T}$ defined on E, the *tree graph* $T(G)$ of G is the graph with vertex set $\mathcal{T}$ and edge set the element pairs (T, T') of $\mathcal{T} \times \mathcal{T}$ which are adjacent in the sense of 1.3. Although $T(G)$ has many interesting properties, we require only the following which is implicit in [5], and [8].

LEMMA 2. *The tree-graph $T(G)$ of any graph G is connected.*

Our principal result is contained in the following theorem. We let $\mathcal{T}_{n-1} = \{T_1, T_2, \ldots, T_{n-1}\}$ be a sequence of spanning trees of graph $G = (V, E)$ such that, for $1 \leqslant j \leqslant n-1$, T_j is a jth-best spanning tree of G. A *representative* R of $\mathcal{T}_{n-1}$ is a subset of E such that $R \cap T_j \neq \emptyset$, for $1 \leqslant j \leqslant n-1$. R is *minimal* if no proper subset of R is also a representative of $\mathcal{T}_{n-1}$.

THEOREM 1. *For any n, $n \geqslant 2$, either $\mathcal{T}_{n-1}$ contains all of the spanning trees of G or there exists an nth-best spanning tree T_n which is adjacent to at least one member of $\mathcal{T}_{n-1}$.*

Proof. The proof is by induction on n. Let $\mathcal{T}_1 = \{T_1\}$ be the family consisting of a best spanning tree T_1 of G. If G is not itself a tree, then, by Lemma 1, for every $y \in E - T_1$, $\{y\}$ not itself a circuit, and any x in the corresponding circuit of $T_1 \cup \{y\}$, $T_1 \cup \{y\} - \{x\}$ is a spanning tree of G distinct from T_1. Let T^* be a best such adjacent spanning tree to T_1. That is, $T^* = T_1 \cup \{y^*\} - \{x^*\}$ where the difference $c(y^*) - c(x^*) \geqslant 0$ is a minimum over all $y \in E - T_1$ and corresponding x in the circuit of $T_1 \cup \{y\}$. For any non-adjacent tree T' let $T' = T_1 \cup \{y_\alpha : \alpha \in A\} - \{x_\alpha : \alpha \in A\}$ where for each α in the index set A, $y_\alpha \in E - T_1$ and x_α is the corresponding edge deleted from the circuit of $T_1 \cup \{y_\alpha\}$. For every $\alpha \in A$, $c(y_\alpha) - c(x_\alpha) \geqslant 0$, since otherwise $T_1 \cup \{y_\alpha\} - \{x_\alpha\}$ would be a spanning tree of G having less weight than T_1. Thus,

$$c(T')=c(T_1)+\sum_{\alpha\in A}[c(y_\alpha)-c(x_\alpha)]\geqslant c(T_1)+[c(y^*)-c(x^*)]=c(T^*),$$

and $T^*=T_2$ is thus a second-best spanning tree of G.

Assume the theorem true for $\mathscr{T}_1, \mathscr{T}_2, \ldots, \mathscr{T}_{n-1}$ and that we wish to prove it for $\mathscr{T}_n$. We first must show that if G does not contain a distinct spanning tree adjacent to at least one member of $\mathscr{T}_{n-1}$, then family $\mathscr{T}_{n-1}$ contains all spanning trees of G. (Note that the analogous statement would not be valid in the case of the nth-shortest path problem.) For suppose no spanning tree of $\mathscr{T}-\mathscr{T}_{n-1}$ was adjacent to any member of $\mathscr{T}_{n-1}$. Then, $\mathscr{T}-\mathscr{T}_{n-1}=\emptyset$, since otherwise $(\mathscr{T}_{n-1}, \mathscr{T}-\mathscr{T}_{n-1})$ would be a vertex bipartition of the tree-graph $T(G)$ with no edge having one end in $\mathscr{T}_{n-1}$ and the other in $\mathscr{T}-\mathscr{T}_{n-1}$. This would contradict the connectivity of $T(G)$.

If $\mathscr{T}_{n-1}$ does not contain all spanning trees of G, then let T^* be the best spanning tree of G not in $\mathscr{T}_{n-1}$ but adjacent to at least one member of $\mathscr{T}_{n-1}$. That is, for some $T\in\mathscr{T}_{n-1}$ and minimal representative R of $\mathscr{T}_{n-1}$, $T^*=T\cup\{y^*\}-\{x^*\}$, where the difference $c(y^*)-c(x^*)\geqslant 0$ is a minimum over all $y\in E-R$ and corresponding x in the circuit of $T\cup\{y\}$. As before, for T' any spanning tree of G not adjacent to T, let

$$T'=T\cup\{y_\alpha\colon a\in A\}-\{x_\alpha\colon \alpha\in A\}.$$

Again we have that for every $\alpha\in A$, $c(y_\alpha)-c(x_\alpha)\geqslant 0$, since otherwise $\hat{T}=T\cup\{y_\alpha\}-\{x_\alpha\}$ would be a spanning tree of G *not* in $\mathscr{T}_{n-1}$ (since $y_\alpha\in E-R$ for a representative R of $\mathscr{T}_{n-1}$) with $c(\hat{T})<c(T)$, contradicting the inductive hypothesis. Accordingly, $c(T')=c(T)+\sum_{\alpha\in A}[c(y_\alpha)-c(x_\alpha)]\geqslant c(T)+[c(y^*)-c(x^*)]=c(T^*)$, and $T^*=T_n$ *is* an nth-best spanning tree of G. □

§3. An algorithm for T_n

The following algorithm makes explicit use of the 'adjacency property' of Theorem 1.

Notation. K is the cardinality of any spanning tree of G.

I is the number of spanning trees found such that T_i, $1\leqslant i\leqslant I$ is an ith best spanning tree of G.

$\{J_i\}_{i=1}^{p}$ is the set of distinct $(K-1)$-subsets of members of $F_I=\{T_1, T_2, \ldots, T_I\}$.

R_i is a minimal representative of F_I. For $1\leqslant i\leqslant p$, $R_i=\bigcup(T_j-J_i)$, the fnion being over all $T_j\supset J_i$

E_i is the edge, if one exists, that extends J_i to a minimum weight spanning tree of $E-R_i$.

(1) *Initialization.* Use Kruskal's algorithm on E to determine T_1. Form $\{J_1, J_2, \ldots, J_K\}$ from T_1 by successively removing each edge from T_1. Let $\{R_1, R_2, \ldots, R_K\}$ be the corresponding set of removed edges. For $i=1, 2, \ldots, K$ apply Kruskal's algorithm to $E-R_i$ to determine E_i. Set $T_2=$minimum weight tree from $\{J_i\cup E_i\}_{i=1}^{K}$. Set $I=2$, $P=K$.

(2) If all n best spanning trees are found, stop.
(3) Let the edges of T_I be numbered in any order from 1 to K. For $j=1$ until K do. Remove the jth edge from T_I to form the subset of edges J.
 (a) If $J \notin \{J_i\}_{i=1}^{p}$ then set $J_{p+1}=J$ and
 set R_{p+1} to be the jth edge of T_I.
 set $p=p+1$.
 (b) If $J \in \{J_i\}_{i=1}^{p}$ then $J=J_L$ for some $J_L \in \{J_i\}_{i=1}^{p}$.
 Add the jth edge of T_I to R_L.
(4) Let Q be the index set such that for $i \in Q$ either J_i was formed by step 3(a) or R_i was modified by 3(b) and E_i was equal to the edge added to R_i.
 For $i \in Q$ determine E_i which extends J_i in $E-R_i$ to a minimum weight spanning tree. If no E_i can be found then the weight associated with J_i is set to M, an arbitrary large number.
(5) Set T_{I+1} to be the minimum weight spanning tree corresponding to the P sets $\{J_i \cup E_i\}_{i=1}^{p}$. If the weight of $T_{I+1}=M$ the algorithm stops as all spanning trees have been found. If this is not the case set $I=I+1$ and return to step 2.

Several programming features have been incorporated. These include scanning $E-R_i$ from an appropriate point onward using the initial ordering of the edges. Although such modifications increase the efficiency of the algorithm they are not essential to the proof of convergence.

Proof of convergence. Let $F_{n-1}=\{T_1, T_2,\ldots, T_{n-1}\}$. The set R_i of step 3 is the unique minimal representative of the subfamily of F_{n-1} whose members contain J_i. Thus if $T(J_i)$ of step 4 is a spanning tree of G, it is optimum among all those distinct from F_{n-1} and adjacent to all spanning trees of F_{n-1} which contain J_i. The set $S=\{T(J_i): |T(J_i)|=K,\ 1 \leqslant i \leqslant p\}$ thus contains optimum spanning trees, distinct from F_{n-1} and adjacent to all possible members of F_{n-1}. By Theorem 1, either $S=\emptyset$, in which case F_{n-1} contains all spanning trees of G, or $S \neq \emptyset$, in which case it suffices to take as a T_n, an optimum element of S.

§4. Extensions

The definitions and basic matroid properties used in this section can be found in Edmonds [2], [3]. A *matroid* $M=(E, F)$ is a finite set E of elements and a non-empty family F of subsets of E, called *independent sets*, such that (1) every subset of an independent set is independent; and (2) for every set $A \subseteq E$, all maximal independent subsets of A have the same cardinality, called the *rank* $r(A)$ of A. A *base* B of M is a maximal independent subset of E. A set $A \subseteq E$ is *dependent* if it is not a member of F. A *circuit* is a minimal dependent set.

The matroid axioms are easily applied to give the following matroid version of Lemma 1.

LEMMA 1′. *Let $\mathscr{B}$ be the family of bases of matroid $M=(E, F)$. Then,*

(1.1′) *all bases of M have the same cardinality, equal to $r(E)$.*

(1.2′) *for any $B\in\mathscr{B}$ and $y\in E-B$, $B\cup\{y\}$ contains a unique circuit Cy.*

(1.3′) *for any two bases B and B^* of $\mathscr{B}$ and any $y\in B^*-B$ there exists an $x\in B$, $y\neq x$, such that $B'=B-\{x\}\cup\{y\}\in\mathscr{B}$. Bases B and B' are said to be* adjacent. *That is,* $|B\cap B'|=r(E)-1$.

(1.4′) *any independent set $J\in F$ is contained in some base B of M.*

We define the *base graph* $B(G)$ of M as the graph with vertex set $\mathscr{B}$ and edge set the element pairs (B, B') of $\mathscr{B}\times\mathscr{B}$ which are adjacent in the sense of (1.3′). Using (1.3′) and (1.4′) gives the following matroid version of Lemma 2.

LEMMA 2′. *The base graph $B(G)$ of any matroid M is connected.*

Theorem 1 is a direct consequence of Lemmas 1 and 2. Its matroid equivalent is Theorem 1′. We let $\mathscr{B}_{n-1}=\{B_1, B_2,\ldots, B_{n-1}\}$ be a sequence of bases of $M=(E, F)$ such that, for $1\leqslant j\leqslant n-1$, B_j is a jth-best base of M. As in the case of trees, the ranking of the $\{B_j\}$ is relative to the sum of the element weights, or, more generally, relative to any non-decreasing function of the element weights.

THEOREM 1′. *For any n, $n\geqslant 2$, either $\mathscr{B}_{n-1}=\mathscr{B}$ or there exists an nth-best base B_n of M which is adjacent to at least one member of $\mathscr{B}_{n-1}$.*

The T_n-algorithm of §3 applies as well in this case, but its overall efficiency depends upon having an efficient way of recognizing 'independence' in M. Whenever this is the case, then Edmonds' 'greedy algorithm' [3] provides a truly efficient algorithm for determining a B_1 as well. This is certainly the case when M is *graphic*, E is the edge set of a graph G and F is the family of forests of G.

REFERENCES

[1] BELLMAN, R. and KALABA, R., *On k-th best policies*. J. Soc. Indust. Appl. Math. *8* (1960), 582–588.

[2] EDMONDS, J., *Matroid partition*. In *Mathematics of the Decision Sciences Part 1*, Lectures in Applied Mathematics Vol. 11. Amer. Math. Soc., Providence, R. I., 1968, pp. 335–345.

[3] EDMONDS, J., *Matroids and the Greedy algorithm*. Math. Programming *1* (1971), 127–135.

[4] HARARY, F., *Graph theory*. Addison-Wesley, Reading, Massachusetts, 1969.

[5] HOLZMANN, C. and HARARY F., *On the tree graph of a matroid*. SIAM J. Appl. Math. *22* (1972), 187–193.

[6] KRUSKAL, J., *On the shortest spanning subtree of a graph and the traveling salesman problem*. Proc. Amer. Math. Soc. 7 (1956), 48–50.

[7] POLLACK, M., *Solutions of the kth best route through a network – a review*. J. Math. Anal. Appl. *3* (1961), 547–559.

[8] SHANK, H. S., *Note on Hamilton circuits in tree graphs*. IEEE Trans. C. T. *15* (1968), 86.

Department of Combinatorics and Optimization,
University of Waterloo,
Waterloo, Ontario, Canada.