Math. Program. 84: 565–576 (1999) / DOI 10.1007/s10107990073b

András Frank

Increasing the rooted-connectivity of a digraph by one

Received December 1996 / Revised version received January 1998
Published online March 16, 1999

Abstract. D.R. Fulkerson [7] described a two-phase greedy algorithm to find a minimum cost spanning arborescence and to solve the dual linear program. This was extended by the present author for "kernel systems", a model including the rooted edge-connectivity augmentation problem, as well. A similar type of method was developed by D. Kornblum [9] for "lattice polyhedra", a notion introduced by A. Hoffman and D.E. Schwartz [8].

In order to unify these approaches, here we describe a two-phase greedy algorithm working on a slight extension of lattice polyhedra. This framework includes the rooted node-connectivity augmentation problem, as well, and hence the resulting algorithm, when appropriately specialized, finds a minimum cost of new edges whose addition to a digraph increases its rooted connectivity by one. The only known algorithm for this problem used submodular flows. Actually, the specialized algorithm solves an extension of the rooted edge-connectivity and node-connectivity augmentation problem.

1. Introduction

Let $G = (V, E)$ be a directed graph with a special node s. D.R. Fulkerson [7] developed an algorithm for finding a minimum cost spanning s-rooted arborescence of G and for solving the corresponding linear programming dual problem that consists of a certain cut-packing problem.

By extending Fulkerson's ideas, [3] described a two-phase greedy algorithm for finding a minimum cost subset F of edges of a digraph $G = (V, E)$ so that at least one element of F enters X for every member X of a given intersecting family $\mathcal{F}$ of subsets of V. ($\mathcal{F}$ is called **intersecting** if $X, Y \in \mathcal{F}$, $X \cap Y \neq \emptyset$ imply that $X \cap Y, X \cup Y \in \mathcal{F}$.) This model includes as a special case the minimum cost arborescence problem. Another interesting special case of it consists of finding a minimum cost of new edges whose addition to an existing digraph $D = (V, A)$ increases the rooted edge-connectivity of D by one. That is, the starting digraph D is assumed to have $k-1$ edge-disjoint paths from s to every other node for some positive integer k, and the requirement for the augmented digraph is to have k edge-disjoint paths from s to every other node.

It is quite natural to consider the corresponding problem concerning rooted node-connectivity augmentation. Here the starting digraph D has $k-1$ openly disjoint paths from s to every other node and the requirement for the augmented digraph is to have k

A. Frank: Department of Operations Research, Eötvös University, Rákóczi út 5, Budapest, Hungary, H-1088 and Ericsson Traffic Laboratory, Laborc u.1, Budapest, Hungary, H-1037, e-mail: `frank@cs.elte.hu`.
Research supported by the Hungarian National Foundation for Scientific Research Grant, OTKA T17580. Part of the research was conducted while the author was visiting the Institut IMAG, University J. Fourier (Grenoble, France), in 1996.

Mathematics Subject Classification (1991): 05C70, 90C27

openly-disjoint paths from s to every other node. (Openly disjoint means that the paths are disjoint apart from their end-nodes.) The model in [3] does not cover this problem, and the only known solution, described in [5], consists of a rather tricky way to reduce the problem (by a method which can be carried out in polynomial time) to a submodular flow problem. Since the latter problem admits strongly polynomial time (combinatorial) solution algorithms so does the rooted connectivity augmentation problem. We stress, however, that this submodular flow approach solves even the more general augmentation problem when there is no any connectivity assumption on the starting digraph. In a separate paper [4], by using the model of the present work, we will show a much simpler way to reduce the rooted connectivity augmentation problem to submodular flows.

The main motivation behind the present work was to construct a two-phase greedy algorithm to

increase, at a minimum cost, the rooted node-connectivity of a digraph by one. $(*)$

Actually, the suggested algorithm works on a model which is a slight generalization of what Hoffman and Schwartz [8] call a lattice polyhedron. This abstract framework enables us to solve not only $(*)$ but the following extension, as well. Suppose that a digraph $D = (V, A)$ contains $k - 1$ openly disjoint paths from s to every node in T where T is a specified subset of $V - \{s\}$. Let $G = (V, E)$ be a digraph on the same node set V with the property that every edge of G has its head in T (no restriction on the tails is made). Given a cost function $c : E \to \mathbf{R}_+$, find a minimum cost subset F of edges of G so that the augmented digraph $G^+ := (V, A + F)$ contains k openly disjoint paths from s to every node of T.

The algorithm below may be considered as a unification of D. Kornblum's algorithm for lattice polyhedra [9], Fulkerson's [7] algorithm for minimum cost spanning arborescences as well as its extension for kernel systems [3]. It is worth mentioning that the first algorithm to compute the minimum cost arborescence is due to Yong-Jin Chu and Tseng-Hong Liu [1]. Their method can also be considered as a two-phase greedy algorithm. We also mention a recent related work by Faigle and Kern [2] in which they exhibit a two-phase greedy algorithm concerning submodular linear programs on forests. This model is a generalization of one by Queyranne, Spieksma and Tardella [10] which is a common generalization of Edmonds' greedy algorithm, the NW-corner rule, and a greedy-type algorithm of Hoffman on Monge-matrices.

2. The problem and the algorithm

The following model was introduced in a slightly less general form by Hoffman and Schwartz [8]. Let $(\mathcal{F}, \leq)$ be a partially ordered set with $\mathcal{F} \neq \emptyset$. If $A \leq B, A \neq B$, we write $A < B$. We say that two elements A, B of $\mathcal{F}$ are **intersecting** if A and B are non-comparable and there is a member $C \in \mathcal{F}$ for which $C < A, C < B$. A subset $\mathcal{F}'$ of $\mathcal{F}$ is called **laminar** if $\mathcal{F}'$ contains no two intersecting members.

Suppose we are given two commutative binary operations, the **meet** $\wedge$ and **join** $\vee$, defined on comparable and on intersecting pairs, with the following properties.

Property 1. If $A \leq B$, then $A \wedge B = A$, $A \vee B = B$.

Property 2. If A and B are intersecting, then $A \wedge B < A, B$ and $A \vee B > A, B$.

We say that a non-negative function $p : \mathcal{F} \to \mathbf{R}_+$ is **intersecting supermodular** if

$$p(A) + p(B) \leq p(A \wedge B) + p(A \vee B)$$

holds whenever $p(A) > 0$, $p(B) > 0$ and A, B are intersecting. Function p is called **decreasing** if $A \leq B$ implies that $p(A) \geq p(B)$.

A function $b : \mathcal{F} \to \mathbf{R}$ is called **intersecting submodular** if

$$b(A) + b(B) \geq b(A \wedge B) + b(A \vee B)$$

holds for intersecting A, B.

Furthermore, we are given a set E and a function $\varrho : \mathcal{F} \to 2^E$ satisfying the following properties.

Property 3. If $A \leq B \leq C$, then $\varrho(A) \cap \varrho(C) \subseteq \varrho(B)$. This is called the **consecutive** property.

Property 4. If A, B are intersecting, then $\varrho(A \vee B) \cup \varrho(A \wedge B) \subseteq \varrho(A) \cup \varrho(B)$.

Property 5. If $\varrho(A) \cap \varrho(B) \neq \emptyset$, then A, B are intersecting or comparable.

In the model of Hoffman and Schwartz the join and meet operations were defined on every pair and both Property 2 and 4 were required for every pair A, B. In this case the requirement in Property 5 holds automatically and therefore Property 5 may be omitted.

For example, let $G = (V, E)$ be a digraph with a distinguished node s. Let $\mathcal{F}$ be an intersecting family of (non-empty) subsets of $V - s$ so that each member of $\mathcal{F}$ is entered by at least one edge of G. Let $\leq$ be the relation of containment (that is, $A \leq B$ if $A \subseteq B$.) For two sets with $A \cap B \neq \emptyset$ let $A \wedge B := A \cap B$ and $A \vee B := A \cup B$. Finally let $\varrho(X)$ denote the subset of edges of G entering X. Then the axioms above are satisfied. This model was introduced in Frank [3] under the name "kernel systems". In Section 4 we show an extension that is still covered by the general framework and that will be the key to the rooted node connectivity augmentation problem.

In what follows we describe the problem and the algorithm concerning the abstract framework introduced above but in order to understand the description and the arguments more easily the reader may find it useful to keep in mind the special case of an intersecting family of sets. Working with this, one has to check that no other features of intersecting families are really used than those described in the axioms above. (For example, a possible "other feature" of an intersecting family of sets is that for subsets X, A, B of V the containments $X \subset A$, $X \subset B$ imply $X \subseteq A \cap B$, while in the abstract model it is quite possible to have two intersecting elements $A, B \in \mathcal{F}$ and a third element $X \in \mathcal{F}$ for which $X < A$, $X < B$ and $A \wedge B$ is not comparable to or larger than X. In the general model it is also possible for some $e \in E$ and for intersecting $A, B \in \mathcal{F}$ that $e \in \varrho(A) \cup \varrho(B)$, $e \notin A \wedge B$, while in the special case of digraphs if an edge e enters two members A and B of an intersecting family of subsets, then e enters $A \cap B$, as well.)

For a vector x in $\mathbf{R}^E$ and for $A \in \mathcal{F}$ let $\varrho_x(A) := \sum(x(e) : e \in \varrho(A))$. For $y : \mathcal{F} \to \mathbf{R}_+$ and $e \in E$, let $m(y, e) := \sum(y(X) : e \in \varrho(X))$.

Let $(\mathcal{F}, \leq)$ and E satisfy the axioms above, let $c : E \to \mathbf{R}_+$ be a cost function, and let p be an intersecting supermodular function on $\mathcal{F}$ which is decreasing. The primal linear programming problem we consider is to minimize cx over the vectors x satisfying

$$\varrho_x(A) \geq p(A) \text{ for every } A \in \mathcal{F}, x \geq 0. \tag{1}$$

The dual linear programming problem is to maximize $\sum\Big(y(X)p(X) : X \in \mathcal{F}, y : \mathcal{F} \to \mathbf{R}_+\Big)$, subject to

$$c(e) \geq m(y, e) \text{ for every } e \in E. \tag{2}$$

In order for (1) to have a solution we will assume that $\varrho(A)$ is non-empty for every $A \in \mathcal{F}$ with $p(A) > 0$. The main goal of the present paper is to describe a combinatorial strongly polynomial algorithm to solve this pair of linear programs. The algorithm terminates by outputting a primal optimal solution which is integer-valued when p is integer-valued, and a dual optimal solution which is integer-valued when c is integer-valued. Thus the algorithm provides a constructive proof of the statement that the linear system (1) is totally dual integral (TDI).

Actually, one can consider the following more general optimization problem. Let $0 \leq f \leq g$ be two capacity functions on E, c a cost function on E, and let p be intersecting supermodular (but not necessarily decreasing). With the proof method of Hoffman and Schwartz (the uncrossing technique) one can prove that the linear system

$$\varrho_x(A) \geq p(A) \text{ for every } A \in \mathcal{F}, f \leq x \leq g \tag{3}$$

is TDI. TDI-ness implies that the primal linear program (min cx : x satifies (3)) has an integer-valued optimum x for an arbitrary non-negative cost-function c provided that f, g, p are integer-valued. By the very definition of TDI-ness, the corresponding dual linear program always has an integer-valued optimum whenever c is integer-valued.

We remark that the proof of Hoffman and Schwartz is not algorithmic and there is no known polynomial time algorithm to compute the primal and dual optima in question, not even in the original model defined by Hoffman and Schwartz.

Let us turn to the description of the algorithm. We call a vector x satisfying (1) **primal feasible** and a vector y satisfying (2) **dual feasible.** For such an x and y we say that an element $e \in E$ is y-**tight** or tight if $c(e) = m(y, e)$ and an element $A \in \mathcal{F}$ is x-**tight** or tight if $\varrho_x(A) = p(A)$. The optimality criteria (that is, the complementary slackness conditions) are

(O1) If $x(e) > 0$, then e is y-tight,
(O2) if $y(A) > 0$, then A is x-tight.

The algorithm consists of two phases. In the first one a feasible dual solution y is constructed in a greedy way. In the second phase, using only y-tight elements, a vector x satisfying **(O2)** is constructed in a greedy way. By this construction the optimality criteria are automatically satisfied and the correctness of the algorithm will be proved by showing that the solution x obtained is actually feasible.

Phase 1. The phase consists of steps. In each step a member A of $\mathcal{F}$ is determined along with an element $e \in \varrho(A)$ and the y-value of A is determined. At the beginning let $y \equiv 0$. Suppose that $A_j \in \mathcal{F}$, $e_j \in E$ and $y(A_j)$ have already been determined for $j \leq i-1$. In Step i $(i = 1, 2, \dots)$ check if there is a member $A \in \mathcal{F}$ for which

$$p(A) > 0 \text{ and } \{e_1, \dots, e_{i-1}\} \cap \varrho(A) = \emptyset. \tag{4}$$

If no such an A exists, then the first phase terminates. Set $t := i-1$, $\mathcal{A} := \{A_1, \dots, A_t\}$ and proceed to the second phase. If there is such an A, then choose a smallest element A_i of $\mathcal{F}$ satisfying (4). (Here "smallest" means that no element of the partially ordered set $(\mathcal{F}, \leq)$ that fulfils (2.4) can be smaller than A_i.) Let $\mu_i := \min(c(e) - m(y, e) : e \in \varrho(A_i))$ and revise y by changing the current $y(A_i) = 0$ to $y(A_i) := \mu_i$. Furthermore let e_i denote one of the elements of $\varrho(A_i)$ where the minimum is attained in the definition of μ_i.

Phase 2. At the beginning let $x \equiv 0$. Consider the elements e_i in reverse order and revise the x-value of elements e_i, as follows. Let $x(e_t) := p(A_t)$ and, if $x(e_t)$, $x(e_{t-1}), \dots, x(e_{i+1})$ $(i \geq 1)$ have already been calculated, then let $x(e_i) := p(A_i) - \varrho_x(A_i)$.

Phase 2 and the whole algorithm terminate when $x(e_1)$ has been determined.

3. Proof of correctness

We will need the following lemma.

Lemma 1. *For a non-negative vector x, the set-function ϱ_x is intersecting submodular.*

Proof. The lemma follows (by taking non-negative linear combination) from the special case when $x(e) = 1$ for any specified element e of E and $x(f) = 0$ for every $f \in E - e$. If $e \notin \varrho(A \wedge B) \cup \varrho(A \vee B)$, then $\varrho_x(A) + \varrho_x(B) \geq 0 = \varrho_x(A \wedge B) + \varrho_x(A \vee B)$. If e belongs to exactly one of the sets $\varrho(A \wedge B)$ and $\varrho(A \vee B)$, then, by Property 4, $e \in \varrho(A) \cup \varrho(B)$ and hence $\varrho_x(A) + \varrho_x(B) \geq 1 = \varrho_x(A \wedge B) + \varrho_x(A \vee B)$. Finally, if e belongs to both $\varrho(A \wedge B)$ and $\varrho(A \vee B)$, then, by the consecutive Property 3, e must belong to both A and B and hence $\varrho_x(A) + \varrho_x(B) = 2 = \varrho_x(A \wedge B) + \varrho_x(A \vee B)$. □

The rule of Phase 1 immediately implies that Phase 1 terminates with a feasible dual solution y and that each edge e_i is y-tight. Hence the vector x provided by Phase 2 satisfies **(O1).** Since $e_i \notin \varrho(A_j)$ for $i < j$, the rule of Phase 2 implies that $\varrho_x(A_i) = p(A_i)$ holds for every member $A_i \in \mathcal{A}$ from which **(O2)** follows. In order to prove correctness of the method, we have to show that x is primal feasible. This is equivalent to requiring that $x \geq 0$ and that $h_x(A) \leq 0$ for all $A \in \mathcal{F}$ where $h_x(A) := p(A) - \varrho_x(A)$ denotes the **deficit** of A. We call a member Z of $\mathcal{F}$ **deficient** if $h_x(Z) > 0$. The following is a direct consequence of rule of Phase 1.

Claim 1. If $X < A_j$ for some $j = 1, \dots, t$ and $p(X) > 0$, then there is an index i for which $i < j$ and $e_i \in \varrho(X)$. □

Claim 2. $\mathcal{A}$ is laminar.

Proof. Suppose indirectly that A_j, A_h are two intersecting elements with $j < h$. By applying Claim 1 to $X := A_j \wedge A_h$, we obtain that there is an e_i for which $i < j$ and $e_i \in \varrho(X)$. By Property 4, $e_i \in \varrho(A_j) \cup \varrho(A_h)$, which contradicts the rule of Phase 1 since $i < j < h$. □

Claim 3. $x \geq 0$.

Proof. If A_i is maximal in $\mathcal{A}$, then $x(e_i) = p(A_i) \geq 0$. If A_i is not maximal in $\mathcal{A}$, then let $A_j \in \mathcal{A}$ be such that $A_j > A_i$ and in addition let A_j be minimal with respect to this property. Since $\mathcal{A}$ is laminar by Claim 2, this A_j is uniquely determined. If there is no $e_k \in \varrho(A_i)$ $(k \neq i)$, then $x(e_i) = p(A_i) \geq 0$ (by the rule of Phase 2). If there is such an e_k, then $k > i$, $A_k > A_i$ and hence $A_k \geq A_j$. By the consecutive Property 3, $e_k \in \varrho(A_j)$ from which $p(A_i) = \varrho_x(A_i) \leq x(e_i) + \varrho_x(A_j) = x(e_i) + p(A_j)$. Since p is decreasing, $p(A_i) \geq p(A_j)$, that is, $x(e_i) \geq 0$. □

Our next goal is to prove that there is no deficient element of $\mathcal{F}$. We assume indirectly the opposite and will derive a contradiction.

Claim 4. For a deficient element $X \in \mathcal{F}$ there is an index j so that $1 \leq j \leq t$ and

$$A_j < X, \; e_j \in \varrho(X). \tag{5}$$

Proof. Let j be the smallest index for which $e_j \in \varrho(X)$. (By the stopping rule of Phase 1 there is such an index for every member Y of $\mathcal{F}$ with positive $p(Y)$.) We claim that $A_j < X$. Suppose this is not true. Since $e_j \in \varrho(X) \cap \varrho(A_j)$, Property 5 implies that A_j and X are either intersecting or comparable, that is, $A_j > X$. In both cases, by Claim 1, there is an index $i < j$ for which $e_i \in \varrho(X \wedge A_j)$. Since $e_i \notin \varrho(A_j)$, Property 4 implies that $e_i \in \varrho(X)$ contradicting the minimal choice of j. □

Claim 5. For a deficient element $X \in \mathcal{F}$ and any index j satisfying (5), there is an index k for which

$$e_k \in \varrho(A_j) - \varrho(X), x(e_k) > 0, \tag{6}$$

Proof. Suppose no such an index k exists. Then, since p is decreasing, $A_j < X$ and $x \geq 0$, we would have $p(X) \leq p(A_j) = \varrho_x(A_j) \leq \varrho_x(X)$, contradicting the assumption $h_x(X) > 0$. □

Claim 6. For a deficient element $X \in \mathcal{F}$ and for any indices j, k satisfying (5) and (6)

$$A_j < A_k, \tag{7a}$$

and

$$A_k \text{ and } X \text{ are intersecting.} \tag{7b}$$

Proof. Since $e_k \neq e_j$ (as $e_k \notin \varrho(X), e_j \in \varrho(X)$), we have $k > j$, and hence, by Property 5 and by Claim 2, $A_j < A_k$. We show now that A_k and X are uncomparable. Indeed, if $A_k < X$, then $A_j < A_k < X$ and, by Property 3, $e_j \in \varrho(A_k)$ which is impossible since $k > j$. Similarly, if $A_k > X$, then $A_k > X > A_j$ and Property 3 imply $e_k \in \varrho(X)$, contradicting (6). That is, A_k and X are indeed uncomparable. Since $A_j < X$ and $A_j < A_k$, that is, there is a member of $\mathcal{F}$ smaller than both X and A_k, we conclude that A_k and X are intersecting. □

Let Z be a deficient element of $\mathcal{F}$. Let us choose a pair of indices j, k satisfying (5) and (6) (with Z in place of X) in such a way that k is as small as possible. By Claim 6 we know that A_k and Z are intersecting and hence $A_k \wedge Z < Z < A_k \vee Z$. The key to our proof is the following lemma.

Lemma 2. *$Z' := A_k \wedge Z$ is not deficient.*

Proof. Suppose, indirectly, that Z' is deficient. By applying Claims 4 and 6 to $X := Z'$ we obtain that there are indices j', k' so that

$$A_{j'} < Z', \; e_{j'} \in \varrho(Z'), \tag{5'}$$

$$e_{k'} \in \varrho(A_{j'}) - \varrho(Z'), \; x(e_{k'}) > 0. \tag{6'}$$

We are going to show that the indices j', k' satisfy (5) and (6) with Z in place of X. To see (5), observe first that $A_{j'} < Z$ follows from $A_{j'} < Z'$ and $Z' < Z$. Furthermore, $A_{j'} < Z' < A_k$ implies that $j' < k$ and hence $e_{j'} \notin \varrho(A_k)$. This and $e_{j'} \in \varrho(A_k \wedge Z)$ imply by Property 4 that $e_{j'} \in \varrho(Z)$. That is, j' and $X := Z$ indeed satisfy (3.1).

To see (6), by (6') all we have to show is that $e_{k'} \notin \varrho(Z)$. If this were false, then $A_{j'} < Z' < Z$ would imply, by Property 3, that $e_{k'} \in \varrho(Z')$, a contradiction to (6'). Thus $X := Z$ and indices j' and k' indeed satisfy (6).

It follows from the minimal choice of k that $k \leq k'$. By applying Claim 6 to Z', j', k', we have $A_{j'} < A_{k'}$. Since $A_{j'} < Z' < A_k$ and $\mathcal{A}$ is laminar, A_k and $A_{k'}$ are comparable. This and $k' \geq k$ imply that $A_{k'} \geq A_k$. Hence $A_{k'} \geq A_k > Z' > A_{j'}$ from which, by $e_{k'} \in \varrho(A_{k'}) \cap \varrho(A_{j'})$ and Property 3, we have $e_{k'} \in \varrho(Z')$, contradicting (6'). □

Among the elements of $\mathcal{F}$ of maximum deficit, let Z be a largest. Let j, k be the indices defined before Lemma 2. By the maximal choice of Z we have $h_x(A_k \vee Z) < h_x(Z)$. On the other hand, by Lemma 1 h_x is intersecting supermodular, and hence, by Lemma 2, $h_x(Z) = h_x(Z) + h_x(A_k) \leq h_x(A_k \wedge Z) + h_x(A_k \vee Z) \leq 0 + h_x(A_k \vee Z)$, a contradiction. This contradiction shows that at the end of the algorithm there cannot be any deficient member of $\mathcal{F}$ and thus the proof of the validity of the algorithm is complete. □ □

In order to actually run the algorithm one needs a subroutine in Phase 1 to compute the smallest element of $\mathcal{F}$ satisfying (4). This subroutine is called at most $|E|$ times. The complexity of the other operations is linear in $|E|$.

Remark 1. The main application of the model above concerns the rooted connectivity augmentation problem of a digraph (to be described in the next section). This requires the supermodular function p in question to be identically 1. For this case a simplified proof of the correctness of the algorithm will be shown in Section 4. I must admit I do not know any natural and/or interesting concrete application in which a more general decreasing intersecting supermodular function is involved. (A possible next research task –comparing the present algorithm to that of Faigle and Kern [2] – may reveal such applications).

4. T-intersecting families

In Section 2 we pointed out that intersecting families form a special case of the general framework. Here we exhibit a slight extension that will be the basis for rooted connectivity augmentation.

Let $G = (V, E)$ be a digraph with a distinguished node s and a subset T of $V - s$ so that

$$\text{every edge of } G \text{ has its head in } T. \tag{8}$$

Let $\mathcal{F}$ be a family of (non-empty) subsets of $V - s$ so that (i) each member of $\mathcal{F}$ is entered by at least one edge of G, (ii) if $X, Y \in \mathcal{F}$ and $X \cap Y \cap T \neq \emptyset$, then $X \cap Y, X \cup Y \in \mathcal{F}$. Let $\leq$ be the relation of containment (that is, $A \leq B$ if $A \subseteq B$.) For two sets with $A \cap B \cap T \neq \emptyset$ let $A \wedge B := A \cap B$ and $A \vee B := A \cup B$. Finally let $\varrho(X)$ denote the subset of edges of G entering X. Then the axioms above are easily seen to be satisfied. Two sets X and Y are **T-intersecting** if none of $X \cap Y \cap T$, $X - Y$, $Y - X$ is empty. A family is **T-laminar** if it contains no two T-intersecting members.

Suppose that $p \equiv 1$. In this case an optimum integer-valued vector x in (1) is a $0-1$ vector so we can look for x in the form $x := \chi_F$ where $F \subseteq E$ is a subset of edges and χ_F denotes its characteristic vector. Since rooted connectivity augmentation was the main motivation of this work and the algorithm as well as its proof is simpler in this case, we briefly outline this simplified version.

For a subset $F \subseteq E$ of edges and a member Z of $\mathcal{F}$, we say that F **covers** or **enters** Z or that F is a **covering** of Z if $\varrho_F(Z) \geq 1$, that is, if F contains an edge entering Z. F **covers** $\mathcal{F}$, if it covers every member of $\mathcal{F}$. (This is equivalent to saying that $x := \chi_F$ satisfies (1).)

We assume that E is a covering of $\mathcal{F}$. For a dual feasible vector y, an edge $e \in E$ is **y-tight** if $c(e) = m(y, e)$. For a covering F of $\mathcal{F}$, a set $A \in \mathcal{F}$ is **F-tight** if $\varrho_F(A) = 1$. Now the optimality criteria are, as follows.

(O1) The elements of F are y-tight.
(O2) If $y(A) > 0$, then $\varrho_F(A) = 1$.

The first phase of the algorithm is identical to the one described Section 2. To recall, its outline is as follows.

Phase 1. At the beginning let $y \equiv 0$. In Step i ($i = 1, 2, \ldots$), we assume that a subset $A_j \in \mathcal{F}$, an edge $e_j \in E$ entering A_j, and dual variable $y(A_j)$ have already been determined for every index $j \leq i-1$. Decide if $\mathcal{F}$ has a member A which is not covered by $\{e_1, \ldots, e_{i-1}\}$. If no such a member exists, Phase 1 terminates. Let $t := i - 1$, $\mathcal{A} := \{A_1, \ldots, A_t\}$ and turn to Phase 2. If there is such an A, then let A_i be the smallest such member of $\mathcal{F}$. Let $\mu_i := \min(c(e) - m(y, e) : e \text{ enters } A_i)$ and revise y on A_i by $y(A_i) := \mu_i$. Let e_i be an edge entering A_i where the minimum is attained.

Phase 2. Starting with $F := \emptyset$, build up F by adding edges one by one according to the next rule. Consider the edge set $\{e_1, \ldots, e_t\}$ computed in Phase 1 in reverse order. Starting with e_t, add the current edge e_i to F if A_i is not yet covered by F.

To prove the correctness of the algorithm, first observe that the vector y constructed in Phase 1 is clearly dual feasible. Furthermore every edge e_i is y-tight. Hence every element of F is y-tight, that is, **(O1)** holds. By the rule of Phase 1, we have:

Claim 1. If $X \subset A_j$ for some index $j = 1, \ldots, t$, and for $X \in \mathcal{F}$, then there is an index i for which $i < j$ and e_i enters X. □

Claim 2. $\mathcal{A}$ is T-laminar.

Proof. Suppose indirectly that A_j, A_h are two T-intersecting sets and $j < h$. By applying Claim 1 to $X := A_j \cap A_h$, we obtain that there is an edge e_i for which $i < j$ and e_i enters X. Then e_i enters at least one of A_j and A_h which contradicts the rule of Phase 1 since $i < j < h$. □

Claim 3. Every set A_i is F-tight.

Proof. By the construction of F, $\varrho_F(A_i) \geq 1$. Suppose indirectly that there is an A_i entered by at least two elements of F and assume that A_i is as large as possible. Let e_j and e_k be two elements of F entering A_i. By the rule of Phase 1 and (8), we have $i < j < k$. Since $\mathcal{A}$ is T-laminar we obtain that $A_i \subset A_j \subset A_k$. But then both e_j and e_k enter A_j, contradicting the maximal choice of A_i. □

By Claim 3 we see that **(O2)** is satisfied.

Claim 4. For every set $Z \in \mathcal{F}$ there is an index j for which e_j enters Z and $A_j \subseteq Z$.

Proof. At the end of Phase 1 there is an edge e_j entering Z. Assume that j is as small as possible. We show that $A_j \subseteq Z$. If this were not true, then $X := A_j \cap Z$ would be a proper subset of A_j. By Claim 1 there is an index $i < j$ for which e_i enters X. Then e_i must enter at least one of A_j and Z. On the other hand e_i cannot enter A_j by the rule of the first phase, and it cannot enter Z either by the minimal choice of j, and this contradiction orives the claim. □

Claim 5. F covers $\mathcal{F}$.

Proof. Suppose indirectly that $\mathcal{F}$ has a member Z not covered by F and let Z be as large as possible. By Claim 4 there is an index j for which e_j enters Z and $A_j \subset Z$. Therefore e_j is not in F and hence (by the rule of Phase 2) there is a set A_k for which $A_k \supset A_j$ and e_k enters A_j but not Z. It follows that A_k and Z are T-intersecting. By the maximality of Z, $\varrho_F(A_k \cup Z) \geq 1$, that is, there is an edge $e_l \in F$ entering $A_k \cup Z$. But then e_l enters at least one of A_k and Z which is impossible since Z is not covered by F, and if e_l enters A_k, then A_k is entered by both e_k and e_l contradicting Claim 3. □

5. Rooted connectivity augmentation

Let $D = (V, A)$ be a digraph with a specified node s and a subset $T \subseteq V - s$. For a node $v \in V$ let $\lambda(s, u; D)$ (($\kappa(s, u; D)$, respectively) denote the maximum number of edge-disjoint (openly disjoint) paths of D from s to v. We say that D is k**-edge-connected from** s **to** T (k**-node-connected from** s **to** T) if $\lambda(s, t; D) \geq k$ ($\kappa(s, t; D) \geq k$) holds for every node $t \in T$. If $T = V - s$, we say that the s**-rooted edge-connectivity** (s**-rooted node-connectivity**) of D is at least k. (s-rooted node-connectivity will be sometimes abbreviated by s-rooted connectivity.)

Let $G = (V, E)$ denote the digraph of possible new edges and let c be a cost function on E. The problem we consider consists of determining a minimum cost subset of edges of G whose addition to D results in a digraph D^+ which is k edge- (node-) connected from s to T. This problem is NP-complete even if $k = 1$, c is $0 - 1$-valued, and the starting digraph D has no edges. (This special case is a directed version of the Steiner tree problem.)

However, if G has the special property that

$$\text{every edge of } G \text{ has its head in } T, \tag{9}$$

then both the rooted edge- and the node-connectivity augmentation problem will be shown below to be a special case of the framework described above. Note that if $T = V - s$, then (9) is automatically satisfied. In particular, one can apply the two-phase greedy algorithm to find the required minimum cost augmentation if the starting digraph D is $(k-1)$ edge- (node-) connected from s to T.

Edge-connectivity. Consider first the rooted edge-connectivity augmentation problem. We assume that $\lambda(s, t; D) \geq k - 1$ for every node $t \in T$. Let $\mathcal{F} := \{Z \subseteq V - s, Z \cap T \neq \emptyset, \varrho_D(Z) = k - 1\}$. Since D is $(k-1)$-edge-connected from s to T, we have $(k-1) + (k-1) = \varrho_D(X) + \varrho_D(Y) \geq \varrho_D(X \cap Y) + \varrho_D(X \cup Y) \geq (k-1) + (k-1)$ for any two T-intersecting members X, Y of $\mathcal{F}$. Hence equality follows everywhere from which $\varrho_D(X \cap Y) = \varrho_D(X \cup Y) = k - 1$, that is, $\mathcal{F}$ is T-intersecting.

By the directed edge-version of Menger's theorem F covers $\mathcal{F}$ if and only if $\lambda(s, t; D^+) \geq k$ for every node $t \in T$ where $D^+ := (V, A + F)$ denotes the augmented digraph. Consequently, the algorithm described above can be applied to compute the minimum cost set F.

Node-connectivity. Let us turn to openly disjoint paths. Let $D' = (V', A')$ denote the starting digraph and let T' be a subset of $V' - s'$. Suppose that $\kappa(s', t'; D') \geq k - 1$ fore evey node $t' \in T'$. Let $G' = (V', E')$ denote the digraph of usable new edges We require that

$$\text{every edge of } G' \text{ has its head in } T'. \tag{9'}$$

The goal is to find a minimum cost subset F of edges of G' whose addition to D' results in a digraph in which there are k openly disjoint paths from s' to every node of T'. This problem can be reduced to the edge-disjoint case by a simple, well-known technique called node-duplication, as follows.

For every node $v \in V'$ replace v by two new nodes v_t and v_h. Let $s := s_h$. Let V denote the set of new nodes and let $T := \{v_t : v \in T'\}$. Let $D = (V, A)$ denote a digraph in which every original edge $uv \in A'$ determines an edge $u_h v_t$ and every node $v \in V$ determines an edge $u_t u_h$. In a digraph $G = (V, E)$ we associate an edge $u_h v_t$ with every original edge $uv \in E'$ (however no edge of type $u_t u_h$ nelongs to G).

It is easy to check that $\kappa(s', v; D') = \lambda(s, v_t; D)$ holds for every node $v \in T'$. Moreover, for a subset $F' \subseteq E'$ of edges, by letting F denote the subset of edges of G corresponding to F', we obtain that the digraph $(V, A' + F')$ contains k openly disjoint paths from s' to v if and only if there are k edge-disjoint paths in $(V, A + F)$ from s to v_t. That is, the rooted node-connectivity augmentation problem reduces to the rooted edge-connectivity augmentation problem.

Therefore, in order to have an estimation for the complexity of the algorithm it suffices to consider the edge-connectivity case. Recall that the family $\mathcal{A}$ provided by Phase 1 is laminar and hence it has at most $2n$ members where $n = |V|$.

The first phase of the algorithm needs a subroutine to check whether, given a subset $E_j = \{e_1, \dots, e_j\} \subseteq E$ of edges, there exists a member of $\mathcal{F}$ not entered by E_j, and if so, the subroutine determines a minimal such member. In the concrete case of rooted edge-connectivity augmentation this subroutine can be easily constructed, as follows. E_j covers every member of $\mathcal{F}$ if and only if there are k edge-disjoint paths from s to every element of T in the digraph $(V, A \cup E_j)$. For one fixed element t of T this can be decided by a max-flow min-cut computation. Moreover, if the MFMC algorithm finds no k edge-disjoint paths from s to t, then it is able to determine a minimal subset Z of nodes for which $t \in Z \subseteq V - s$ and exactly $k - 1$ edges from $A \cup E_j$ enter Z. Therefore by $|T|$ applications of the MFMC computation the required subroutine is indeed available. Let $M(n)$ denote the complexity of an MFMC algorithm on a digraph with n nodes. Then the first phase of the algorithm requires at most $O(|\mathcal{A}||T|M(n)) \leq O(n^2 M(n))$ elementary steps. Since there are MFMC algorithms of complexity $O(n^3)$ and this part majorizes the complexity of the whole algorithm, we can conclude that the overall number of steps of the algorithm is at most $O(n^5)$.

With some care however this bound can be reduced. Namely, suppose for a specified node $v \in T$ that an MFMC algorithm has already computed $(k-1)$ edge-disjoint paths from s to v along with a minimal set Z possessing exactly $k - 1$ edges of the digraph $(V, E \cup \{e_1, \dots e_j\})$. Suppose we add a new edge e_{j+1} to the digraph. Then we need only one augmenting step of the MFMC algorithm to decide whether there are k edge-disjoint paths from s to t, and this can be done in linear time. That is, at the beginning we need $|T|$ MFMC computations but after that taking one subsequent edge

e_j into considerations only $O(|T|(|E| + |E_j|)) \leq O(n^3)$ steps are required. Hence the complexity of the algorithm is at most $O(n^4)$.

Remark 2. The reader may think that working with a general T and assuming (9) is a bit artificial and it would be more natural to consider only the special case $T = V - s$. Note however that the case $T = V - s$ for openly disjoint paths does not reduce to the case $T = V - s$ for edge-disjoint paths.

Remark 3. Let $G = (V, E)$ be a directed graph with a specified node s and a terminal set $T \subseteq V - s$. Let $c : E \to \mathbf{R}_+$ be a cost function such that $c(uv) = 0$ for every $uv \in E$ with $v \in V - T$. The problem $(*)$ of finding a minimum cost arborescence of root s that contains every element of T is a special case of the abstract framework described above (by letting $\mathcal{F}$ consist of all subsets of $V - s$ with a non-empty intersection with T). Problem $(*)$ is apparently more general than the problem of finding a minimum cost spanning arborescence (where $T := V - s$), however there is a straightforward elementary construction to reduce $(*)$ to this latter problem.

Acknowledgements. Special thanks are due to Ulrich Faigle and to his Ph.D. student Daniel Paulusma who drew my attention to a gap in the original proof of the validity of the algorithm.

References

1. Chu, Y.-J., Liu, T.-H. (1965): On the shortest arborescence of a directed graph. Scientia Sinica, Vol. XIV, No. 10, pp 1396–1400
2. Faigle, U., Kern, W. (1996): Submodular linear programs on forests. Math. Program. **72**, 195–206
3. Frank, A. (1979): Kernel systems of directed graphs. Acta Sci. Math. (Szeged) **41**, 63–76
4. Frank, A. (1998): A note on finding minimum cost k-root-connected subgraphs. In preparation
5. Frank, A., Tardos, É. (1989): An application of submodular flows. Linear Algebra Appl. **114/115**, 329–348
6. Fujishige, S. (1991): Submodular functions and optimization. Ann. Discrete Math. **47**, North-Holland
7. Fulkerson, D.R. (1974): Packing rooted directed cuts in a weighted directed graph. Math. Program. **6**, 1–13
8. Hoffman, A., Schwartz, D.E. (1978): On lattice polyhedra. In: Hajnal, A., Sós, V.T., eds., Proceedings of Fifth Hungarian Combinatorial Coll., pp. 593–598. North-Holland, Amsterdam
9. Kornblum, D. (1978): Greedy algorithms for some optimization problems on a lattice polyhedron. PhD thesis, Graduate Center of the City University of New York
10. Queyranne, M., Spieksma, F., Tardella, F. (1993): A general class of greadily solvable linear programs. In: Rinaldi, G., Wolsey, L., eds., Integer Programming and Combinatorial Optimization. Proceedings of the Third IPCO Conference, CORE, Louvain-La-Neuve, pp. 385-399