Egerváry Research Group
on Combinatorial Optimization

Technical reportS

TR-2011-13. Published by the Egerváry Research Group, Pázmány P. sétány 1/C, H–1117, Budapest, Hungary. Web site: www.cs.elte.hu/egres . ISSN 1587–4451.

Covering minimum cost arborescences

Attila Bernáth and Gyula Pap

December 2011

Covering minimum cost arborescences

Attila Bernáth⋆ and Gyula Pap⋆⋆

Abstract

Given a digraph $D = (V, A)$ with a designated root node $r \in V$ and arc-costs $c : A \to \mathbb{R}$, we consider the problem of finding a minimum cardinality subset H of the arc set A such that H intersects every minimum cost r-arborescence. We give a polynomial algorithm for finding such an arc set H. The algorithm solves a weighted version as well, in which a nonnegative weight function $w : A \to \mathbb{R}_+$ is also given, and we want to find a subset H of the arc set such that H intersects every minimum cost r-arborescence, and $w(H)$ is minimum.

1 Introduction

Let $D = (V, A)$ be a digraph with vertex set V and arc set A. A *spanning arborescence* is a subset $B \subseteq A$ that is a spanning tree in the undirected sense, and every node has in-degree at most one. Thus there is exactly one node, the root node, with in-degree zero. Equivalently, a spanning arborescence is a subset $B \subseteq A$ with the property that there is a root node $r \in V$ such that $\varrho_B(r) = 0$, and $\varrho_B(v) = 1$ for $v \in V - r$, and B contains no cycle. An **arborescence** will mean a spanning arborescence, unless stated otherwise. If $r \in V$ is the root of the spanning arborescence B then we will say that B is an r-**arborescence**.

The Minimum Cost Arborescence Problem is the following: given a digraph $D = (V, A)$, a designated root node $r \in V$ and a cost function $c : A \to \mathbb{R}$, find an r-arborescence $B \subseteq A$ such that the cost $c(B) = \sum_{b \in B} c(b)$ of B is smallest possible. Fulkerson [2] has given a two-phase algorithm for solving this problem, and he also characterized minimum cost arborescences. Naoyuki Kamiyama in [4] raised the following question.

Problem 1. Given a digraph $D = (V, A)$, a designated root node $r \in V$ and a cost function $c : A \to \mathbb{R}$, find a subset H of the arc set such that H intersects every minimum cost r-arborescence, and $|H|$ is minimum.

⋆MTA-ELTE Egerváry Research Group, Department of Operations Research, Eötvös University, Pázmány Péter sétány 1/C, Budapest, Hungary, H-1117. E-mail: `bernath@cs.elte.hu`.

⋆⋆MTA-ELTE Egerváry Research Group, Department of Operations Research, Eötvös University, Pázmány Péter sétány 1/C, Budapest, Hungary, H-1117. E-mail: `gyuszko@cs.elte.hu`.

The minimum in Problem 1 measures the robustness of the minimum cost arborescences, since it asks to delete a minimum cardinality set of arcs in order to destroy all minimum cost r-arborescences.

This problem was raised and investigated by Naoyuki Kamiyama in [4], where he solved special cases of this problem and he investigated some necessary and sufficient conditions for the minimum in this problem. In this paper we give a polynomial time algorithm solving Problem 1. In fact, our algorithm will solve the following, more general problem, too.

Problem 2. Given a digraph $D = (V, A)$, a designated root node $r \in V$, a cost function $c : A \to \mathbb{R}$ and a nonnegative weight function $w : A \to \mathbb{R}_+$, find a subset H of the arc set such that H intersects every minimum cost r-arborescence, and $w(H)$ is minimum.

The rest of this paper is organized as follows. In Section 2 we give variants of the problem based on Fulkerson's characterization of minimum cost arborescences. These variants are all equivalent with Problem 1 as simple reductions show, but we deal with the variant that can be handled more conveniently. In Section 3 we solve the special case of covering all arborescences: this is indeed a very special case, but the answer is very useful in the solution of the general case. Section 4 contains our main result broken down into two steps: in Section 4.1 we prove a min-min formula that gives a useful reformulation of our problem, and –after introducing some essential results and techniques in Section 4.2– we finally give a polynomial time algorithm solving Problems 1 and 2 in Section 4.3.

2 The problem and its variants

In this paper we investigate Problem 1 and the more general Problem 2. One interpretation of these problems is that we want to *cover the minimum cost common bases of two matroids*: one matroid being the graphic matroid of D (in the undirected sense), the other being a partition matroid with partition classes $\delta^{in}(v)$ for every $v \in V$. A related problem for matroids, the problem of *covering all minimum cost bases of a matroid* is solved in [3]. For sake of simplicity we will mostly speak about Problem 1, and in Section 4.3 we sketch the necessary modifications of our algorithm needed to solve Problem 2. Note that Problem 2 with an integer weight function w can be reduced to Problem 1 by replacing an arc $a \in A$ (of weight $w(a)$) with $w(a)$ parallel copies (each of weight 1): this reduction is however not polynomial. On the other hand the algorithm we give for Problem 1 can be simply modified to solve Problem 2 in strongly polynomial time.

Let us give some more definitions. The arc set of the digraph D will also be denoted by $A(D)$. Given a digraph $D = (V, A)$ and a node set $Z \subseteq V$, let $D[Z]$ be the digraph obtained from D by deleting the nodes of $V - Z$ (and all the arcs incident with them). If $B \subseteq A$ is a subset of the arc set, then we will sometimes abuse the notation by identifying B and the graph (V, B): thus $B[Z]$ is obtained from (V, B) by deleting the nodes of $V - Z$ (and the arcs of B incident with them). The set of arcs of D entering Z is denoted $\delta_D^{in}(Z)$, the number of these arcs is $\varrho_D(Z) = |\delta_D^{in}(Z)|$.

The following theorem of Fulkerson characterizes the minimum cost arborescences and leads us to a more convenient, but equivalent problem.

Theorem 2.1 (Fulkerson, [2]). *There exists a subset $A' \subseteq A$ of arcs (called* tight arcs*) and a laminar family $\mathcal{L} \subseteq 2^{V-r}$ such that an r-arborescence is of minimum cost if and only if it uses only tight arcs and it enters every member of $\mathcal{L}$ exactly once. The set A' and $\mathcal{L}$ can be found in polynomial time.*

Since non-tight arcs do not play a role in our problems, we can forget about them, so we assume that $A' = A$ from now on.

Let $\mathcal{L}$ be a laminar family of subsets of V. A spanning arborescence $B \subseteq A$ in D is called an *$\mathcal{L}$-nice arborescence* if both of the following hold.

1. $|\delta_B^{in}(F)| \leq 1$ for all $F \in \mathcal{L}$, and
2. $|\delta_B^{in}(F)| = 0$ for all $F \in \mathcal{L}$ containing the root r of B.

We point out that the second condition in the above definition is needed because we don't want to fix the root of the arborescences: this will be natural in the solution we give for Problem 1. The result of Fulkerson leads us to the following problem.

Problem 3. Given a digraph $D = (V, A)$, a designated root node $r \in V$ and a laminar family $\mathcal{L} \subseteq 2^V$, find a subset H of the arc set such that H intersects every r-rooted $\mathcal{L}$-nice arborescence and $|H|$ is minimum.

Note that in this problem we allow that $r \in F$ for some members $F \in \mathcal{L}$. By Fulkerson's Theorem above, if we have a polynomial algorithm for Problem 3 then we can also solve Problem 1 in polynomial time with this algorithm. However, this can be reversed by the next claim.

Claim 2.2. *If we have a polynomial algorithm solving Problem 1 then we can also solve Problem* 3 *in polynomial time.*

Proof. Let the cost of an arc $a \in A$ be equal to the number of sets $F \in \mathcal{L}$ such that a enters F. Then an r-arborescence is of minimum cost if and only if it is $\mathcal{L}$-nice, if there exists an $\mathcal{L}$-nice arborescence at all. □

We point out that the construction in the above proof also shows how to find an $\mathcal{L}$-nice arborescence, if it exists at all. So we can turn our attention to Problem 3. However, in order to have a more compact answer, it is more convenient to consider the following, equivalent problem instead, in which the root is not designated.

Problem 4. Given a digraph $D = (V, A)$ and a laminar family $\mathcal{L} \subseteq 2^V$, find a subset H of the arc set such that H intersects every $\mathcal{L}$-nice arborescence and $|H|$ is minimum.

Claim 2.3. *There exists a polynomial algorithm solving Problem 3 if and only if there exists a polynomial algorithm solving Problem 4.*

Proof. Assume that there exists a polynomial time algorithm for Problem 4 and consider an instance of Problem 3. Since arcs entering r will not be used in an optimal solution of Problem 3, we can assume that there are no such arcs. But then the $\mathcal{L}$-nice arborescences are all rooted in r, so our algorithm covering all $\mathcal{L}$-nice arborescences can be used for solving Problem 3, too.

For the other direction assume that we have a polynomial time algorith solving Problem 3 and consider an instance of Problem 4 given with $D = (V, A)$ and $\mathcal{L} \subseteq 2^V$. Let $V' = V + r'$ with a new node r' and let $D' = (V', A')$ where A' consists of the arcs in A and an arc of multiplicity $|A|+1$ from r' to every $v \in V$. Finally let $\mathcal{L}' = \mathcal{L} + \{V\}$. Now consider Problem 3 with input D', r' and $\mathcal{L}'$. Observe that $\mathcal{L}$-nice arborescences in D and (r'-rooted) $\mathcal{L}'$-nice arborescences in D' correspond to each-other in a natural way, and an optimal solution to this instance of Problem 3 will not contain any arc of form $r'v$ (if it contains one then it has to contain all parallel copies, but we included those with a large multiplicity). □

The main result of this paper is a polynomial algorithm solving Problem 4, and thus, by Claims 2.2 and 2.3, for Problems 1 and 3. For a digraph $D = (V, A)$, and a laminar family $\mathcal{L}$ of subsets of V, let $\gamma(D, \mathcal{L})$ denote the minimum number of arcs deleted from D to obtain a digraph that does not contain an $\mathcal{L}$-nice arborescence, that is,

$$\gamma(D, \mathcal{L}) := \min\{|H| : H \subseteq A \text{ such that } D - H \text{ contains no } \mathcal{L}\text{-nice arborescence }\}. \tag{1}$$

3 Covering all arborescences – a special case

In the proof of our main result below, we will use its special case when the laminar family $\mathcal{L}$ is empty. This special case amounts to the following well-known characterization of the existence of a spanning arborescence.

Lemma 3.1. *For any digraph $D = (V, A)$ exactly one of the following two alternatives holds:*

1. *there exists a spanning arborescence,*
2. *there exist two disjoint non-empty subsets $Z_1, Z_2 \subset V$ such that $\varrho_D(Z_1) = \varrho_D(Z_2) = 0$.*

This characterization also implies a formula to determine the minimum number of edges to be deleted to destroy all arborescences. The characterization is based on double cuts.

Definition 3.2. For a digraph $D = (V, A)$, a *double cut* $\delta^{in}(Z_1) \cup \delta^{in}(Z_2)$ is determined by a pair of non-empty disjoint node subsets $Z_1, Z_2 \subseteq V$. The minimum cadinality of a double cut is denoted by $\mu(D)$, that is

$$\mu(D) := \min\{|\delta^{in}(Z_1)| + |\delta^{in}(Z_2)| : Z_1 \cap Z_2 = \emptyset \neq Z_1, Z_2 \subsetneq V\}. \tag{2}$$

Corollary 3.3. *For any digraph $D = (V, A)$ the following equation holds: $\gamma(D, \emptyset) = \mu(D)$.*

We point out that a minimum double cut can be found in polynomial time by a simple reduction to minimum cut. Furthermore we will need the following observation.

Lemma 3.4. *Given a digraph $D = (V, A)$ let $R = \{r \in V$: there exists an r-rooted spanning arborescence in $D\}$. Then $D[R]$ is a strongly connected digraph, and $\varrho_D(R) = 0$.*

4 Covering nice arborescences

Given a laminar family $\mathcal{L} \subseteq 2^V$, for $F \in \mathcal{L} \cup \{V\}$, let $\mathcal{L}_F := \{F' \in \mathcal{L}, F' \subseteq F\}$. A simple, albeit quite important observation is that members of the laminar family also induce a nice arborescence with respect to the laminar family, thus we obtain the following Claim.

Claim 4.1. *For any $\mathcal{L}$-nice arborescence B, and any $F \in \mathcal{L} \cup \{V\}$, $B[F]$ is an $\mathcal{L}_F$-nice arborescence in $D[F]$.*

The following observation is crucial in our proofs. Given a digraph $D = (V, A)$ and a laminar family $\mathcal{L} \subseteq 2^V$, for an arbitrary member $F \in \mathcal{L}$ and arc $a = xy \in A$ *leaving* F, let $\tilde{D}$ be the graph obtained from D by changing the tail of a for an arbitrary other node $x' \in F$, that is $\tilde{D} = D - xy + x'y$ (where $x, x' \in F$ and $y \notin F$). This operation will be called a **tail-relocation**. Then clearly there is a natural bijection between the arcs of D and those of $\tilde{D}$, but even more importantly, this bijection also induces a bijection between the $\mathcal{L}$-nice arborescences in D and those in $\tilde{D}$. This is formulated in the following claim.

Claim 4.2. *Let $B \subseteq A$ and $xy \in B$. Then $B - xy + x'y$ is an $\mathcal{L}$-nice arborescence in $\tilde{D}$ if and only if B is an $\mathcal{L}$-nice arborescence in D.*

The claim also implies that $\gamma(D, \mathcal{L}) = \gamma(\tilde{D}, \mathcal{L})$.

4.1 A "min-min" formula

Our approach to determine $\gamma(D, \mathcal{L})$ is broken down into two steps. First, we prove a "min-min" formula, that is, we show that a set H that attains the minimum in (1) is equal to a special arc subset called an $\mathcal{L}$-*double-cut*. The second step will be the construction of an algorithm to find a minimum cardinality $\mathcal{L}$-double-cut.

So what is this first step – the min-min formula all about? It expresses that in order to to cover optimally the $\mathcal{L}$-nice arborescences we need to consider the problem of covering the $\mathcal{L}_F$-nice arborescences for every $F \in \mathcal{L} \cup \{V\}$.

Definition 4.3. For a set $Z \subseteq V$, let $\mathcal{L}_Z$ denote the family of sets in $\mathcal{L}$ not disjoint from Z, that is, let

$$\mathcal{L}_Z := \{F \in \mathcal{L} : F \cap Z \neq \emptyset\}. \tag{3}$$

Then an $\mathcal{L}$-*cut* $M(Z)$ is defined as the set of arcs entering Z, but not leaving any set in $\mathcal{L}_Z$, that is, let

$$M(Z) := M_{D,\mathcal{L}}(Z) := \delta_D^{in}(Z) - \bigcup_{F \in \mathcal{L}_Z} \left(\delta_D^{out}(F)\right). \tag{4}$$

Note that for a set $F \in \mathcal{L} \cup \{V\}$ this definition of $\mathcal{L}_F$ does not contradict with the definition given in the beginning of Section 4. Thus $M(Z)$ consists of those arcs entering Z, but not leaving any of those sets in $\mathcal{L}$ that have non-empty intersection with Z. A set function f is given by the cardinality of an $\mathcal{L}$-cut, that is, we define

$$f(Z) := f_D(Z) := f_{D,\mathcal{L}}(Z) := |M_{D,\mathcal{L}}(Z)|. \tag{5}$$

It is useful to observe that

$$f_{D,\mathcal{L}}(Z) \geq f_{D[F],\mathcal{L}_F}(Z \cap F) \text{ for any } F \in \mathcal{L}. \tag{6}$$

The motivation for f and $M(Z)$ is that $H = M(Z)$ is a set of arcs the deletion of which destroys all nice arborescences rooted outside of Z, as claimed by the following lemma.

Lemma 4.4. *For any $\emptyset \neq Z \subsetneq V$, there is no $\mathcal{L}$-nice arborescence in $D - M(Z)$ rooted in a node $s \in V - Z$.*

Proof. Let $\bar{D} = D - M(Z)$. We prove the lemma by induction on $|\mathcal{L}|$: the base case when $\mathcal{L} = \emptyset$ is obvious. So let $|\mathcal{L}| > 0$ and assume that $P \subseteq A - M(Z)$ is an s-rooted $\mathcal{L}$-nice arborescence in $\bar{D}$ (where $s \in V - Z$). First observe that if $F \in \mathcal{L}_Z$ is arbitrary then, by the induction hypothesis, the root of the $\mathcal{L}_F$-nice arborescence $P[F]$ must be in $F \cap Z$. Let $v \in Z$ be arbitrary and consider the unique path in P from s to v: assume that $a \in A(\bar{D})$ is the first arc on this path that enters Z. Then there must exist a set $F \in \mathcal{L}_Z$ such that a leaves F. But the root of $P[F]$ must precede a on this path, and it lies in Z, a contradiction. □

For any $F \in \mathcal{L} \cup \{V\}$ and nonempty disjoint subsets $Z_1, Z_2 \subseteq F$ the set of arcs in $M_{D[F],\mathcal{L}_F}(Z_1) \cup M_{D[F],\mathcal{L}_F}(Z_2)$ will be called an $\mathcal{L}$-*double cut*, and we introduce the following notation for the minimum cardinality of an $\mathcal{L}$-double cut:

$$\Theta_F := \Theta_{F,D} := \Theta_{F,D,\mathcal{L}} := \min\{f_{D[F],\mathcal{L}_F}(Z_1) + f_{D[F],\mathcal{L}_F}(Z_2) : \emptyset \neq Z_1, Z_2 \subseteq F, Z_1 \cap Z_2 = \emptyset\}.$$

The following simple observation is worth mentioning.

Claim 4.5. *Given a digraph $D = (V, A)$ and a laminar family $\mathcal{L} \subseteq 2^V$, then $f_{D,\mathcal{L}}(Z) \leq \varrho_D(Z)$ holds for every $Z \subseteq V$. Consequently, $\Theta_{F,D,\mathcal{L}} \leq \mu(D[F])$ holds for any $F \in \mathcal{L} \cup \{V\}$.*

Note that the tail-relocation operation introduced above does not change f-value of any set $Z \subseteq V$, that is $f_{D,\mathcal{L}}(Z) = f_{D',\mathcal{L}}(Z)$, if D' is obtained from D by (one or several) tail-relocation. Consequently, this operation does not modify the Θ value, either, that is $\Theta_{F,D,\mathcal{L}} = \Theta_{F,D',\mathcal{L}}$ for any $F \in \mathcal{L} \cup \{V\}$. The following "min-min" theorem motivates the definition of Θ.

Theorem 4.6. *For a digraph $D = (V, A)$, and a laminar family $\mathcal{L}$ of subsets of V, the minimum number of arcs to be deleted from D to obtain a digraph that does not contain an $\mathcal{L}$-nice arborescence is attained on an $\mathcal{L}$-double cut, that is*

$$\gamma(D, \mathcal{L}) = \min_{F \in \mathcal{L} \cup \{V\}} \Theta_{F,D,\mathcal{L}}.$$

Proof. By Lemma 4.4, $\gamma(D, \mathcal{L}) \leq \min_{F \in \mathcal{L} \cup \{V\}} \Theta_F$, since if we delete an arc set $M_{D[F]}(Z_1) \cup M_{D[F]}(Z_2)$ for some $F \in \mathcal{L} \cup \{V\}$ and non-empty disjoint $Z_1, Z_2 \subseteq F$, then no $\mathcal{L}_F$-nice arborescence survives in $D[F]$ (since its root can neither be in $F - Z_1$, nor in $F - Z_2$, by Lemma 4.4, and $F = (F - Z_1) \cup (F - Z_2)$).

Assume that $H \subseteq A$ is such that $|H| < \min_{F \in \mathcal{L} \cup \{V\}} \Theta_F$: we will show that there exists an $\mathcal{L}$-nice arborescence in $\bar{D} = D - H$, proving the theorem. It suffices to show the following lemma.

Lemma 4.7. *If $f_{\bar{D}[F]}(Z_1) + f_{\bar{D}[F]}(Z_2) > 0$ for any $F \in \mathcal{L} \cup \{V\}$ and non-empty disjoint sets $Z_1, Z_2 \subseteq F$, then there exists a $\mathcal{L}$-nice arborescence in $\bar{D}$.*

Proof. We will use induction on $|\mathcal{L}| + |V| + |A(\bar{D})|$. If $\mathcal{L} = \emptyset$ then the lemma is true by Lemma 3.1. Otherwise let $F \in \mathcal{L}$ be an inclusionwise minimal member of $\mathcal{L}$: again by Lemma 3.1, there exists a spanning arborescence in $\bar{D}[F]$. Let R be the subset of nodes of F that can be the root of a spanning arborescence in $\bar{D}[F]$, i.e. $R = \{r \in F :$ there exists an r-rooted arborescence (spanning F) in $\bar{D}[F]\}$.

1. Assume first that $|R| \geq 2$ and let $\bar{D}_1 = \bar{D}/R$ obtained by contracting R. For any set $Z \subseteq V$ which is either disjoint form R, or contains R, let Z/R be its (well-defined) image after the contraction and let $\mathcal{L}_1 = \{X/R : X \in \mathcal{L}\}$. By induction, there exists an $\mathcal{L}_1$-nice arborescence P in $\bar{D}_1$, since $f_{\bar{D}[X/R]}(Z/R) = f_{\bar{D}[X]}(Z)$ for any $X/R \in \mathcal{L}_1$ and $Z/R \subseteq X/R$. It is clear that we can create an $\mathcal{L}$-nice arborescence in $\bar{D}$ from P: we describe one possible way. Consider the unique arc in P that enters F and assume that the pre-image of this arc has head $r \in R$. Delete every arc from P induced by F/R and substitute them with an arbitrary r-rooted arborescence (spanning F) of $\bar{D}[F]$. This clearly gives an $\mathcal{L}$-nice arborescence.

2. So we can assume that $R = \{r\}$. Next assume that there exists an arc $uv \in A(\bar{D})$ entering F with $r \neq v$. Let $\bar{D}_2 = \bar{D} - uv$: we claim that there exists an $\mathcal{L}$-nice arborescence in $\bar{D}_2$ (which is clearly an $\mathcal{L}$-nice arborescence in $\bar{D}$, too). If this does not hold then by the induction there must exist a set $F' \in \mathcal{L}$ and non-empty disjoint subsets $Z_1, Z_2 \subseteq F'$ with $\sum_{i=1,2} f_{\bar{D}_2[F']}(Z_i) = 0$. Since $\sum_{i=1,2} f_{\bar{D}[F']}(Z_i) > 0$, the arc uv must be equal to (say) $M_{\bar{D}[F'],\mathcal{L}}(Z_1)$ (while $M_{\bar{D}[F'],\mathcal{L}}(Z_2) = \emptyset$). This implies that uv enters Z_1, while $r \in Z_1$ must also hold, otherwise $f_{\bar{D}[F']}(Z_1) \geq 2$ would hold, since v is reachable from r in $\bar{D}[F']$. Let $Z_1' = Z_1 - (F - r)$ and observe that $f_{\bar{D}[F']}(Z_1') = 0$: this is because the arcs in $\delta^{in}_{\bar{D}[F']}(Z_1') - \delta^{in}_{\bar{D}[F']}(Z_1)$ all leave F, since $\varrho_{\bar{D}[F]}(r) = 0$ by Lemma 3.4. Thus $f_{\bar{D}[F']}(Z_1') + f_{\bar{D}[F']}(Z_2) = 0$, a contradiction.

3. So we can also assume that the arcs of $\bar{D}$ entering F all enter r. Let $\mathcal{L}_2 = \mathcal{L} - \{F\}$: then clearly $f_{\bar{D}[F'],\mathcal{L}_2}(Z) \geq f_{\bar{D}[F'],\mathcal{L}}(Z)$ for any $F' \in \mathcal{L}_2$ and $Z \subseteq F'$, so by induction there exists an $\mathcal{L}_2$-nice arborescence in $\bar{D}$: by our assumptions this is also $\mathcal{L}$-nice, so the theorem is proved.

□

4.2 Double cuts and arborescences

In this section we give some important results for the main theorem.

Definition 4.8. A family of sets $\mathcal{F} \subseteq 2^V$ of a finite ground set V is said to satisfy the Helly-property, if any sub-family $\mathcal{X}$ of pairwise intersecting members of $\mathcal{F}$ has a non-empty intersection, i.e. $\mathcal{X} \subseteq \mathcal{F}$ and $X \cap X' \neq \emptyset$ for every $X, X' \in \mathcal{X}$ implies that $\cap\mathcal{X} \neq \emptyset$.

The following definition is taken from [1].

Definition 4.9. Given a digraph $G = (V, A)$, a non-empty subset of nodes $X \subseteq V$ is called *in-solid*, if $\varrho(Y) > \varrho(X)$ holds for every nonempty $Y \subsetneq X$.

Theorem 4.10 (Bárász, Becker, Frank [1])**.** *The family of in-solid sets of a digraph satisfies the Helly-property.*

The authors of [1] prove in fact more: they show that the family of in-solid sets is a *subtree-hypergraph*, but we will only use the Helly property here. The following theorem formulates the key observation for the main result.

Theorem 4.11. *In a digraph $G = (V, A)$ there exists a node $t \in V$ such that $\varrho(Z) \geq \frac{\mu(G)}{2}$ for every non-empty $Z \subseteq V - t$.*

Proof. Consider the family $\mathcal{X} = \{X \subseteq V : X$ is in-solid and $\varrho(X) < \frac{\mu(G)}{2}\}$. If there were two disjoint members $X, X' \in \mathcal{X}$ then $\varrho(X) + \varrho(X') < \mu(G)$ would a contradict the definition of $\mu(G)$. Therefore, by the Helly-property of the in-solid sets, there exists a node $t \in \cap\mathcal{X}$. This node satisfies the requirements of the theorem, since if there was a non-empty $Z \subseteq V - t$ with $\varrho(Z) < \frac{\mu(G)}{2}$, then Z would necessarily contain an in-solid set $Z' \subseteq Z$ with $\varrho(Z') < \varrho(Z)$ (this follows from the definition of in-solid sets), contradicting the choice of t. □

In a digraph $G = (V, A)$, a node $t \in V$ with the property $\varrho(Z) \geq \frac{\mu(G)}{2}$ for every non-empty $Z \subseteq V - t$ will be called an *anchor node* of G.

4.3 A polynomial-time algorithm

In this section we present a polynomial time algorithm to determine the robustness of nice arborescences, which also implies a polynomial time algorithm to determine the robustness of minimum cost arborescences. A sketch of the algorithm goes as follows. For a minimal member F of $\mathcal{L}$, we apply Theorem 4.11, and find its anchor node t_F. We replace the tail of every arc leaving F by t_F, remove F from $\mathcal{L}$, and repeat until $\mathcal{L}$ goes empty. This way we construct a sequence of digraphs on the same node set: let D' be the last member of this sequence. Then for any $t \in \cup\mathcal{L}$ we construct another digraph D_t from D': for every $F \in \mathcal{L}$ with $t \in F$ and every arc of D' leaving F we replace the tail of this arc with t. Then we determine minimum double cuts in $D'[F]$ for every $F \in \mathcal{L} \cup \{V\}$, and we also determine minimum double cuts in $D_t[F]$ for every $F \in \mathcal{L} \cup \{V\}$ with $t \in F$: this way we have determined $O(n^2)$ double cuts altogether. Each of these double cuts also determines an $\mathcal{L}$-double cut in D, and we pick the one with the smallest cardinality, to claim that it actually is optimal.

Algorithm COVERING_NICE_ARBORESCENCES
begin

INPUT A digraph $D = (V, A)$ and a laminar family $\mathcal{L} \subseteq 2^V$
OUTPUT $\gamma(D, \mathcal{L})$

1.1. Let $D_0 = D$, $i = 1$ and $\mathcal{L}' = \mathcal{L}$.
1.2. While $\mathcal{L}' \neq \emptyset$ do
1.3. Choose an inclusionwise minimal set $F \in \mathcal{L}'$
1.4. Let $t_F \in F$ be such that $\varrho_{D_{i-1}[F]}(Z) \geq \frac{\mu(D_{i-1}[F])}{2}$ for every non-empty $Z \subseteq F - t_F$ (this node exists by Thm 4.11 applied to $G = D_{i-1}[F]$)
1.5. D_i is obtained from D_{i-1} by changing the tail of every arc leaving F to t_F
1.6. Let $i = i + 1$ and $\mathcal{L}' = \mathcal{L}' - F$
1.7. Let $D' = D_{|\mathcal{L}|}$.
1.8. For every $t \in \cup\mathcal{L}$
1.9. Let D_t be obtained from D' by changing the tail of every arc leaving a set $F \in \mathcal{L}$ with $t \in F$ to t
1.10. Return $\min\{\min\{\mu(D'[F]) : F \in \mathcal{L} \cup \{V\}\}, \min\{\mu(D_t[F] : t \in \cup\mathcal{L}, F \in \mathcal{L} \cup \{V\}, t \in F\}\}$.

end

The algorithm above is formulated in a way that it returns the optimum $\gamma(D, \mathcal{L})$ in question, but by the correspondance between the arc set of D and that of D' and D_t in the algorithm, clearly we can also return the optimal arc set, too. It is also clear that the algorithm can be formulated to run in strongly polynomial time for Problem 2, too: we only need to modify the definition of $\mu(G)$ and the tail-relocation operation in a natural way such that the weights are taken into account.

Theorem 4.12. *The Algorithm COVERING_NICE_ARBORESCENCES returns a correct answer.*

Proof. First of all, since $\Theta_{F,D} = \Theta_{F,D'} = \Theta_{F,D_t}$ for any $F \in \mathcal{L} \cup \{V\}$ and $t \in F \cap \cup\mathcal{L}$, and $\Theta_{F,D'} \leq \mu(D'[F])$ and $\Theta_{F,D_t} \leq \mu(D_t[F])$, the algorithm returns an upper bound for the optimum $\gamma(D, \mathcal{L})$ in question by Theorem 4.6.

On the other hand, assume that F is an inclusionwise minimal member of $\mathcal{L} \cup \{V\}$ such that the optimum $\gamma(D, \mathcal{L}) = \Theta_{F,D}$ (such a set exists again by Theorem 4.6). Assume furthermore that the non-empty disjoint sets $Z_1, Z_2 \subseteq F$ are such that $\Theta_{F,D} = f_{D[F]}(Z_1) + f_{D[F]}(Z_2)$. The following sequence of observations proves the theorem.

1. First observe, that any member $F' \in \mathcal{L}$ which is a proper subset of F can intersect at most one of Z_1 and Z_2. Assume the contrary, and note that $f_{D[F]}(Z_i) \geq f_{D[F']}(Z_i \cap F')$ holds for $i = 1, 2$, contradicting the minimal choice of F.

2. Next observe that there do not exist two disjoint members $F', F'' \in \mathcal{L}_{Z_1 \cup Z_2}$ that are proper subsets of F such that $t_{F'}$ and $t_{F''}$ are both outside $Z_1 \cup Z_2$. To see this assume again the contrary and let F', F'' be two inclusionwise minimal such sets. By exchanging the roles of Z_1 and Z_2 or the roles of F' and F'' we arrive at the following two cases: either both F' and F'' intersect Z_1, or F' intersects Z_1 and F'' intersects Z_2. The proof is analogous for both cases. Assume first that both F' and F'' intersect Z_1. Then we have

$$\begin{aligned}\gamma(D, \mathcal{L}) = \Theta_{F,D} = \Theta_{F,D'} = f_{D'[F]}(Z_1) + f_{D'[F]}(Z_2) \geq \\ \geq f_{D'[F]}(Z_1) \geq f_{D'[F']}(Z_1 \cap F') + f_{D'[F'']}(Z_1 \cap F'') = \\ = \varrho_{D'[F']}(Z_1 \cap F') + \varrho_{D'[F'']}(Z_1 \cap F'') \geq \\ \geq \frac{\mu(D'[F'])}{2} + \frac{\mu(D'[F''])}{2} > \gamma(D, \mathcal{L}),\end{aligned} \quad (7)$$

a contradiction. Here the second inequality follows from the definition of the function f, the equality following it is because $t_{F'''} \in Z_1$ if $F''' \in \mathcal{L}_{Z_1}$ is a proper subset of F' or F''. The next inequality follows from the definition of $t_{F'}$ and $t_{F''}$, and the last (strict) inequality is by the minimal choice of F.

In the other case, when F' intersects Z_1 and F'' intersects Z_2, we get the contradiction in a similar way:

$$\begin{aligned}\gamma(D, \mathcal{L}) = \Theta_{F,D} = \Theta_{F,D'} = f_{D'[F]}(Z_1) + f_{D'[F]}(Z_2) \geq \\ \geq f_{D'[F']}(Z_1 \cap F') + f_{D'[F'']}(Z_2 \cap F'') = \varrho_{D'[F']}(Z_1 \cap F') + \varrho_{D'[F'']}(Z_2 \cap F'') \geq \\ \geq \frac{\mu(D'[F'])}{2} + \frac{\mu(D'[F''])}{2} > \gamma(D, \mathcal{L}).\end{aligned} \quad (8)$$

3. Therefore we are left with two cases. In the first case assume that $t_{F'} \in Z_1 \cup Z_2$ for any $F' \in \mathcal{L}_{Z_1 \cup Z_2}$ that is proper subsets of F. In that case we have that $f_{D'[F]}(Z_i) = \varrho_{D'[F]}(Z_i)$ for both $i = 1, 2$, and thus $\gamma(D, \mathcal{L}) = \Theta_{F,D'} = \sum_{i=1,2} \varrho_{D'[F]}(Z_i) \geq \mu(D'[F]) \geq \Theta_{F,D'}$.

4. In our last case there exists a unique inclusionwise minimal $F' \in \mathcal{L}_{Z_1 \cup Z_2}$ such that F' is proper subsets of F and $t_{F'} \notin Z_1 \cup Z_2$. Assume without loss of generality that F' intersects Z_1 and choose an arbitrary $t \in F' \cap Z_1$. Then $f_{D_t[F]}(Z_i) = \varrho_{D_t[F]}(Z_i)$ for both $i = 1, 2$, and thus $\gamma(D, \mathcal{L}) = \Theta_{F,D_t} = \sum_{i=1,2} \varrho_{D_t[F]}(Z_i) \geq \mu(D_t[F]) \geq \Theta_{F,D_t}$.

□

5 Acknowledgements

We thank Naoyuki Kamiyama for calling our attention to this problem at the 7th Hungarian-Japanese Symposium on Discrete Mathematics and Its Applications in Kyoto. We would like to thank Kristóf Bérczi, András Frank, Erika Kovács, Tamás Király and Zoltán Király of the Egerváry Research Group for useful discussions and remarks. The authors received a grant (no. CK 80124) from the National Development Agency of Hungary, based on a source from the Research and Technology Innovation Fund.

References

[1] Mihály Bárász, Johanna Becker, and András Frank, *An algorithm for source location in directed graphs*, Oper. Res. Lett. **33** (2005), no. 3, 221–230.

[2] D. R. Fulkerson, *Packing rooted directed cuts in a weighted directed graph*, Mathematical Programming **6** (1974), 1–13, 10.1007/BF01580218.

[3] The EGRES Group, *Covering minimum cost spanning trees*, EGRES QP-2011-08, `www.cs.elte.hu/egres`.

[4] Naoyuki Kamiyama, *Robustness of minimum cost arborescences*, ISAAC (Takao Asano, Shin-Ichi Nakano, Yoshio Okamoto, and Osamu Watanabe, eds.), Lecture Notes in Computer Science, vol. 7074, Springer, 2011, pp. 130–139.