Covering Directed Graphs by In-trees

Naoyuki KAMIYAMA[1] Naoki KATOH[2]

Department of Architecture and Architectural Engineering, Kyoto University,
Kyotodaigaku-Katsura, Nishikyo-ku, Kyoto, 615-8540, Japan.
E-mail: {is.kamiyama, naoki}@archi.kyoto-u.ac.jp

Abstract

Given a directed graph $D = (V, A)$ with a set of d specified vertices $S = \{s_1, \dots, s_d\} \subseteq V$ and a function $f\colon S \to \mathbb{Z}_+$ where $\mathbb{Z}_+$ denotes the set of non-negative integers, we consider the problem which asks whether there exist $\sum_{i=1}^{d} f(s_i)$ in-trees denoted by $T_{i,1}, T_{i,2}, \dots, T_{i,f(s_i)}$ for every $i = 1, \dots, d$ such that $T_{i,1}, \dots, T_{i,f(s_i)}$ are rooted at s_i, each $T_{i,j}$ spans vertices from which s_i is reachable and the union of all arc sets of $T_{i,j}$ for $i = 1, \dots, d$ and $j = 1, \dots, f(s_i)$ covers A. In this paper, we prove that such set of in-trees covering A can be found by using an algorithm for the weighted matroid intersection problem in time bounded by a polynomial in $\sum_{i=1}^{d} f(s_i)$ and the size of D. Furthermore, for the case where D is acyclic, we present another characterization of the existence of in-trees covering A, and then we prove that in-trees covering A can be computed more efficiently than the general case by finding maximum matchings in a series of bipartite graphs.

1 Introduction

The problem for covering a graph by subgraphs with specified properties (for example, trees or paths) is very important from practical and theoretical viewpoints and have been extensively studied. For example, Nagamochi and Okada [1] studied the problem for covering a set of vertices of a given undirected tree by subtrees, and Arkin et al. [2] studied the problem for covering a set of vertices or edges of a given undirected graph by subtrees or paths. These results were motivated by vehicle routing problems. Moreover, Even et al. [3] studied the covering problem motivated by nurse station location problems.

This paper studies the problem for covering a directed graph by rooted trees which is motivated by the following evacuation planning problem. Given a directed graph which models a city, vertices model intersections and buildings, and arcs model roads connecting these intersections and buildings. People exist not only at vertices but also along arcs. Suppose we have to give several evacuation instructions for evacuating all people to some safety place. In order to avoid disorderly confusion, it is desirable that one evacuation instruction gives a single evacuation path for each person and these paths do not cross each other. Thus, we want each evacuation instruction to become an in-tree rooted at some safety place. Moreover, the number of instructions for each safety place is bounded in proportion to a size of each safety place.

The above evacuation planning problem is formulated as the following covering problem defined on a directed graph. We are given a directed graph $D = (V, A, S, f)$ which consists of a vertex set V, an arc set A, a set of d specified vertices $S = \{s_1, \dots, s_d\} \subseteq V$ and a function $f\colon S \to \mathbb{Z}_+$ where $\mathbb{Z}_+$ denotes the set of non-negative integers. In the above evacuation planning problem, S corresponds to a set of safety places, and $f(s_i)$ represents the upper bound of the number of evacuation instructions for $s_i \in S$. For each $i = 1, \dots, d$, we define $V_D^i \subseteq V$ as the set of vertices in V from which s_i is reachable in D, and we define an in-tree rooted at s_i which spans V_D^i as a (D, s_i)*-in-tree*. We define a set $\mathcal{T}$ of $\sum_{i=1}^{d} f(s_i)$ subgraphs of D as a D*-canonical set of in-trees* if $\mathcal{T}$ contains exactly $f(s_i)$ (D, s_i)-in-trees for every $i = 1, \dots, d$. If every two distinct in-trees of a

[1]Supported by JSPS Research Fellowships for Young Scientists.

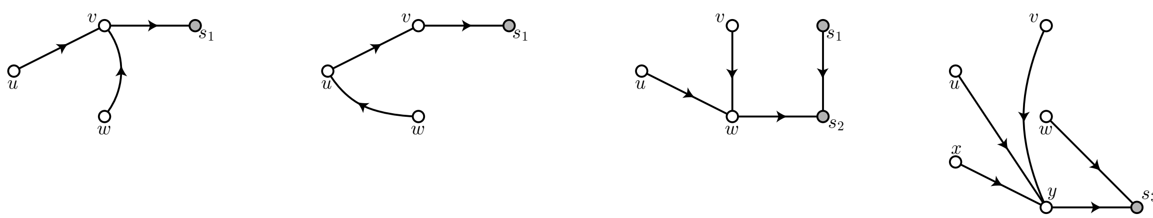
[2]Supported by the project *New Horizons in Computing*, Grant-in-Aid for Scientific Research on Priority Areas, MEXT Japan.

D-canonical set $\mathcal{T}$ of in-trees are arc-disjoint, we call $\mathcal{T}$ a *D-canonical set of arc-disjoint in-trees*. Furthermore, if the union of arc sets of all in-trees of a D-canonical set $\mathcal{T}$ of in-trees is equal to A, we say that $\mathcal{T}$ *covers* A.

Four in-trees illustrated in Figure 2 compose a D-canonical set $\mathcal{T}$ of in-trees which covers the arc set of a directed graph $D = (V, A, S, f)$ illustrated in Figure 1(a) where $S = \{s_1, s_2, s_3\}$, $f(s_1) = 2$, $f(s_2) = 1$ and $f(s_3) = 1$. However, $\mathcal{T}$ is not a D-canonical set of arc-disjoint in-trees.

(a) (b)

Figure 1: (a) Directed graph D. (b) Transformed graph D^*.



(a) (b) (c) (d)

Figure 2: (a) (D, s_1)-in-tree. (b) (D, s_1)-in-tree. (c) (D, s_2)-in-tree. (d) (D, s_3)-in-tree.

We will study the problem for *covering directed graphs by in-trees* (in short CDGI), and we will present characterizations for a directed graph $D = (V, A, S, f)$ for which there exists a feasible solution of CDGI(D), and an algorithm for CDGI(D).

Problem:	CDGI(D)
Input:	a directed graph D;
Output:	a D-canonical set of in-trees which covers the arc set of D, if one exists.

A special class of the problem CDGI(D) in which S consists of a single vertex was considered by Vidyasankar [4]. He showed the necessary and sufficient condition in terms of linear inequalities that there exists a feasible solution of this problem (a weaker version was shown by Frank [5]). However, to the best of our knowledge, an algorithm for CDGI(D) was not presented.

Our results: We first show that CDGI(D) can be viewed as some type of the connectivity augmentation problem. After this, we will prove that this connectivity augmentation problem can be solved by using an algorithm for the weighted matroid intersection problem in time bounded by a polynomial in $\sum_{i=1}^{d} f(s_i)$ and the size of D (this generalizes the result by Frank [6]). Furthermore, for the case where D is acyclic, we show another characterization for D that there exists a feasible solution of CDGI(D). Moreover, we prove that in this case CDGI(D) can be solved more efficiently than the general case by finding maximum matchings in a series of bipartite graphs instead of using an algorithm for the weighted matroid intersection problem.

Outline: The rest of this paper is organized as follows. Section 2 gives necessary definitions and fundamental results. In Section 3, we give an algorithm for the problem CDGI by using an algorithm for the weighted matroid intersection problem. In Section 4, we consider the acyclic case.

2 Preliminaries

Let $D = (V, A, S, f)$ be a connected directed graph which may have multiple arcs. Let $S = \{s_1, \ldots, s_d\}$. For $B \subseteq A$, let $\partial^-(B)$ (resp. $\partial^+(B)$) be a set of tails (resp. heads) of arcs in B. For $e \in A$, we write $\partial^-(e)$ and $\partial^+(e)$ instead of $\partial^-(\{e\})$ and $\partial^+(\{e\})$, respectively. For $W \subseteq V$, we define $\delta_D(W) = \{e \in A \colon \partial^-(e) \in W, \partial^+(e) \notin W\}$. For $v \in V$, we write $\delta_D(v)$ instead of $\delta_D(\{v\})$. For two distinct vertices $u, v \in D$, we denote by $\lambda(u, v; D)$ the local arc connectivity from u to v in D, i.e., $\lambda(u, v; D) = \min\{|\delta_D(W)| \colon u \in W, v \notin W, W \subseteq V\}$. We call a subgraph T of D *forest* if T has no cycle when we ignore the direction of arcs in T. If a forest T is connected, we call T *tree*. If every arc of an arc set B is parallel to some arc in A, we say that B is *parallel* to A. We denote a directed graph obtained by adding an arc set B to A by $D + B$, i.e., $D + B = (V, A \cup B, S, f)$. For $S' \subseteq S$, let $f(S') = \sum_{s_i \in S'} f(s_i)$. For $v \in V$, we denote by $R_D(v)$ a set of vertices in S which are reachable from v in D. For $W \subseteq V$, let $R_D(W) = \bigcup_{v \in W} R_D(v)$.

For an arc set B which is parallel to A, we clearly have for every $v \in V$

$$R_D(v) = R_{D+B}(v). \tag{1}$$

From (1), we have for every $i = 1, \ldots, d$

$$V_D^i = V_{D+B}^i. \tag{2}$$

We define D^* as a directed graph obtained from D by adding a new vertex s^* and connecting s_i to s^* with $f(s_i)$ parallel arcs for every $i = 1, \ldots, d$ (see Figure 1). We denote by A^* the arc set of D^*. From the definition of D^*,

$$|A^*| = \textstyle\sum_{v \in V} |\delta_{D^*}(v)| = |A| + f(S). \tag{3}$$

We say that D is *(S, f)-proper* when $|\delta_{D^*}(v)| \leq f(R_D(v))$ holds for every $v \in V$.

2.1 Rooted arc-connectivity augmentation by reinforcing arcs

Given a directed graph $D = (V, A, S, f)$, we call an arc set B with $A \cap B = \emptyset$ which is parallel to A a *D^*-rooted connector* if $\lambda(v, s^*; D^* + B) \geq f(R_D(v))$ holds for every $v \in V$. Notice that since a D^*-rooted connector B is parallel to A, B does not contain an arc which is parallel to an arc entering into s^* in D^*. Then, the problem *rooted arc-connectivity augmentation by reinforcing arcs* (in short RAA-RA) is formally defined as follows.

Problem:	RAA-RA(D^*)
Input:	D^* of a directed graph D;
Output:	a D^*-rooted connector B whose size is minimum among all D^*-rooted connectors.

Notice that the problem RAA-RA(D^*) is not equivalent to the local arc-connectivity augmentation problem with minimum number of reinforcing arcs from $v \in V$ to $s_i \in R_D(v)$. For example, we consider D^* illustrated in Figure 3(a) of a directed graph $D = (V, A, S, f)$ where $S = \{s_1, s_2\}$, $f(s_1) = 2$ and $f(s_2) = 2$. The broken lines in Figure 3(b) represent a minimum D^*-rooted connector. For the problem that asks to increase the v-s_i local arc-connectivity for every $v \in V$ and $s_i \in R_D(v)$ to $f(s_i)$ by adding minimum parallel arcs to A (this problem is called the problem *increasing arc-connectivity by reinforcing arcs* in [7], in short IARA(D^*)), an optimal solution is a set of broken lines in Figure 3(c). While it is known [7] that IARA(D^*) is $\mathcal{NP}$-hard, it is known [6] that RAA-RA(D^*) in which S consists of a single element can be solved in time bounded by a polynomial in $f(S)$ and the size of D by using an algorithm for the weighted matroid intersection.

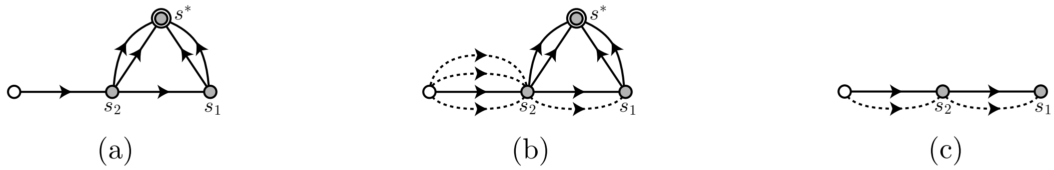


Figure 3: (a) Input. (b) Optimal solution for RAA-RA. (c) Optimal solution for IARA.

2.2 Matroids on arc sets of directed graphs

In this subsection, we define two matroids $\boldsymbol{M}(D^*)$ and $\boldsymbol{U}(D^*)$ on A^* for a directed graph $D = (V, A, S, f)$, which will be used in the subsequent discussion. We denote by $\boldsymbol{M} = (E, \mathcal{I})$ a matroid on E whose collection of independent sets is $\mathcal{I}$. Introductory treatment of a matroid is given in [8].

For $i = 1, \ldots, d$ and $j = 1, \ldots, f(s_i)$, we define $\boldsymbol{M}_{i,j}(D^*) = (A^*, \mathcal{I}_{i,j}(D^*))$ where $I \subseteq A^*$ belongs to $\mathcal{I}_{i,j}(D^*)$ if and only if both of a tail and a head of every arc in I are contained in $V_D^i \cup \{s^*\}$ and a directed graph $(V_D^i \cup \{s^*\}, I)$ is a forest. $\boldsymbol{M}_{i,j}(D^*)$ is clearly a matroid (i.e. graphic matroid). Moreover, we denote the union of $\boldsymbol{M}_{i,j}(D^*)$ for $i = 1, \ldots, d$ and $j = 1, \ldots, f(s_i)$ by $\boldsymbol{M}(D^*) = (A^*, \mathcal{I}(D^*))$ in which $I \subseteq A^*$ belongs to $\mathcal{I}(D^*)$ if and only if I can be partitioned into $\{I_{i,1}, \ldots, I_{i,f(s_i)} \colon i = 1, \ldots, d\}$ such that each $I_{i,j}$ belongs to $\mathcal{I}_{i,j}(D^*)$. $\boldsymbol{M}(D^*)$ is also a matroid (see Chapter 12.3 in [8]. This matroid is also called *matroid sum*). When $I \in \mathcal{I}(D^*)$ can be partitioned into $\{I_{i,1}, \ldots, I_{i,f(s_i)} \colon i = 1, \ldots, d\}$ such that a directed graph $(V_D^i \cup \{s^*\}, I_{i,j})$ is a tree for every $i = 1, \ldots, d$ and $j = 1, \ldots, f(s_i)$, we call I a *base of* $\boldsymbol{M}(D^*)$.

Next we define another matroid. We define $\boldsymbol{U}(D^*) = (A^*, \mathcal{J}(D^*))$ where $I \subseteq A^*$ belongs to $\mathcal{J}(D^*)$ if and only if I satisfies

$$|\delta_{D^*}(v) \cap I| \leq \begin{cases} f(R_D(v)), & \text{if } v \in V, \\ 0, & \text{if } v = s^*. \end{cases} \tag{4}$$

Since $\boldsymbol{U}(D^*)$ is a direct sum of uniform matroids, $\boldsymbol{U}(D^*)$ is also a matroid (see Exercise 7 of pp.16 and Example 1.2.7 in [8]). We call $I \in \mathcal{J}(D^*)$ a *base of* $\boldsymbol{U}(D)$ when (4) holds with equality.

For two matroids $\boldsymbol{M}(D^*)$ and $\boldsymbol{U}(D^*)$, we call an arc set $I \subseteq A^*$ D^*-*intersection* when $I \in \mathcal{I}(D^*) \cap \mathcal{J}(D^*)$. If a D^*-intersection I is a base of both $\boldsymbol{M}(D^*)$ and $\boldsymbol{U}(D^*)$, we call I *complete*.

When we are given a weight function $w \colon A^* \to \mathbb{R}_+$ where $\mathbb{R}_+$ denotes the set of non-negative reals, we define the weight of $I \subseteq A^*$ (denoted by $w(I)$) by the sum of weights of all arcs I. The *weighted matroid intersection problem* (in short WMI) is then defined as follows [9].

Problem:	WMI(D^*)
Input:	D^* of a directed graph D and a weight function $w \colon A^* \to \mathbb{R}_+$;
Output:	a complete D^*-intersection I whose weigh is minimum among all complete D^*-intersections, if one exists.

Lemma 2.1 *We can solve WMI(D^*) in $O(M|A^*|^6)$ time where $M = \sum_{v \in V} f(R_D(v))$.*

Proof. See Appendix A. ◻

2.3 Results from [10]

In this section, we introduce results concerning packing of in-trees given by Kamiyama et al. [10] which plays a crucial role in this paper.

Theorem 2.2 ([10]) *Given a directed graph $D = (V, A, S, f)$, the following three statements are equivalent: (i) For every $v \in V$, $\lambda(v, s^*; D^*) \geq f(R_D(v))$ holds. (ii) There exists a D-canonical set of arc-disjoint in-trees. (iii) There exists a complete D^*-intersection.*

Although the following theorem is not explicitly proved in [10], we can easily obtain it from the proof of Theorem 2.2 in [10].

Theorem 2.3 ([10]) *Given a directed graph $D = (V, A, S, f)$ which satisfies the condition of Theorem 2.2, we can find a D-canonical set of arc-disjoint in-trees in $O(M^2|A|^2)$ time where $M = \sum_{v \in V} f(R_D(v))$.*

From Theorem 2.2, we obtain the following corollary.

Corollary 2.4 *Given a directed graph $D = (V, A, S, f)$ and an arc set B with $A \cap B = \emptyset$ which is parallel to A, the following three statements are equivalent: (i) B is a D^*-rooted connector. (ii) There exists a $(D + B)$-canonical set of arc-disjoint in-trees. (iii) There exists a complete $(D + B)^*$-intersection.*

Proof. The equivalence of (ii) and (iii) follows from Theorem 2.2.
(i)→(ii): Since B is parallel to A, we clearly have

$$(D + B)^* = D^* + B. \tag{5}$$

Since B is a D^*-rooted connector and from (5) and (1), we have for every $v \in V$

$$\lambda(v, s^*; (D + B)^*) = \lambda(v, s^*; D^* + B) \geq f(R_D(v)) = f(R_{D+B}(v)).$$

From this inequality and Theorem 2.2, this part follows.
(ii)→(i): Since there exists a $(D + B)$-canonical set of arc-disjoint in-trees and from (5), Theorem 2.2 and (1), we have for every $v \in V$

$$\lambda(v, s^*; D^* + B) = \lambda(v, s^*; (D + B)^*) \geq f(R_{D+B}(v)) = f(R_D(v)).$$

This proves that B is a D^*-rooted connector. ◻

3 An Algorithm for Covering by In-trees

Given a directed graph $D = (V, A, S, f)$, we present in this section an algorithm for CDGI(D). The time complexity of the proposed algorithm is bounded by a polynomial in $f(S)$ and the size of D. We first prove that CDGI(D) can be reduced to RAA-RA(D^*). After this, we show that RAA-RA(D^*) can be solved by using an algorithm for the weighted matroid intersection problem.

3.1 Reduction from CDGI to RAA-RA

If $D = (V, A, S, f)$ is not (S, f)-proper, i.e., $|\delta_{D^*}(v)| > f(R_D(v))$ for some $v \in V$, there exists no feasible solution of CDGI(D) since there can not be a D-canonical set of in-trees that covers $\delta_{D^*}(v)$ from the definition of a D-canonical set of in-trees. Thus, we assume in the subsequent discussion that D is (S, f)-proper.

Proposition 3.1 *Given an (S, f)-proper directed graph $D = (V, A, S, f)$, the size of a D^*-rooted connector is at least $\sum_{v \in V} f(R_D(v)) - (|A| + f(S))$.*

Proof. See Appendix B. ◻

For an (S, f)-proper directed graph $D = (V, A, S, f)$, we define opt_D by

$$\mathsf{opt}_D = \textstyle\sum_{v \in V} f(R_D(v)) - (|A| + f(S)). \tag{6}$$

From Proposition 3.1, the size of a D^*-rooted connector is at least opt_D.

Lemma 3.2 *Given an (S, f)-proper directed graph $D = (V, A, S, f)$, there exists a feasible solution of CDGI(D) if and only if there exists a D^*-rooted connector whose size is equal to opt_D.*

Proof. **Only if-part**: Suppose there exists a feasible solution of CDGI(D), i.e., there exists a D-canonical set $\mathcal{T}$ of in-trees which covers A. For each $i = 1, \ldots, d$, we denote $f(s_i)$ (D, s_i)-in-trees of $\mathcal{T}$ by $T_{i,1}, \ldots, T_{i,f(s_i)}$. For each $e \in A$, let $P_e = \{(i, j)\colon e \text{ is contained in } T_{i,j}\}$. Since $\mathcal{T}$ covers A, each $e \in A$ is contained in at least one in-tree of $\mathcal{T}$. Thus, $|P_e| \geq 1$ holds for every $e \in A$. We define an arc set B by $B = \bigcup_{e \in A} \{|P_e| - 1 \text{ copies of } e\}$. We will prove that B is a D^*-rooted connector whose size is equal to opt_D.

We first prove $|B| = \mathsf{opt}_D$. For this, we show that for every $v \in V$

$$\textstyle\sum_{e \in \delta_D(v)} (|P_e| - 1) = f(R_D(v)) - |\delta_{D^*}(v)|. \tag{7}$$

Let us first consider $v \notin S$. For $s_i \in R_D(v)$, $T_{i,j}$ contains v since $T_{i,j}$ spans V_D^i and s_i is reachable from v. Hence, since $T_{i,j}$ is an in-tree and v is not a root of $T_{i,j}$ from $v \notin S$, $T_{i,j}$ contains exactly one arc $e \in \delta_D(v)$, i.e., (i, j) is contained in P_e for exactly one arc $e \in \delta_D(v)$. Thus, $\sum_{e \in \delta_D(v)} |P_e| = \sum_{s_i \in R_D(v)} f(s_i) = f(R_D(v))$. From this equation and since $|\delta_D(v)| = |\delta_{D^*}(v)|$ follows from $v \notin S$, (7) holds. In the case of $v \in S$, for $s_i \in R_D(v) \setminus \{v\}$, (i, j) is contained in P_e for exactly one arc $e \in \delta_D(v)$ as in the case of $v \notin S$. Thus, $\sum_{e \in \delta_D(v)} |P_e| = f(R_D(v)) - f(v)$. From this equation and $|\delta_{D^*}(v)| = |\delta_D(v)| + f(v)$,

$$\textstyle\sum_{e \in \delta_D(v)} (|P_e| - 1) = f(R_D(v)) - f(v) - |\delta_D(v)| = f(R_D(v)) - |\delta_{D^*}(v)|.$$

This completes the proof of (7). Since B contains $|P_e| - 1$ copies of $e \in A$,

$$\begin{aligned} |B| &= \textstyle\sum_{v \in V} \sum_{e \in \delta_D(v)} (|P_e| - 1) = \sum_{v \in V} (f(R_D(v)) - |\delta_{D^*}(v)|) \quad \text{(from (7))} \\ &= \mathsf{opt}_D \quad \text{(from (3) and (6))}. \end{aligned}$$

What remains is to prove that B is a D^*-rooted connector. From Corollary 2.4, it is sufficient to prove that there exists a $(D+B)$-canonical set of arc-disjoint in-trees. For this, we will construct from $\mathcal{T}$ a set $\mathcal{T}'$ of arc-disjoint in-trees which consists of $T'_{i,1}, \ldots, T'_{i,f(s_i)}$ for $i = 1, \ldots, d$, and we prove that $\mathcal{T}'$ is a $(D + B)$-canonical set of in-trees. Each $T'_{i,j}$ is constructed from $T_{i,j}$ as follows. When $e \in A$ is contained in more than one in-tree of $\mathcal{T}$, in order to construct $\mathcal{T}'$ from $\mathcal{T}$, we need to replace e of $T_{i,j}$ by an arc in B which is parallel to e for every $(i, j) \in P_e$ except one in-tree. For $(i_{\min}, j_{\min}) \in P_e$ which is lexicographically smallest in P_e, we allow $T'_{i_{\min}, j_{\min}}$ to use e, while for $(i, j) \in P_e \setminus (i_{\min}, j_{\min})$, we replace e of $T_{i,j}$ by an arc in B which is parallel to e so that for distinct $(i, j), (i', j') \in P_e \setminus (i_{\min}, j_{\min})$, the resulting $T'_{i,j}$ and $T'_{i',j'}$ contain distinct arcs which are parallel to e, respectively (see Figure 4).

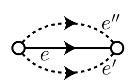


Figure 4: Illustration of the replacing operation. Let e be an arc in A, and let e', e'' be arcs in B. Assume that $P_e = \{(1, 1), (1, 2), (2, 1)\}$. In this case, $T_{1,1}$, $T_{1,2}$ and $T_{2,1}$ contain e. Then, $T'_{1,1}$ contains e, $T'_{1,2}$ contains e', and $T'_{2,1}$ contains e''.

We will do this operation for every $e \in A$. Let $\mathcal{T}'$ be the set of in-trees obtained by performing the above operation for every $e \in A$. Here we show that $\mathcal{T}'$ is a $(D+B)$-canonical set of arc-disjoint in-trees. Since $T'_{i,j}$ and $T'_{i',j'}$ are arc-disjoint for $(i, j) \neq (i', j')$ from the way of constructing $\mathcal{T}'$, it is sufficient to prove that $T'_{i,j}$ is a $(D + B, s_i)$-in-tree. Since $T'_{i,j}$ is constructed by replacing arcs of $T_{i,j}$ by the corresponding parallel arc in B and $T_{i,j}$ is an in-tree rooted at s_i, $T'_{i,j}$ is also an in-tree rooted at s_i. Since $T_{i,j}$ spans V_D^i and from (2), $T'_{i,j}$ spans V_{D+B}^i. Hence, $T'_{i,j}$ is a $(D+B, s_i)$-in-tree. This completes the proof.

If-part: Let B be a D^*-rooted connector with $|B| = \mathsf{opt}_D$. From Corollary 2.4, there exists a $(D+B)$-canonical set $\mathcal{T}'$ of arc-disjoint in-trees. For each $i = 1, \ldots, d$, we denote $f(s_i)$ $(D+B, s_i)$-in-trees of $\mathcal{T}'$ by $T'_{i,1}, \ldots, T'_{i,f(s_i)}$. We will prove that we can construct from $\mathcal{T}'$ a D-canonical set of in-trees covering A. We first construct from $\mathcal{T}'$ a set $\mathcal{T}$ of in-trees which consists of $T_{i,j}$ for $i = 1, \ldots, d$ and $j = 1, \ldots, f(s_i)$ by the following procedure Replace.

Procedure Replace: For each $i = 1, \ldots, d$ and $j = 1, \ldots, f(s_i)$, set $T_{i,j}$ to be a directed graph obtained from $T'_{i,j}$ by replacing every arc $e \in B$ which is contained in $T'_{i,j}$ by an arc in A which is parallel to e.

From now on, we prove that $\mathcal{T}$ is a D-canonical set of in-trees which covers A. It is not difficult to prove that $\mathcal{T}$ is a D-canonical set of in-trees from the definition of the procedure Replace in the same manner as the last part of the proof of the "only if-part". Thus, it is sufficient to prove that $\mathcal{T}$ covers A. For this, we first show that $\mathcal{T}'$ covers $A \cup B$. From $A \cap B = \emptyset$, $|B| = \mathsf{opt}_D$ and (6),

$$|A \cup B| = |A| + \mathsf{opt}_D = \textstyle\sum_{v \in V} f(R_D(v)) - f(S). \tag{8}$$

Recall that each $v \in V$ is contained in $f(R_{D+B}(v))$ in-trees of $\mathcal{T}'$ from the definition of a $(D+B)$-canonical set of in-trees. Thus, since in-trees of $\mathcal{T}'$ are arc-disjoint, it holds for each $v \in V$ that the number of arcs in $\delta_{D+B}(v)$ which are contained in in-trees of $\mathcal{T}'$ is equal to (i) $f(R_{D+B}(v))$ if $v \in V \setminus S$, or (ii) $f(R_{D+B}(v)) - f(v)$ if $v \in S$. Hence, the number of arcs in $A \cup B$ contained in in-trees of $\mathcal{T}'$ is equal to

$$\begin{aligned}&\textstyle\sum_{v \in V \setminus S} f(R_{D+B}(v)) + \sum_{v \in S}(f(R_{D+B}(v)) - f(v)) \\ &= \textstyle\sum_{v \in V} f(R_{D+B}(v)) - f(S) = \sum_{v \in V} f(R_D(v)) - f(S) \quad \text{(from (1))}.\end{aligned} \tag{9}$$

Since any arc of $\mathcal{T}'$ is in $A \cup B$ and the number of arcs in $A \cup B$ is equal to that of $\mathcal{T}'$ from (8) and (9), $\mathcal{T}'$ contains all arcs in A. Thus, $\mathcal{T}$ covers A from the definition of the procedure Replace. $\square$

As seen in the proof of the "if-part" of Lemma 3.2, if we can find a D^*-rooted connector B with $|B| = \mathsf{opt}_D$, we can compute a D-canonical set of in-trees which covers A by using the procedure Replace from a $(D+B)$-canonical set of arc-disjoint in-trees. Furthermore, we can construct a $(D+B)$-canonical set of arc-disjoint in-trees by using the algorithm of Theorem 2.3. Since the optimal value of RAA-RA(D^*) is at least opt_D from Proposition 3.1, we can test if there exists a D^*-rooted connector whose size is equal to opt_D by solving RAA-RA(D^*). Assuming that we can solve RAA-RA(D^*), our algorithm for finding a D-canonical set of in-trees which covers A called Algorithm CR can be illustrated as Algorithm 1 below.

Lemma 3.3 *Given a directed graph $D = (V, A, f, S)$, Algorithm* CR *correctly finds a D-canonical set of in-trees which covers A in $O(\gamma_1 + |V||A| + M^4)$ time if one exists where γ_1 is the time required to solve RAA-RA(D^*) and $M = \sum_{v \in V} f(R(v))$.*

Proof. See Appendix C. $\square$

3.2 Reduction from RAA-RA to WMI

From the algorithm CR in Section 3.1, in order to present an algorithm for CDGI(D), what remains is to show how we solve RAA-RA(D^*). In this section, we will prove that we can test whether there exists a D^*-rooted connector whose size is equal to opt_D (i.e., Steps 4 and 5 in the algorithm CR)

Algorithm 1 Algorithm CR

Input: a directed graph $D = (V, A, S, f)$
Output: a D-canonical set of in-trees covering A, if one exists

```
if D is not (S, f)-proper then
    Halt (there exists no D-canonical set of in-trees covering A)
end if
Find an optimal solution B of RAA-RA(D*)
if |B| > opt_D then
    Halt (there exists no D-canonical set of in-trees covering A)
else
    Construct a (D + B)-canonical set T' of arc-disjoint in-trees
    Construct a set T of in-trees from T' by using the procedure Replace
    return T
end if
```

by reducing it to the problem WMI. Our proof is based on the algorithm of [6] for RAA-RA(D^*) in which S consists of a single vertex. We extend the idea of [6] to the case of $|S| > 1$ by using Theorem 2.2. We define a directed graph D_+ obtained from D by adding opt_D parallel arcs to every $e \in A$. Then, we will compute a D^*-rooted connector whose size is equal to opt_D by using an algorithm for WMI(D^*_+) as described below. Since the number of arcs in a D^*-rooted connector whose size is equal to opt_D which are parallel to one arc in A is at most opt_D, it is enough to add opt_D parallel arcs to each arc of A in D_+ in order to find a D^*-rooted connector whose size is equal to opt_D.

We denote by A_+ and A^*_+ the arc sets of D_+ and D^*_+, respectively. If $I \subseteq A^*_+$ is a complete D^*_+-intersection, since I is a base of $\boldsymbol{U}(D^*_+)$ and from (4) and (1),

$$|I| = \textstyle\sum_{v \in V} f(R_{D_+}(v)) = \sum_{v \in V} f(R_D(v)). \tag{10}$$

We define a weight function $w \colon A^*_+ \to \mathbb{R}_+$ by

$$w(e) = \begin{cases} 0, & \text{if } e \in A^*, \\ 1, & \text{otherwise.} \end{cases} \tag{11}$$

The following lemma shows the relation between RAA-RA(D^*) and WMI(D^*_+).

Lemma 3.4 *Given an (S, f)-proper directed graph $D = (V, A, S, f)$, there exists a D^*-rooted connector whose size is equal to opt_D if and only if there exists a complete D^*_+-intersection whose weight is equal to opt_D.*

To prove Lemma 3.4, we need to show the following two lemmas.

Lemma 3.5 *Given a directed graph $D = (V, A, S, f)$ and an arc set B which is parallel to A, (i) if there is a complete D^*-intersection I, I is also a complete $(D + B)^*$-intersection, and (ii) if there is a complete $(D + B)^*$-intersection I such that $I \subseteq A^*$, I is also a complete D^*-intersection.*

Proof. See Appendix D. □

Lemma 3.6 *Given D^*_+ of an (S, f)-proper directed graph $D = (V, A, S, f)$ and a weight function $w \colon A^*_+ \to \mathbb{R}_+$ defined by (11), if there exists a complete D^*_+-intersection $I \subseteq A^*_+$, $w(I) \geq \mathsf{opt}_D$. Moreover, $w(I) = \mathsf{opt}_D$ if and only if $A^* \subseteq I$.*

Proof. See Appendix E. ◻

Proof of Lemma 3.4. **Only if-part**: Assume that there exists a D^*-rooted connector whose size is equal to opt_D. Since D_+ has opt_D parallel arcs to every $e \in A$, there exists a D^*-rooted connector $B \subseteq A_+ \setminus A$ with $|B| = \mathsf{opt}_D$. Let us fix a D^*-rooted connector $B \subseteq A_+ \setminus A$ with $|B| = \mathsf{opt}_D$. From (i) of Lemma 3.5, in order to prove the "only if-part", it is sufficient to prove that there exists a complete $(D+B)^*$-intersection I with $w(I) = \mathsf{opt}_D$. Since there exists a complete $(D+B)^*$-intersection I from Corollary 2.4, we will prove that $w(I) = \mathsf{opt}_D$. Since the arc set of $(D+B)^*$ is equal to $A^* \cup B$ and I is a $(D+B)^*$-intersection, $I \subseteq A^* \cup B$ holds. Thus, since $w(A^* \cup B) = |B| = \mathsf{opt}_D$ follows from (11), $w(I) \leq w(A^* \cup B) = \mathsf{opt}_D$ holds. Hence, $w(I) = \mathsf{opt}_D$ follows from Lemma 3.6. This completes the proof.
If-part: Assume that there exists a complete D_+^*-intersection I with $w(I) = \mathsf{opt}_D$. Let B be $I \setminus A^*$, and we will prove that B is a D^*-rooted connector with $|B| = \mathsf{opt}_D$. We first prove B is a D^*-rooted connector by using (ii) of Lemma 3.5 and Corollary 2.4. We set B and D in Lemma 3.5 to be $A_+ \setminus (A \cup B)$ and $D + B$, respectively. Notice that $(D+B) + (A_+ \setminus (A \cup B)) = D_+$ follows from $B \subseteq A_+$ and $A_+ \setminus (A \cup B)$ is parallel to $A \cup B$. From $B = I \setminus A^*$, we have $I \subseteq A^* \cup B$. Thus, I is a complete $(D+B)^*$-intersection since I is a complete D_+^*-intersection and from (ii) of Lemma 3.5. Hence, from Corollary 2.4, B is a D^*-rooted connector.

What remains is to prove that $|B| = \mathsf{opt}_D$. From Lemma 3.6 and $w(I) = \mathsf{opt}_D$, $A^* \subseteq I$ holds. Thus, from $B = I \setminus A^*$ and (10), $|B| = |I \setminus A^*| = |I| - |A^*| = \sum_{v \in V} f(R_D(v)) - (|A| + f(S))$. This equation and (6) complete the proof. ◻

As seen in the proof of the "if-part" of Lemma 3.4, if we can find a complete D_+^*-intersection I with $w(I) = \mathsf{opt}_D$, we can find a D^*-rooted connector B with $|B| = \mathsf{opt}_D$ by setting $B = I \setminus A^*$. Furthermore, we can obtain a complete D_+^*-intersection whose weight is equal to opt_D if one exists by using the algorithm for WMI(D_+^*) since the optimal value of WMI(D_+^*) is at least opt_D from Lemma 3.6. The formal description of the algorithm called Algorithm RW for finding a D^*-rooted connector whose size is equal to opt_D is illustrated in Algorithm 2.

Algorithm 2 Algorithm RW

Input: D^* of an (S, f)-proper directed graph $D = (V, A, S, f)$
Output: a D^*-rooted connector whose size is equal to opt_D, if one exits

```
1: Find an optimal solution I of WMI(D*_+) with a weight function w defined by (11)
2: if there exists no solution of WMI(D*_+) or w(I) > opt_D then
3:     Halt (There exists no D*-rooted connector whose size is equal to opt_D)
4: end if
5: return I \ A*
```

Lemma 3.7 *Given D^* of an (S, f)-proper directed graph $D = (V, A, f, S)$, Algorithm* RW *correctly finds a D^*-rooted connector whose size is equal to opt_D in $O(\gamma_2 + M|A|)$ time if one exists where γ_2 is the time required to solve WMI(D_+^*) and $M = \sum_{v \in V} f(R_D(v))$.*

Proof. The lemma immediately follows from Lemma 3.4. ◻

3.3 Algorithm for CDGI

We are ready to explain the formal description of our algorithm called Algorithm Covering for CDGI(D). Algorithm Covering is the same as Algorithm CR such that Steps 4, 5 and 6 are replaced by Algorithm RW.

Theorem 3.8 *Given a directed graph $D = (V, A, S, f)$, Algorithm* Covering *correctly finds a D-canonical set of in-trees which covers A in $O(M^7|A|^6)$ time if one exits where $M = \sum_{v \in V} f(R_D(v))$.*

Proof. The correctness of the algorithm follows from Lemmas 3.3 and 3.7. We then consider the time complexity of this algorithm. From Lemmas 3.3 and 3.7, what remains is to analyze the time required to solve WMI(D^*_+). If D is (S, f)-proper, $|A^*| = \sum_{v \in V} |\delta_{D^*}(v)| \leq \sum_{v \in V} f(R_D(v)) = M$. Thus, since D^*_+ has opt_D parallel arcs of every $e \in A$, $|A^*_+| = |A^*| + \sum_{e \in A} \mathsf{opt}_D \leq M + M|A|$. Hence we have $|A^*_+| = O(M|A|)$. Thus, from Lemma 2.1, we can solve WMI(D^*) in $O(M^7|A|^6)$ time. From this discussion and Lemmas 3.3 and 3.7, we obtain the theorem. ◽

4 Acyclic Case

In this section, we show that in the case where $D = (V, A, S, f)$ is acyclic, a D-canonical set of in-trees covering A can be computed more efficiently than the general case. For this, we prove the following theorem.

Theorem 4.1 *Given an acyclic directed graph $D = (V, A, S, f)$, there exists a D-canonical set of in-trees which covers A if and only if*

$$|B| \leq f(R_D(\partial^+(B))) \text{ for every } v \in V \text{ and } B \subseteq \delta_D(v). \tag{12}$$

Sketch of Proof. For each $v \in V$, we define an undirected bipartite graph $G_v = (X_v \cup Y_v, E_v)$ which is necessary to prove the theorem. Let $X_v = \{x_e \colon e \in \delta_D(v)\}$ and $Y_v = \{y_{i,j} \colon s_i \in R_D(v), j = 1, \ldots, f(s_i)\}$. $x_e \in X_v$ and $y_{i,j} \in Y_v$ are connected by an edge in E_v if and only if s_i is reachable from $\partial^+(e)$ (see Figure 5).

(a) (b)

Figure 5: (a) Input acyclic directed graph D. (b) Bipartite graph G_u for u in (a).

It is well-known that (12) is equivalent to the necessary and sufficient condition that for any $v \in V$, there exists a matching in G_v which saturates vertices in X_v (e.g., Theorem 16.7 in Chapter 16 of [12]). Thus it is sufficient to prove that there exists a D-canonical set of in-trees which covers A if and only if for any $v \in V$, there exists a matching in G_v which saturates vertices in X_v. The proof is in Appendix F. ◽

From Theorem 4.1, instead of the algorithm presented in Section 3, we can more efficiently find a D-canonical set of in-trees covering A by finding a maximum matching in a bipartite graph $O(|V|)$ times. In regard to algorithms for finding a maximum matching in a bipartite graph, see e.g. [13].

Corollary 4.2 *Given an acyclic directed graph $D = (V, A, S, f)$, we can find a D-canonical set of in-trees which covers A in $O(\mathsf{match}(M + |A|, M|A|))$ time if one exists where $\mathsf{match}(n, m)$ represents the time required to find maximum matching in a bipartite graph with n vertices and m arcs and $M = \sum_{v \in V} f(R_D(v))$.*

Acknowledgement: We thank Prof. Tibor Jordán, who informed us of the paper [6].

References

[1] Nagamochi, H., Okada, K.: Approximating the minmax rooted-tree cover in a tree. Inf. Process. Lett. **104**(5) (2007) 173–178

[2] Arkin, E.M., Hassin, R., Levin, A.: Approximations for minimum and min-max vehicle routing problems. J. Algorithms **59**(1) (2006) 1–18

[3] Even, G., Garg, N., Könemann, J., Ravi, R., Sinha, A.: Min-max tree covers of graphs. Oper. Res. Lett. **32**(4) (2004) 309–315

[4] Vidyasankar, K.: Covering the edge set of a directed graph with trees. Discrete Mathmatics **24** (1978) 79–85

[5] Frank, A.: Covering branchings. Acta Scientiarum Mathematicarum [Szeged] **41** (1979) 77–81

[6] Frank, A.: Rooted k-connections in digraphs. Discrete Applied Mathmatics (to appear).

[7] Jordan, T.: Two $\mathcal{NP}$-complete augmentation problems. Technical Report 8, Department of Mathematics and Computer Science, Odense University (1997)

[8] Oxley, J.G.: Matroid theory. Oxford University Press (1992)

[9] Frank, A.: A weighted matroid intersection algorithm. J. Algorithms **2**(4) (1981) 328–336

[10] Kamiyama, N., Katoh, N., Takizawa, A.: Arc-disjoint in-trees in directed graphs. In: Proc. the nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA2008). (2008) 518–526

[11] Knuth, D.: Matroid partitioning. Technical Report STAN-CS-73-342, Computer Science Department, Stanford University (1974)

[12] Schrijver, A.: Combinatorial Optimization: Polyhedra and Efficiency (Algorithms and Combinatorics). Springer-Verlag (2003)

[13] Hopcroft, J.E., Karp, R.M.: An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. SIAM J. Comput. **2**(4) (1973) 225–231

A Proof of Lemma 2.1

To prove the lemma, we use the following theorem concerning a matroid.

Theorem A.1 ([11]) *Given a matroid $\boldsymbol{M} = (E, \mathcal{I})$ which is a union of t ($\leq |E|$) matroids $\boldsymbol{M}_1 = (E, \mathcal{I}_1), \ldots, \boldsymbol{M}_t = (E, \mathcal{I}_t)$, we can test if a given set belongs to $\mathcal{I}$ in $O(|E|^3\gamma)$ time where γ is the time required to test if a given set belongs to $\mathcal{I}_1, \ldots, \mathcal{I}_t$.*

Theorem A.2 ([9]) *Given two matroids $\boldsymbol{M}_1 = (E, \mathcal{I}_1)$ and $\boldsymbol{M}_2 = (E, \mathcal{I}_2)$ with a weight function $w \colon E \to \mathbb{R}_+$ and a non-negative integer $k \in \mathbb{Z}_+$, we can find $I \in \mathcal{I}_1 \cap \mathcal{I}_2$ with $|I| = k$ whose weight is minimum among all $I' \in \mathcal{I}_1 \cap \mathcal{I}_2$ with $|I'| = k$ in $O(k|E|^3 + k|E|^2\gamma)$ time if one exists where γ is the time required to test if a given set belongs to both $\mathcal{I}_1$ and $\mathcal{I}_2$.*

We consider the time required to test if a given set belongs to both $\mathcal{I}(D^*)$ and $\mathcal{J}(D^*)$. Since it is not difficult to see that we can test is a given set belongs to each $\mathcal{I}_{i,j}(D^*)$ in $O(|A^*|)$ time, we can test if a given set belongs to $\mathcal{I}(D^*)$ in $O(|A^*|^4)$ time from Theorem A.1. For $\mathcal{J}(D^*)$, the time complexity is clearly $O(|A^*|)$ time. The size of every complete D^*-intersection is equal to M from (4). Thus, the total time required for solving WMI(D^*) is $O(M|A^*|^6)$ from Theorem A.2. □

B Proof of Proposition 3.1

Let B be a D^*-rooted connector. From the definition of a D^*-rooted connector, $|\delta_{D^*+B}(v)| \geq f(R_D(v))$ holds for every $v \in V$. Thus, the number of arcs of $D^* + B$ is at least $\sum_{v \in V} f(R_D(v))$. Since the number of arcs of D^* is equal to $|A| + f(S)$ from (3), the proposition holds. □

C Proof of Lemma 3.3

The correctness of the algorithm follows from Lemma 3.2. Thus, we consider the time complexity. In Step 1, we have to compute $R_D(v)$ for every $v \in V$. This can be done in $O(|V||A|)$ time by applying depth-first search from every $s_i \in S$. After this, the time required to test whether $|\delta_{D^*}(v)| \leq f(R_D(v))$ for all $v \in V$ is $O(|A|)$. Thus, the time required for Step 1 is $O(|V||A|)$. Since the number of arcs of $D + B$ is at most M for a D^*-rooted connector B with $|B| = \mathsf{opt}_D$ from (6), the time required for Step 8 is $O(M^4)$ from Theorem 2.3. Moreover, since the number of arcs of $D + B$ is at most M, the time required for Step 9 is $O(M)$ from the definition of Procedure $\mathsf{Replace}$. Hence, since the time required for Step 4 is γ_1, the lemma follows. □

D Proof of Lemma 3.5

(i): We first prove that I is a base of $\boldsymbol{M}((D + B)^*)$. Since I is a base of $\boldsymbol{M}(D^*)$, I can be partitioned into $\{I_{i,1}, \ldots, I_{i,f(s_i)} \colon i = 1, \ldots, d\}$ such that a directed graph $(V_D^i \cup \{s^*\}, I_{i,j})$ is a tree for every $i = 1, \ldots, d$ and $j = 1, \ldots, f(s_i)$. Thus, since each $(V_{D+B}^i \cup \{s^*\}, I_{i,j})$ is a tree from (2), I is a base of $\boldsymbol{M}((D + B)^*)$.

Next we prove that I is a base of $\boldsymbol{U}((D + B)^*)$. Since I is a base of $\boldsymbol{U}(D^*)$, $|\delta_{D^*}(v) \cap I|$ is equal to (i) $f(R_D(v))$ if $v \in V$, or (ii) 0 if $v = s^*$. Furthermore, since $I \cap B = \emptyset$ follows from $I \subseteq A^*$, $|\delta_{D^*}(v) \cap I|$ is equal to $|\delta_{(D+B)^*}(v) \cap I|$ for every $v \in V$. Thus, $|\delta_{(D+B)^*}(v) \cap I|$ is equal to (i) $f(R_D(v)) = f(R_{D+B}(v))$ from (1) if $v \in V$, or (ii) 0 if $v = s^*$. This proves that I is a base of $\boldsymbol{U}((D + B)^*)$.

(ii): This part can be proved in the same manner as in the proof of (i). □

E Proof of Lemma 3.6

From (11), we have $w(I) = |I| - |I \cap A^*|$. Furthermore, from (3) and (10), $|I| - |I \cap A^*| \geq |I| - |A^*| = \sum_{v \in V} f(R_D(v)) - (|A| + f(S))$. Thus, $w(I) \geq \mathsf{opt}_D$ follows from (6). From the above equation, $w(I) = \mathsf{opt}_D$ if and only if $|I \cap A^*| = |A^*|$. This proves the rest of the lemma. ◽

F Proof of Theorem 4.1

If-part: Since D has no cycle, we can label vertices in V as follows, based on topological ordering: (i) A label of each vertex is an integer between 1 and $|V|$. (ii) For any $e \in A$, a label of $\partial^+(e)$ is smaller than that of $\partial^-(e)$. For $W \subseteq V$, we denote by $D[W]$ a subgraph of $D = (V, A, S, f)$ induced by W with a set of specified vertices $S \cap W$ and a restriction of f on $S \cap W$. Let V_t be the set of all vertices whose label is at most t. We prove by induction on t. For $t = 1$, it is clear that there exists a $D[V_1]$-canonical set of in-trees covering the arc set of $D[V_1]$. Assume that in the case of $t \geq 1$, there exists a $D[V_t]$-canonical set $\mathcal{T}$ of in-trees covering the arc set of $D[V_t]$. For $s_i \in S \cap V_t$ and $j = 1, \ldots, f(s_i)$, let $T_{i,j}$ be an in-tree of $\mathcal{T}$ which is rooted at s_i and spans vertices in V_t from which s_i is reachable. Let v be a vertex whose label is equal to $t + 1$. We assume that $v \notin S$. The case of $v \in S$ can be proved in the same manner. In this case, from $S \cap V_t = S \cap V_{t+1}$, we will construct a set $\mathcal{T}'$ of in-trees which consists of $T'_{i,1}, \ldots, T'_{i,f(s_i)}$ for $s_i \in S \cap V_t$ $(= S \cap V_{t+1})$ such that each $T'_{i,j}$ is obtained from $T_{i,j}$. We first consider $T'_{i,j}$ for $s_i \in (S \cap V_t) \setminus R_D(v)$. For $s_i \in (S \cap V_t) \setminus R_D(v)$, from $V^i_{D[V_t]} = V^i_{D[V_{t+1}]}$ holds, $T_{i,j}$ is also a $(D[V_{t+1}], s_i)$-in-tree. Thus, we set $T'_{i,j} = T_{i,j}$. Next we consider $T'_{i,j}$ for $s_i \in R_D(v)$. For $s_i \in R_D(v)$, since $V^i_{D[V_{t+1}]} = V^i_{D[V_t]} \cup \{v\}$ holds, we need to add an arc in $\delta_D(v)$ to $T_{i,j}$. Here we use a matching $\mathcal{M}$ in G_v which saturates vertices in X_v. For each edge $x_e y_{i,j} \in \mathcal{M}$, we set $T'_{i,j}$ be an in-tree obtained by adding an arc e to $T_{i,j}$. If there exists $y_{i',j'} \in Y_v$ which is not contained in any edge in $\mathcal{M}$, we arbitrarily choose an arc $e' \in \delta_D(v)$ such that $x_{e'}$ is a neighbour of $y_{i',j'}$ in G_v and we set $T'_{i',j'}$ to be an in-tree obtained by adding e' to $T'_{i',j'}$. From the way of construction, $\mathcal{T}'$ is clearly a $D[V_{t+1}]$-canonical set of in-trees. Since M saturates vertices in X_v, $T'_{i,1}, \ldots, T'_{i,f(s_i)}$ with $s_i \in R_D(v)$ contain all arcs in $\delta_D(v)$. Thus, since $\mathcal{T}$ covers the arc set of $D[V_t]$ from the induction hypothesis, $\mathcal{T}'$ covers the arc set of $D[V_{t+1}]$.

Only if-part: Assume that there exists a D-canonical set $\mathcal{T}$ of in-trees covering A. For $i = 1, \ldots, d$, we denote $f(s_i)$ (D, s_i)-in-trees of $\mathcal{T}$ by $T_{i,1}, \ldots, T_{i,f(s_i)}$. Let us fix $v \in V$, and for X_v and Y_v we define a set E' in which an edge $x_e y_{i,j}$ is contained in E' if and only if $e \in \delta_D(v)$ is contained in $T_{i,j}$. If $e \in \delta_D(v)$ is contained in $T_{i,j}$, s_i is reachable from $\partial^+(e)$. Thus, E' is a subset of E_v. Since $\mathcal{T}$ covers A, each $e \in \delta_D(v)$ is contained in at least one in-tree in $\mathcal{T}$. That is, E' saturates X_v. Since $T_{i,j}$ is an in-tree, each $y_{i,j}$ is contained in exactly one edge in E'. Thus, it is not difficult to see that a matching in G_v which saturates vertices in X_v can be obtained from E'. This completes the proof. ◽