Journal of Interconnection Networks
Vol. 9, Nos. 1 & 2 (2008) 141–155

COST-CONSTRAINED MINIMUM-DELAY MULTICASTING

SATOSHI TAYU, TURKI GHAZI AL-MUTAIRI and SHUICHI UENO

Department of Communications and Integrated Systems, Tokyo Institute of Technology, Tokyo 152-8550-S3-57, Japan
{tayu, ueno}@lab.ss.titech.ac.jp

Received 13 November 2007
Revised 18 June 2008

We consider a problem of cost-constrained minimum-delay multicasting in a network, which is to find a Steiner tree spanning the source and destination nodes such that the maximum total delay along a path from the source node to a destination node is minimized, while the sum of link costs in the tree is bounded by a constant. The problem is NP-hard even if the network is series-parallel. We present a fully polynomial time approximation scheme for the problem if the network is series-parallel.

Keywords: Bicriteria Steiner tree problem; Fully polynomial time approximation scheme (FPTAS); Multicasting; NP-hardness; Series-parallel graph.

1. Introduction

The multicasting is the simultaneous transmission of data from a source node to multiple destination nodes in a network. The multicasting involves the generation of a multicast tree, which is a Steiner tree spanning the source and destination nodes. The performance of multicasting is determined by both the cost of the multicast tree and the maximum delay between the source node and a destination node in the tree. Therefore, constructing efficient multicasting is formulated as a bicriteria Steiner tree problem.

In connection with the problem, the following problem has been considered in the literature [1,4,6,7,10]. The delay-constrained minimum cost multicast tree problem (DCMCMT) is to construct a multicast tree such that the cost of the tree is minimized while the delay between the source node and a destination node in the tree is bounded by a constant integer. DCMCMT is NP-hard since it reduces to the Steiner tree problem, which is well-known to be NP-hard. Chen and Xue proposed a fully polynomial time approximation scheme (FPTAS) for DCMCMT if the number of destination nodes is bounded by a constant [1], while many heuristic algorithms have been proposed in [4,6,7,10]. We present a pseudo-polynomial time algorithm for DCMCMT if the network is series-parallel.

We also consider the following problem, which is another variant of the problem of constructing efficient multicasting. The cost-constrained minimum delay multicast tree problem (CCMDMT) is to construct a multicast tree such that the maximum delay between the source node and a destination node in the tree is minimized while the cost of the tree is bounded by a constant integer. CCMDMT is NP-hard since it reduces to the cost-constrained shortest path problem (CCSP) which is known to be NP-hard [3]. In fact, CCMDMT is NP-hard even for series-parallel networks, since CCSP is NP-hard for series-parallel networks as mentioned by Chen and Xue [2]. We present in this paper a pseudo-polynomial time algorithm and an FPTAS for CCMDMT if the network is series-parallel. This paper is the first to consider CCMDMT, as far as the authors know.

2. Problems

We consider a connected graph G with vertex set $V(G)$ and edge set $E(G)$. Each edge e is assigned a cost $\gamma(e)$ and a delay $\delta(e)$ which are assumed to be non-negative integers. The cost of a subgraph H of G, denoted by $\gamma(H)$, is defined as $\gamma(H) = \sum_{e\in E(H)} \gamma(e)$. The delay of a path P in G, denoted by $\delta(P)$, is defined as $\delta(P) = \sum_{e\in E(P)} \delta(e)$. A vertex s is designated as the source and a set D of vertices is designated as the destinations. A tree T is called a multicast tree if $\{s\} \cup D \subseteq V(T)$. The delay of a multicast tree T, denoted by $\delta(T)$, is defined as $\delta(T) = \max\{\delta(P(s,d)) | d \in D, P(s,d)\text{: } (s,d)\text{-path in } T\}$. Let Γ and Δ be positive integers. The delay-constrained minimum cost multicast tree problem (DCMCMT) is to construct a multicast tree T such that $\delta(T) \leq \Delta$ and $\gamma(T)$ is minimized, while the cost-constrained minimum delay multicast tree problem (CCMDMT) is to construct a multicast tree T such that $\gamma(T) \leq \Gamma$ and $\delta(T)$ is minimized.

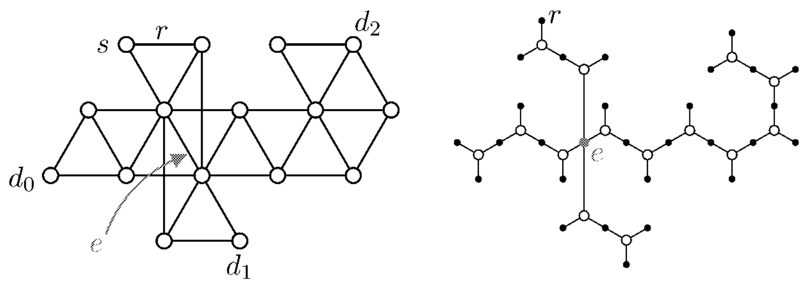
3. Pseudo-Polynomial Time Algorithms

A graph is said to be *series-parallel* if it contains no subdivision of K_4 as a subgraph. A maximal series-parallel graph is called a 2-*tree*. The 2-trees can be defined recursively as follows: (1) K_2 is a 2-tree on two vertices; (2) Given a 2-tree on n vertices ($n \geq 2$), a graph obtained from G by adding a new vertex adjacent to the ends of an edge of G is a 2-tree on $n+1$ vertices. A 2-tree on $n \geq 2$ vertices has $2n-3$ edges by definition.

In this section, we will show an $\mathcal{O}(n\Delta^3)$ time algorithm and an $\mathcal{O}(n^4{\delta_{\max}}^3)$ time algorithm to solve DCMCMT and CCMDMT, respectively, for a series-parallel graph G with n vertices, where $\delta_{\max} = \max\{\delta(e)|e \in E(G)\}$. We use methods similar to those used in [2]. We first augment a connected series-parallel graph with n vertices to a 2-tree on n vertices using a linear time algorithm presented in [8]. Each added edge has infinite cost and delay so that the added edges are never chosen in an optimal multicast tree. We next find an optimal multicast tree in the 2-tree. The algorithms are based on the dynamic programming.

3.1. *Preliminaries*

Let G be a 2-tree and $C_3(G)$ be the set of triangles of G. A tree T_G is defined as follows: $V(T_G) = E(G) \cup C_3(G)$; for any $e \in E(G)$ and $\nabla \in C_3(G)$, $(e, \nabla) \in E(T_G)$ if and only if $e \in E(\nabla)$. It is easy to see that T_G thus defined is indeed a tree since G is a 2-tree. T_G is considered as a rooted tree with root r, where r is an edge incident to s in G. Figure 1 shows a 2-tree G with root r and destinations d_0, d_1, and d_2, and *the* corresponding rooted tree T_G.



(a) 2-tree G. (b) T_G.

Fig. 1. 2-tree G and T_G.

Let $\mathcal{C}(p)$ be the set of all children of $p \in E(G)$ in T_G. Notice that a child of p is a triangle in G. Let $\mathcal{D}(\nabla)$ be the set of triangles which are descendants of $\nabla \in C_3(G)$ in T_G. For $\mathcal{C}'(p) \subseteq \mathcal{C}(p)$, $G[p, \mathcal{C}'(p)]$ is a subgraph of G induced by the edges of triangles in $\bigcup_{\nabla \in \mathcal{C}'(p)} \mathcal{D}(\nabla)$ together with edge p.

Let $\prec$ be a partial order on $V(G)$ satisfying the following conditions:

- $s \prec v$ for all $v \in V(G)$;
- If ∇ is a triangle with $V(\nabla) = \{x, y, z\}$, and edge (x, z) is the parent of ∇ with $x \prec z$, then $x \prec y$ and $y \prec z$.

Such an order can be constructed recursively from the root of T_G as follows: First, we define $s \prec v$ for edge $r = (s, v)$. For every edge $p = (x, z)$ with $x \prec z$, if p has a child triangle C, we define $x \prec y$ and $y \prec z$ for vertex $y \in V(C) \setminus \{x, z\}$. We continue this process until $\prec$ is defined on every pair of endvertices of an edge. Then the transitive reflexive closure of $\prec$ is the desired partial order.

For any edge $p = (x, y)$ with $x \prec y$ and $\mathcal{C}'(p) \subseteq \mathcal{C}(p)$, $H_{\bullet\bullet}^{[p,\mathcal{C}'(p)]}$, $H_{\bullet\circ}^{[p,\mathcal{C}'(p)]}$, $H_{\circ\bullet}^{[p,\mathcal{C}'(p)]}$, and $H_{\bullet\bullet}^{[p,\mathcal{C}'(p)]}$ are subgraphs of $G[p, \mathcal{C}'(p)]$ such that each subgraph contains the vertices (destinations) in $D \cap V(G[p, \mathcal{C}'(p)])$ and;

- $H_{\bullet\bullet}^{[p,\mathcal{C}'(p)]}$ is a tree including both x and y,
- $H_{\bullet\circ}^{[p,\mathcal{C}'(p)]}$ is a tree with $x \in V(H_{\bullet\circ}^{[p,\mathcal{C}'(p)]})$ and $y \notin V(H_{\bullet\circ}^{[p,\mathcal{C}'(p)]})$,
- $H_{\circ\bullet}^{[p,\mathcal{C}'(p)]}$ is a tree with $x \notin V(H_{\circ\bullet}^{[p,\mathcal{C}'(p)]})$ and $y \in V(H_{\circ\bullet}^{[p,\mathcal{C}'(p)]})$,
- $H_{\bullet\bullet}^{[p,\mathcal{C}'(p)]}$ consists of vertex-disjoint two trees $T_x^{[p,\mathcal{C}'(p)]}$ and $T_y^{[p,\mathcal{C}'(p)]}$ such that

$x \in V(T_x^{[p,\mathcal{C}'(p)]})$ and $y \in V(T_y^{[p,\mathcal{C}'(p)]})$.

Finally, let $\mathbb{S}_\Delta = \{-\infty, 0, 1, \ldots, \Delta\}$.

3.2. *Functions*

Let $p = (x, y)$ be an edge with $x \prec y$ and $\mathcal{C}'(p) \subseteq \mathcal{C}(p)$.

$W_{\bullet\!\!\bullet}(p, \mathcal{C}'(p); \xi_x, \xi_{xy})$ is the minimum cost of a tree $H_{\bullet\!\!\bullet}^{[p,\mathcal{C}'(p)]}$ in $G[p, \mathcal{C}'(p)]$ such that $\max\{\delta(x, d) | d \in D \cap V(G[p, \mathcal{C}'(p)])\} \le \xi_x$ and $\delta(x, y) \le \xi_{xy}$, where $\delta(u, v)$ is the delay of (u, v)-path in tree $H_{\bullet\!\!\bullet}^{[p,\mathcal{C}'(p)]}$, and $\xi_x, \xi_{xy} \in \mathbb{S}_\Delta$.

$\overline{W}_{\bullet\!\!\bullet}(p, \mathcal{C}'(p); \xi_y, \xi_{xy})$ is the minimum cost of a tree $H_{\bullet\!\!\bullet}^{[p,\mathcal{C}'(p)]}$ in $G[p, \mathcal{C}'(p)]$ such that $\max\{\delta(y, d) | d \in D \cap V(G[p, \mathcal{C}'(p)])\} \le \xi_y$ and $\delta(x, y) \le \xi_{xy}$, where $\delta(u, v)$ is the delay of (u, v)-path in tree $H_{\bullet\!\!\bullet}^{[p,\mathcal{C}'(p)]}$, and $\xi_y, \xi_{xy} \in \mathbb{S}_\Delta$.

$W_{\bullet\circ}(p, \mathcal{C}'(p); \xi_x)$ is the minimum cost of a tree $H_{\bullet\circ}^{[p,\mathcal{C}'(p)]}$ in $G[p, \mathcal{C}'(p)]$ such that $\max\{\delta(x, d) | d \in D \cap V(G[p, \mathcal{C}'(p)])\} \le \xi_x$, where $\delta(x, d)$ is the delay of (x, d)-path in tree $H_{\bullet\circ}^{[p,\mathcal{C}'(p)]}$, and $\xi_x \in \mathbb{S}_\Delta$.

$W_{\circ\bullet}(p, \mathcal{C}'(p); \xi_y)$ is the minimum cost of a tree $H_{\circ\bullet}^{[p,\mathcal{C}'(p)]}$ in $G[p, \mathcal{C}'(p)]$ such that $\max\{\delta(y, d) | d \in D \cap V(G[p, \mathcal{C}'(p)])\} \le \xi_y$, where $\delta(x, d)$ is the delay of (x, d)-path in tree $H_{\circ\bullet}^{[p,\mathcal{C}'(p)]}$, and $\xi_y \in \mathbb{S}_\Delta$.

$W_{\bullet\bullet}(p, \mathcal{C}'(p); \xi_x, \xi_y)$ is the minimum cost of a forest $H_{\bullet\bullet}^{[p,\mathcal{C}'(p)]}$ in $G[p, \mathcal{C}'(p)]$ such that $\max\{\delta(x, d) | d \in D \cap V(T_x^{[p,\mathcal{C}'(p)]})\} \le \xi_x$, and $\max\{\delta(y, d) | d \in D \cap V(T_y^{[p,\mathcal{C}'(p)]})\} \le \xi_y$, where $\delta(x, d)$ is the delay of (x, d)-path in tree $T_x^{[p,\mathcal{C}'(p)]}$ and $\delta(y, d)$ is the delay of (y, d)-path in tree $T_y^{[p,\mathcal{C}'(p)]}$, and $\xi_x, \xi_y \in \mathbb{S}_\Delta$.

$N_{\circ\circ}(p)$ is defined to be 0 if $G[p, \mathcal{C}(p)]$ has no destination and ∞ otherwise.

For an edge $p = (x, y) \in E(G)$ with $x \prec y$ and $\mathcal{C}'(p) \subseteq \mathcal{C}(p)$, the *table* $\mathcal{W}(p, \mathcal{C}'(p))$ for p and $\mathcal{C}'(p)$ is the list of values of $W_{\bullet\!\!\bullet}(p, \mathcal{C}'(p); \xi_x, \xi_{xy})$, $\overline{W}_{\bullet\!\!\bullet}(p, \mathcal{C}'(p); \xi_y, \xi_{xy})$, $W_{\bullet\circ}(p, \mathcal{C}'(p); \xi_x)$, $W_{\circ\bullet}(p, \mathcal{C}'(p); \xi_y)$, and $W_{\bullet\bullet}(p, \mathcal{C}'(p); \xi_x, \xi_y)$ for every $\xi_x, \xi_y, \xi_{xy} \in \mathbb{S}_\Delta$.

The following is immediate from the definition of functions above.

Lemma 3.1. *For any* $\xi \in \{0, 1, \ldots, \Delta\}$, $\min\{W_{\bullet\circ}(r, \mathcal{C}(r); \xi), W_{\bullet\!\!\bullet}(r, \mathcal{C}(r); \xi, \Delta)\}$ *is the minimum cost of a multicast tree* T *of* G *with* $\delta(T) \le \xi$, *where* $r = (s, y) \in E(G)$ *is the root of* T_G. ∎

Lemma 3.2. *Any function in* $\mathcal{W}(p, \mathcal{C}'(p))$ *is non-increasing for* ξ_x, ξ_y, *and* ξ_{xy}. ∎

3.3. *Basic Algorithm* BA$(G, s, D, \gamma, \delta, \Delta)$

We describe in this subsection a basic algorithm BA$(G, s, D, \gamma, \delta, \Delta)$ which computes $\mathcal{W}(r, \mathcal{C}(r))$ for a 2-tree G with n vertices in $\mathcal{O}(n\Delta^3)$ time.

BA$(G, s, D, \gamma, \delta, \Delta)$ first computes T_G and chooses an edge incident with s in G as the root of T_G.

Then, BA$(G, s, D, \gamma, \delta, \Delta)$ recursively computes tables $\mathcal{W}(p, \mathcal{C}'(p))$ for all $p \in$

$E(G)$. When we compute functions in each table, we distinguish three cases.

Case 1 : $\mathcal{C}'(p) = \emptyset$.
For every $p = (x, y) \in E(G)$ with $x \prec y$, and $\xi_x, \xi_y, \xi_{xy} \in \mathbb{S}_\Delta$,

$$W_{\bullet\!-\!\bullet}(p, \emptyset; \xi_x, \xi_{xy}) = \begin{cases} \gamma(p) \text{ if the following conditions are satisfied:} \\ \quad \text{(i) if } y \in D \text{ then } \xi_x \geq \delta(p); \\ \quad \text{(ii) if } x \in D \text{ then } \xi_x \geq 0; \\ \quad \text{(iii) } \xi_{xy} \geq \delta(p), \\ \infty \text{ otherwise.} \end{cases}$$

$$\overline{W}_{\bullet\!-\!\bullet}(p, \emptyset; \xi_y, \xi_{xy}) = \begin{cases} \gamma(p) \text{ if the following conditions are satisfied:} \\ \quad \text{(i) if } y \in D \text{ then } \xi_y \geq 0; \\ \quad \text{(ii) if } x \in D \text{ then } \xi_y \geq \delta(p); \\ \quad \text{(iii) } \xi_{xy} \geq \delta(p), \\ \infty \text{ otherwise.} \end{cases}$$

$$W_{\bullet\circ}(p, \emptyset; \xi_x) = \begin{cases} 0 \text{ if } y \notin D, \text{ and if } x \in D \text{ then } \xi_x \geq 0, \\ \infty \text{ otherwise.} \end{cases}$$

$$W_{\circ\bullet}(p, \emptyset; \xi_y) = \begin{cases} 0 \text{ if } x \notin D, \text{ and if } y \in D \text{ then } \xi_y \geq 0, \\ \infty \text{ otherwise.} \end{cases}$$

$$W_{\bullet\bullet}(p, \emptyset; \xi_x, \xi_y) = \begin{cases} 0 \text{ if the following conditions are satisfied:} \\ \quad \text{(i) if } x \in D \text{ then } \xi_x \geq 0, \\ \quad \text{(ii) if } y \in D \text{ then } \xi_y \geq 0, \\ \infty \text{ otherwise.} \end{cases}$$

Case 2 : $\mathcal{C}'(p) = \{\nabla\}$ for some $\nabla \in \mathcal{C}(p)$.
For every $p = (x, z) \in E(G)$ with $\mathcal{C}(p) \neq \emptyset$ and $x \prec z$, for every $\nabla \in \mathcal{C}(p)$ with $V(\nabla) = \{x, y, z\}$, $E(\nabla) = \{p = (x, z), q = (x, y), t = (y, z)\}$, and $x \prec y \prec z$, and for every $\xi_x, \xi_y, \xi_{xy} \in \mathbb{S}_\Delta$, the functions are computed as follows.

$$\begin{aligned} & W_{\bullet\!-\!\bullet}(p, \{\nabla\}; \xi_x, \xi_{xz}) \;=\; \\ & \min \left\{ \begin{array}{l} \min \big\{ \gamma(p) + W_{\bullet\circ}(q, \mathcal{C}(q); \xi'_x) + W_{\circ\bullet}(t, \mathcal{C}(t); \xi''_z) \;\big|\; \xi_{xz} \geq \delta(p), \\ \quad \xi_x \geq \max\{\xi'_x, \delta(p) + \xi''_z\}, (\xi'_x, \xi''_z) \in \mathbb{S}^2_\Delta \big\}, \\ \min \big\{ \gamma(p) + W_{\bullet\!-\!\bullet}(q, \mathcal{C}(q); \xi'_x, \xi'_{xy}) + W_{\bullet\bullet}(t, \mathcal{C}(t); \xi''_y, \xi''_z) \;\big| \\ \quad \xi_{xz} \geq \delta(p), \xi_x \geq \max\{\xi'_x, \xi'_{xy} + \xi''_y, \delta(p) + \xi''_z\}, \\ \quad (\xi'_x, \xi'_{xy}, \xi''_y, \xi''_z) \in \mathbb{S}^4_\Delta \big\}, \\ \min \big\{ \gamma(p) + W_{\bullet\bullet}(q, \mathcal{C}(q); \xi'_x, \xi'_y) + \overline{W}_{\bullet\!-\!\bullet}(t, \mathcal{C}(t); \xi''_z, \xi''_{yz}) \;\big| \\ \quad \xi_{xz} \geq \delta(p), \xi_x \geq \max\{\xi'_x, \delta(p) + \xi''_z, \delta(p) + \xi''_{yz} + \xi'_y\}, \\ \quad (\xi'_x, \xi'_y, \xi''_z, \xi''_{yz}) \in \mathbb{S}^4_\Delta \big\}, \\ \min \big\{ W_{\bullet\!-\!\bullet}(q, \mathcal{C}(q); \xi'_x, \xi'_{xy}) + W_{\bullet\!-\!\bullet}(t, \mathcal{C}(t); \xi''_y, \xi''_{yz}) \;\big| \\ \quad \xi_{xz} \geq \xi'_{xy} + \xi''_{yz}, \xi_x \geq \max\{\xi'_x, \xi'_{xy} + \xi''_y\}, \\ \quad (\xi'_x, \xi'_{xy}, \xi''_y, \xi''_{yz}) \in \mathbb{S}^4_\Delta \big\} \end{array} \right\}, \end{aligned} \tag{3.1}$$

$$\overline{W}_{\bullet\bullet}(p,\{\nabla\};\xi_z,\xi_{xz}) =$$
$$\min\left\{\begin{array}{l}
\min\{\gamma(p)+W_{\bullet\circ}(q,\mathcal{C}(q);\xi'_x)+W_{\circ\bullet}(t,\mathcal{C}(t);\xi''_z) \mid \xi_{xz}\geq\delta(p),\\
\quad \xi_x\geq\max\{\xi'_x,\delta(p)+\xi''_z\},(\xi'_x,\xi''_z)\in\mathbb{S}^2_\Delta\},\\
\min\{\gamma(p)+\overline{W}_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_y,\xi'_{xy})+W_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_y,\xi''_z) \mid\\
\quad \xi_{xz}\geq\delta(p),\xi_x\geq\max\{\xi'_x,\xi'_{xy}+\xi''_y,\delta(p)+\xi''_z\},\\
\quad (\xi'_y,\xi'_{xy},\xi''_y,\xi''_z)\in\mathbb{S}^4_\Delta\},\\
\min\{\gamma(p)+W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_y)+\overline{W}_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_z,\xi''_{yz}) \mid\\
\quad \xi_{xz}\geq\delta(p),\xi_x\geq\max\{\xi'_x,\delta(p)+\xi''_z,\delta(p)+\xi''_{yz}+\xi'_y\},\\
\quad (\xi'_y,\xi''_y,\xi''_z,\xi''_{yz})\in\mathbb{S}^4_\Delta\},\\
\min\{\overline{W}_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_y,\xi'_{xy})+\overline{W}_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_z,\xi''_{yz}) \mid\\
\quad \xi_{xz}\geq\xi'_{xy}+\xi''_{yz},\xi_z\geq\max\{\xi'_y+\xi''_{yz},\xi''_z\},\\
\quad (\xi'_y,\xi'_{xy},\xi''_z,\xi''_{yz})\in\mathbb{S}^4_\Delta\}
\end{array}\right\}, \tag{3.2}$$

$$W_{\bullet\circ}(p,\{\nabla\};\xi_x) =$$
$$\min\left\{\begin{array}{l}
\min\{W_{\bullet\circ}(q,\mathcal{C}(q);\xi'_x)+N_{\circ\circ}(t) \mid \xi_x\geq\xi'_x,\xi'_x\in\mathbb{S}_\Delta\},\\
\min\{W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_{xy})+W_{\bullet\circ}(t,\mathcal{C}(t);\xi''_y) \mid\\
\quad \xi_x\geq\max\{\xi'_x,\xi'_{xy}+\xi''_y\},(\xi'_x,\xi'_{xy},\xi''_y)\in\mathbb{S}^3_\Delta\}
\end{array}\right\}, \tag{3.3}$$

$$W_{\circ\bullet}(p,\{\nabla\};\xi_z) =$$
$$\min\left\{\begin{array}{l}
\min\{N_{\circ\circ}(q)+W_{\circ\bullet}(t,\mathcal{C}(t);\xi''_z) \mid \xi_z\geq\xi''_z,\xi''_z\in\mathbb{S}_\Delta\},\\
\min\{W_{\circ\bullet}(q,\mathcal{C}(q);\xi'_y)+\overline{W}_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_z,\xi''_{yz}) \mid\\
\quad \xi_z\geq\max\{\xi''_z,\xi''_{yz}+\xi'_y\},(\xi'_y,\xi''_z,\xi''_{yz})\in\mathbb{S}^3_\Delta\}
\end{array}\right\}, \tag{3.4}$$

$$W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_z) =$$
$$\min\left\{\begin{array}{l}
\min\{W_{\bullet\circ}(q,\mathcal{C}(q);\xi'_x)+W_{\circ\bullet}(t,\mathcal{C}(t);\xi''_z) \mid \xi_x\geq\xi'_x,\xi_z\geq\xi''_z,\\
\quad (\xi'_x,\xi''_z)\in\mathbb{S}^2_\Delta\},\\
\min\{W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_{xy})+W_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_y,\xi''_z) \mid\\
\quad \xi_x\geq\max\{\xi'_x,\xi'_{xy}+\xi''_y\},\xi_z\geq\xi''_z,(\xi'_x,\xi'_{xy},\xi''_y,\xi''_z)\in\mathbb{S}^4_\Delta\},\\
\min\{W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_y)+\overline{W}_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_z,\xi''_{yz}) \mid \xi_x\geq\xi'_x,\\
\quad \xi_z\geq\max\{\xi''_z,\xi''_{yz}+\xi'_y\},(\xi'_x,\xi'_{xy},\xi''_z,\xi''_{yz})\in\mathbb{S}^4_\Delta\}
\end{array}\right\}. \tag{3.5}$$

Case 3 : $\mathcal{C}'(p)=\mathcal{C}''(p)\cup\{\nabla\}$ for some $\mathcal{C}''(p)\subseteq\mathcal{C}(p)$ and $\nabla\in\mathcal{C}(p)-\mathcal{C}''(p)$.
For every $p=(x,y)\in E(G)$, $\mathcal{C}'(p)\subseteq\mathcal{C}(p)$, $\nabla\in\mathcal{C}'(p)$, and $\xi_x,\xi_y,\xi_{xy}\in\mathbb{S}_\Delta$, the functions are computed as follows:

$$W_{\bullet\bullet}(p,\mathcal{C}'(p);\xi_x,\xi_{xy}) =$$
$$\min\left\{\begin{array}{l}
\min\{W_{\bullet\bullet}(p,\mathcal{C}''(p);\xi'_x,\xi'_{xy})+W_{\bullet\bullet}(p,\{\nabla\};\xi''_x,\xi''_y) \mid \xi_{xy}\geq\xi'_{xy},\\
\quad \xi_x\geq\max\{\xi'_x,\xi''_x,\xi''_y+\xi'_{xy}\},(\xi'_x,\xi'_{xy},\xi''_x,\xi''_y)\in\mathbb{S}^4_\Delta\},\\
\min\{W_{\bullet\bullet}(p,\mathcal{C}''(p);\xi'_x,\xi'_y)+W_{\bullet\bullet}(p,\{\nabla\};\xi''_x,\xi''_{xy}) \mid \xi_{xy}\geq\xi''_{xy},\\
\quad \xi_x\geq\max\{\xi'_x,\xi''_x,\xi'_y+\xi''_{xy}\},(\xi'_x,\xi'_y,\xi''_x,\xi''_{xy})\in\mathbb{S}^4_\Delta\}
\end{array}\right\},$$

$$\overline{W}_{\bullet\!-\!\bullet}(p,\mathcal{C}'(p);\xi_y,\xi_{xy}) \;=\; \min\left\{\begin{array}{l} \min\big\{\overline{W}_{\bullet\!-\!\bullet}(p,\mathcal{C}''(p);\xi'_y,\xi'_{xy}) + W_{\bullet\bullet}(p,\{\nabla\};\xi''_x,\xi''_y) \;\big|\; \xi_{xy}\ge\xi'_{xy}, \\ \quad \xi_y\ge \max\{\xi'_y,\xi''_y,\xi''_x+\xi'_{xy}\}, (\xi'_y,\xi'_{xy},\xi''_x,\xi''_y)\in\mathbb{S}^4_\Delta\big\}, \\ \min\big\{W_{\bullet\bullet}(p,\mathcal{C}''(p);\xi'_x,\xi'_y) + \overline{W}_{\bullet\!-\!\bullet}(p,\{\nabla\};\xi''_y,\xi''_{xy}) \;\big|\; \xi_{xy}\ge\xi''_{xy}, \\ \quad \xi_y\ge \max\{\xi'_y,\xi''_y,\xi'_x+\xi''_{xy}\}, (\xi'_y,\xi'_y,\xi''_y,\xi''_{xy})\in\mathbb{S}^4_\Delta\big\} \end{array}\right\},$$

$$W_{\bullet\circ}(p,\mathcal{C}'(p);\xi_x) = W_{\bullet\circ}(p,\mathcal{C}''(p);\xi_x) + W_{\bullet\circ}(p,\{\nabla\};\xi_x),$$
$$W_{\circ\bullet}(p,\mathcal{C}'(p);\xi_y) = W_{\circ\bullet}(p,\mathcal{C}''(p);\xi_y) + W_{\circ\bullet}(p,\{\nabla\};\xi_y),$$
$$W_{\bullet\bullet}(p,\mathcal{C}'(p);\xi_x,\xi_y) = W_{\bullet\bullet}(p,\mathcal{C}''(p);\xi_x,\xi_y) + W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_y).$$

The computation of the tables for functions proceeds as follows. We first compute $\mathcal{W}(p,\mathcal{C}(p)) = \mathcal{W}(p,\emptyset)$ for every leaf p of T_G as in Case 1 above.

For every triangle ∇ with parent p and children q and t, $\mathcal{W}(p,\{\nabla\})$ is computed using tables $\mathcal{W}(q,\mathcal{C}(q))$ and $\mathcal{W}(t,\mathcal{C}(t))$ as in Case 2.

For every $p\in E(G)$ with $\mathcal{C}(p) = \{\nabla_1,\nabla_2,\ldots,\nabla_{|\mathcal{C}(p)|}\}$, $\mathcal{W}(p,\mathcal{C}(p))$ is computed as follows. Let $\mathcal{C}^{(i)}(p) = \{\nabla_1,\nabla_2,\ldots,\nabla_i\}$ for $1\le i\le|\mathcal{C}(p)|$. $\mathcal{W}(p,\mathcal{C}^{(i)}(p))$ is computed using $\mathcal{W}(p,\mathcal{C}^{(i-1)}(p))$ and $\mathcal{W}(p,\{\nabla_i\})$ as in Case 3 for $2\le i\le|\mathcal{C}(p)|$.

Finally, $\mathrm{BA}(G,s,D,\gamma,\delta,\Delta)$ outputs $\mathcal{W}(r,\mathcal{C}(r))$.

3.4. *Analysis of* $\mathbf{BA}(G,s,D,\gamma,\delta,\Delta)$

We use the following lemmas to prove Theorem 3.1 below. Lemmas 3.3 and 3.4 are rather obvious.

Lemma 3.3. $\mathrm{BA}(G,s,D,\gamma,\delta,\Delta)$ *computes* $\mathcal{W}(r,\mathcal{C}(r))$, *correctly.* ■

Lemma 3.4. $\mathcal{W}(p,\emptyset)$ *is computed in* $\mathcal{O}(\Delta^2)$ *time for any leaf* p *of* T_G. ■

Lemma 3.5. *Let* ∇ *be a triangle with parent* p *and children* q *and* t. *Given* $\mathcal{W}(q,\mathcal{C}(q))$ *and* $\mathcal{W}(t,\mathcal{C}(t))$, $\mathcal{W}(p,\{\nabla\})$ *is computed in* $\mathcal{O}(\Delta^3)$ *time.*

Proof. We only show the computational time for (3.1) in $\mathcal{W}(p,\{\nabla\})$, that is, we show how to compute $W_{\bullet\!-\!\bullet}(p,\{\nabla\};\xi_x,\xi_{xz})$ from tables $\mathcal{W}(q,\mathcal{C}(q))$ and $\mathcal{W}(t,\mathcal{C}(t))$. Those for (3.2)–(3.5) in $\mathcal{W}(p,\{\nabla\})$ can be shown similarly.

Let $E(\nabla) = \{p=(x,z), q=(x,y), t=(y,z)\}$. By definition, $G[p,\{\nabla\}] = G[q,\mathcal{C}(q)]\cup G[t,\mathcal{C}(t)]\cup\{p\}$ (See Fig. 2). Let $D_q = D\cap V(G[q,\mathcal{C}(q)])$, $D_t = D\cap V(G[t,\mathcal{C}(t)])$, and $D_\nabla = D\cap V(G[p,\{\nabla\}])$. Then, $D_\nabla = D_q\cup D_t$, by definition.

For any tree $M\subseteq G$ and $u,v\in V(M)$, let $\delta_M(u,v) = \sum_{e\in E(P_M(u,v))}\delta(e)$, where $P_M(u,v)$ is the path of M connecting u and v, and let $\delta_M(u) = \max_{d\in D\cap V(M_u)}\delta_{M_u}(u,d)$, where M_u is the connected component of M including u and we define $\delta_M(u) = -\infty$ if $D\cap V(M_u)=\emptyset$.

We show the computational time of (3.1) to be $\mathcal{O}(\Delta)$ for any $(\xi_x,\xi_{xz})\in\mathbb{S}^2_\Delta$.

Case 1: There exists an $H^{[p,\{\nabla\}]}_{\bullet\!-\!\bullet}$.

Let H be an $H^{[p,\{\nabla\}]}_{\bullet\!-\!\bullet}$ with the minimum cost. Then, H includes both x and y,

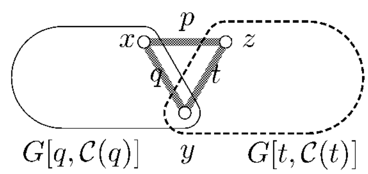
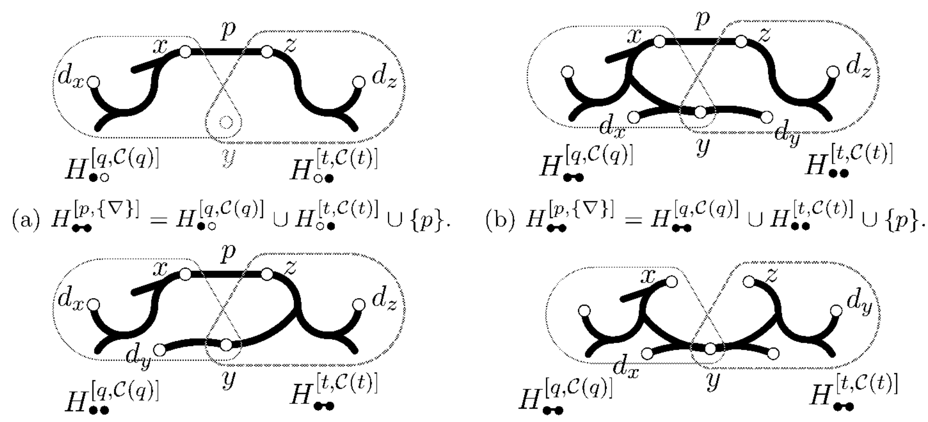


Fig. 2. $G[p, \{\nabla\}] = G[q, \mathcal{C}(q)] \cup G[t, \mathcal{C}(t)] \cup \{p\}$.



(a) $H_{\bullet\bullet}^{[p,\{\nabla\}]} = H_{\bullet\circ}^{[q,\mathcal{C}(q)]} \cup H_{\circ\bullet}^{[t,\mathcal{C}(t)]} \cup \{p\}$. (b) $H_{\bullet\bullet}^{[p,\{\nabla\}]} = H_{\bullet\bullet}^{[q,\mathcal{C}(q)]} \cup H_{\bullet\bullet}^{[t,\mathcal{C}(t)]} \cup \{p\}$.

(c) $H_{\bullet\bullet}^{[p,\{\nabla\}]} = H_{\bullet\bullet}^{[q,\mathcal{C}(q)]} \cup H_{\bullet\bullet}^{[t,\mathcal{C}(t)]} \cup \{p\}$. (d) $H_{\bullet\bullet}^{[p,\{\nabla\}]} = H_{\bullet\bullet}^{[q,\mathcal{C}(q)]} \cup H_{\bullet\bullet}^{[t,\mathcal{C}(t)]}$.

Fig. 3. $H_{\bullet\bullet}^{[p,\{\nabla\}]}$.

$\delta_H(x) \le \xi_x$, $\delta_H(x, z) \le \xi_{xz}$, and $\gamma(H) = W_{\bullet\bullet}(p, \{\nabla\}; \xi_x, \xi_{xz})$. Let $H_q = H \cap G[q, \mathcal{C}(q)]$ and $H_t = H \cap G[t, \mathcal{C}(t)]$. Since $H \subseteq G[p, \{\nabla\}]$ is a tree including both x and y, there exists exactly one path connecting x and y on H. We distinguish the proof into the following four cases.

Case 1-1: $p \in E(H_{\bullet\bullet}^{[p,\{\nabla\}]})$ and $y \notin V(H_{\bullet\bullet}^{[p,\{\nabla\}]})$.
In this case, H_q is an $H_{\bullet\circ}^{[q,\mathcal{C}(q)]}$, H_t is an $H_{\circ\bullet}^{[t,\mathcal{C}(t)]}$, $\delta(p) \le \xi_{xz}$, and

$$\begin{aligned} W_{\bullet\bullet}(p, \{\nabla\}; \xi_x, \xi_{xz}) = \min\big\{ & W_{\bullet\circ}(q, \mathcal{C}(q); \xi'_x) + W_{\circ\bullet}(t, \mathcal{C}(t); \xi''_z) + \gamma(p) \; \big| \\ & \xi_x \ge \max\{\xi'_x, \xi''_z + \delta(p)\}, \xi_{xz} \ge \delta(p) \big\} \end{aligned} \tag{3.6}$$

(See Fig. 3 (a)). Let $d_x \in D_q$ be a destination satisfying $\delta_{H_q}(x) = \delta_{H_q}(x, d_x)$ if $D_q \ne \emptyset$ and $d_z \in D_t$ be a destination satisfying $\delta_{H_t}(z) = \delta_{H_t}(z, d_z)$ if $D_t \ne \emptyset$. Since $p \in E(H)$ and $y \notin V(H)$, we have

$$\begin{aligned} \delta_H(x) &= \max\{\max_{d'_x \in D_q} \delta_{H_q}(x, d'_x), \max_{d'_z \in D_t} \delta_{H_t}(z, d'_z) + \delta(p)\} \\ &= \max\{\delta_{H_q}(x, d_x), \delta_{H_t}(z, d_z) + \delta(p)\}, \end{aligned} \tag{3.7}$$

where $\delta_{H_q}(x, d_x) = -\infty$ if such d_x does not exists and $\delta_{H_t}(z, d_z) = -\infty$ if such d_z does not exists.

Since H_q is an $H^{[q,\mathcal{C}(q)]}_{\bullet\circ}$ and $\delta_{H_q}(x) \le \xi_x$, we have $W_{\bullet\circ}(q,\mathcal{C}(q);\xi_x) \le \gamma(H_q)$. By the similar arguments, we also have $W_{\circ\bullet}(t,\mathcal{C}(t);\xi_x - \gamma(p)) \le \gamma(H_t)$. Thus from Lemma 3.2 and (3.7), for $\xi'_x = \xi_x$ and $\xi''_z = \xi_x - \gamma(p)$, we have

$$\begin{aligned} W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_{xz}) &= \gamma(H) \\ &= \gamma(H_q) + \gamma(H_t) + \gamma(p) \\ &\ge W_{\bullet\circ}(q,\mathcal{C}(q);\xi'_x) + W_{\circ\bullet}(t,\mathcal{C}(t);\xi''_z) + \gamma(p). \end{aligned}$$

From (3.6), the optimality of H, the inequality holds with equality, that is,

$$W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_{xz}) = W_{\bullet\circ}(q,\mathcal{C}(q);\xi'_x) + W_{\circ\bullet}(t,\mathcal{C}(t);\xi''_z) + \gamma(p). \tag{3.8}$$

Therefore, $W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_{xz})$ can be computed in $\mathcal{O}(1)$ time by computing (3.8).

Case 1-2: $p \in E(H^{[p,\{\nabla\}]}_{\bullet\bullet})$, $y \in V(H^{[p,\{\nabla\}]}_{\bullet\bullet})$, and $H \cap G[q,\mathcal{C}(q)]$ is connected.
In this case, $H \cap G[t,\mathcal{C}(t)]$ is disconnected (See Fig. 3 (b)), and

$$\begin{aligned} W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_{xz}) = \min\big\{ & W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_{xy}) + W_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_y,\xi''_z) + \gamma(p) \;\big|\; \\ & \xi_x \ge \max\{\xi'_x, \xi''_z + \delta(p), \xi'_{xy} + \xi''_y\}, \xi_{xz} \ge \delta(p)\big\}. \end{aligned}$$

By the similar arguments to (3.8) in Case 1-1, assuming $\xi'_x = \xi_x$, $\xi''_z = \xi_x - \delta(p)$, and $\xi'_{xy} = \xi_x - \xi''_y$, we have

$$\begin{aligned} & W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_{xz}) \\ & = \min\{W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_{xy}) + W_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_y,\xi''_z) + \gamma(p) \;\big|\xi''_y \in \mathbb{S}_\Delta\}. \end{aligned}$$

$W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_{xz})$ can be obtained in $\mathcal{O}(\Delta)$ time since we only need to check the value of $W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_{xy}) + W_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_y,\xi''_z) + \gamma(p)$ for all $\xi''_y \in \mathbb{S}_\Delta$.

Case 1-3: $p \in E(H^{[p,\{\nabla\}]}_{\bullet\bullet})$, $y \in V(H^{[p,\{\nabla\}]}_{\bullet\bullet})$, and $H \cap G[t,\mathcal{C}(t)]$ is connected.
In this case, $H \cap G[q,\mathcal{C}(q)]$ is disconnected (See Fig. 3 (c)). Then, by the similar argument to Case 1-2,

$$\begin{aligned} & W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_{xz}) \\ & \ge \min\big\{W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_y) + \overline{W}_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_z,\xi''_{yz}) + \gamma(p) \;\big| \\ & \qquad\qquad \xi_x \ge \max\{\xi'_x, \delta(p) + \xi''_z, \delta(p) + \xi''_{yz} + \xi'_y\}, \xi_{xz} \ge \delta(p)\big\} \end{aligned}$$

can be computed in $\mathcal{O}(\Delta)$ time. In fact, this can be done by checking $W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_y) + \overline{W}_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_z,\xi''_{yz}) + \gamma(p)$ only for all $\xi''_{yz} \in \mathbb{S}_\Delta$ by putting $\xi'_x = \xi_x$, $\xi''_z = \xi_x - \delta(p)$, $\xi'_y = \xi_x - \delta(p) - \xi''_{yz}$.

Case 1-4: $p \notin E(H^{[p,\{\nabla\}]}_{\bullet\bullet})$ (See Fig. 3 (d)).
In this case, we can similarly compute

$$\begin{aligned} & W_{\bullet\bullet}(p,\{\nabla\};\xi_x,\xi_{xz}) \;\ge \\ & \quad \min\big\{W_{\bullet\bullet}(q,\mathcal{C}(q);\xi'_x,\xi'_{xy}) + W_{\bullet\bullet}(t,\mathcal{C}(t);\xi''_y,\xi''_{yz}) \;\big| \\ & \qquad \xi_x \ge \max\{\xi'_x, \xi'_{xy} + \xi''_y\}, \xi_{xz} \ge \xi'_{xy} + \xi''_{yz}\big\}. \end{aligned}$$

in $\mathcal{O}(\Delta)$ time since we only need to consider the cases that $\xi'_x = \xi_x$, $\xi''_y = \xi_x - \xi'_{xy}$, and $\xi''_{yz} = \xi_{xz} - \xi'_{xy}$ for all $\xi'_{xy} \in \mathbb{S}_\Delta$.

Case 2: If such H does not exist for $(\xi_x, \xi_{xz}) \in \mathbb{S}^2_\Delta$, all four equations in right hand side of (3.1) are ∞ and it is verified that the computational time of $W_{\bullet\bullet}(p, \{\nabla\}; \xi_x, \xi_{xz})$ to be $\mathcal{O}(\Delta)$ as mentioned above.

Since $W_{\bullet\bullet}(p, \{\nabla\}; \xi_x, \xi_{xz})$ can be computed in $\mathcal{O}(\Delta)$ time for each $(\xi_x, \xi_{xz}) \in \mathbb{S}^2_\Delta$, those for all $(\xi_x, \xi_{xz}) \in \mathbb{S}^2_\Delta$ can be computed in $\mathcal{O}(\Delta^3)$ time. ■

Lemma 3.6. *Let $p \in E(G)$, $\mathcal{C}(p) = \{\nabla_1, \nabla_2, \ldots, \nabla_{|\mathcal{C}(p)|}\}$, and $\mathcal{C}^{(i)}(p) = \{\nabla_1, \nabla_2, \ldots, \nabla_i\}$ for $1 \le i \le |\mathcal{C}(p)|$. Given $\mathcal{W}(p, \mathcal{C}^{(i-1)}(p))$ and $\mathcal{W}(p, \{\nabla_i\})$, $\mathcal{W}(p, \mathcal{C}^{(i)}(p))$ is computed in $\mathcal{O}(\Delta^3)$ time for $2 \le i \le |\mathcal{C}(p)|$.* ■

Lemma 3.6 can be shown in a similar way to the proof of Lemma 3.5.

Theorem 3.1. For a 2-tree G on n vertices, $\mathrm{BA}(G, s, D, \gamma, \delta, \Delta)$ computes $\mathcal{W}(r, \mathcal{C}(r))$ in $\mathcal{O}(n\Delta^3)$ time.

Proof. The tables $\mathcal{W}(p, \emptyset)$ for all leaves p can be computed in $\mathcal{O}(n\Delta^2)$ time by Lemma 3.4. Since the number of triangles is $\mathcal{O}(n)$, the tables $\mathcal{W}(p, \{\nabla\})$ for all triangles ∇ can be computed in $\mathcal{O}(n\Delta^3)$ time by Lemma 3.5. By Lemma 3.6, $\mathcal{W}(p, \mathcal{C}(p))$ can be computed in $\mathcal{O}(|\mathcal{C}(p)|\Delta^3)$ time. Since $\sum_{p \in E(G)} |\mathcal{C}(p)| = \mathcal{O}(n)$, the tables $\mathcal{W}(p, \mathcal{C}(p))$ for all edges p can be computed in $\mathcal{O}(n\Delta^3)$ time. It follows that $\mathrm{BA}(G, s, D, \gamma, \delta, \Delta)$ computes $\mathcal{W}(r, \mathcal{C}(r))$ in $\mathcal{O}(n\Delta^3)$ time by Lemma 3.3. ■

By Lemma 3.1 and Theorem 3.1, $\mathrm{BA}(G, s, D, \gamma, \delta, \Delta)$ computes the minimum cost of a multicast tree with delay at most ξ for any $\xi \in \{0, 1, \ldots, \Delta\}$. If we perform some bookkeeping operations such as recording how the minimum was achieved during the computation of the tables for functions, we can construct a delay-constrained minimum cost multicast tree in the same time complexity. Thus, we have the following.

Corollary 3.1. *Given a 2-tree G on n vertices, s, D, γ, δ, Δ, and an integer ξ, $0 \le \xi \le \Delta$, a minimum cost multicast tree T with $\delta(T) \le \xi$ can be constructed in $\mathcal{O}(n\Delta^3)$ time.* ■

We denote by $\mathrm{MT}(G, s, D, \gamma, \delta, \Delta, \xi)$ such an $\mathcal{O}(n\Delta^3)$ time algorithm constructing a minimum cost multicast tree T with $\delta(T) \le \xi$ for a given 2-tree G, s, D, γ, δ, Δ, and an integer ξ, $0 \le \xi \le \Delta$.

3.5. *Pseudo-Polynomial Time Algorithm for DCMCMT*

Given a connected series-parallel graph G' with cost and delay functions γ' and δ', we denote by $\mathrm{E{\scriptsize XT}}(G', \delta', \gamma')$ a linear time procedure for augmenting G' to a 2-tree G with $V(G) = V(G')$ [8], and extending γ' and δ' to γ and δ, respectively, by defining $\gamma(e) = \infty$ and $\delta(e) = \infty$ for each $e \in E(G) - E(G')$, and $\gamma(e) = \gamma'(e)$ and

$\delta(e) = \delta'(e)$ for each $e \in E(G')$. Then, it is easy to see that Algorithm 1 shown in Fig. 4. solves DCMCMT for series-parallel graphs, and we have the following by Theorem 3.1.

Theorem 3.2. For a series-parallel graph G with n vertices and a positive integer Δ, Algorithm 1 solves DCMCMT in $\mathcal{O}(n\Delta^3)$ time. ∎

```
Input  a series-parallel graph G', s ∈ V(G'), D ⊆ V(G'),
       γ' : E(G') → ℕ, δ' : E(G') → ℕ, Δ ∈ ℤ⁺.
Output a minimum cost multicast tree T with delay at
       most Δ.
begin
       EXT(G', γ', δ');
       BA(G, s, D, γ, δ, Δ);
       MT(G, s, D, γ, δ, Δ, Δ);
       if γ(T) < ∞
         return T;
       else
         return "NO";
       endif
end
```

Fig. 4. Algorithm 1.

3.6. *Pseudo-Polynomial Time Algorithm for CCMDMT*

Given a cost bound Γ and the table $\mathcal{W}(r, \mathcal{C}(r))$ for functions, we denote by $\textsc{Min\_Delay}(\Gamma, \mathcal{W}(r, \mathcal{C}(r)))$ a linear time procedure for computing the minimum ξ satisfying $\min\{W_{\bullet\bullet}(r, \mathcal{C}(r); \xi, \Delta), W_{\bullet\circ}(r, \mathcal{C}(r); \xi)\} \leq \Gamma$ if exists. It returns ∞ if there exists no such ξ.

Since the number of edges of multicast tree is at most $n-1$, the maximum delay of a multicast tree is at most $(n-1)\delta_{\max}$, where $\delta_{\max} = \max_{e \in E(G')} \delta'(e)$. Thus, it is easy to see that Algorithm 2 shown in Fig. 5 is a pseudo-polynomial time algorithm for CCMDMT, and we have the following by Theorem 3.1.

Theorem 3.3. For a series-parallel graph G with n vertices and a non-negative integer Γ, Algorithm 2 solves CCMDMT in $\mathcal{O}(n^4 {\delta_{\max}}^3)$ time if $\delta_{\max} \geq 1$. ∎

4. FPTAS for CCMDMT

We use standard techniques [2,3,5,9] to turn $\mathrm{BA}(G, s, D, \gamma, \delta, \Delta)$ into an FPTAS for CCMDMT. We show in Section 4.1 a pair of upper and lower bounds U and L for

Input a series-parallel graph G',
$s \in V(G')$, $D \subseteq V(G')$,
$\gamma' : E(G') \to \mathbb{N}$, $\delta' : E(G') \to \mathbb{N}$, $\Gamma \in \mathbb{Z}^+$.
Output a minimum delay multicast tree T with cost at most Γ.
begin
$\delta_{\max} := \max_{e \in E(G')} \delta'(e)$;
$\Delta' := (n-1)\delta_{\max}$;
$\textsc{Ext}(G', \gamma', \delta')$;
$\mathrm{BA}(G, s, D, \gamma, \delta, \Delta')$;
$\textsc{Min\_Delay}(\Gamma, \mathcal{W}(r, \mathcal{C}(r)))$;
if $\xi < \infty$
$\mathrm{MT}(G, s, D, \gamma, \delta, \Delta, \xi)$;
return T;
else
return "NO";
endif
end

Fig. 5. Algorithm 2.

the minimum delay of a cost constrained multicast tree such that $U/L \leq n-1$. For any $\varepsilon > 0$, we show in Section 4.2 a $(1+\varepsilon)$-approximation algorithm for CCMDMT. The algorithm runs in $\mathcal{O}(n^7/\varepsilon^3)$ time, provided that we have a pair of upper and lower bounds U and L for the delay of a cost constrained multicast tree such that $U/L = \mathcal{O}(n)$. It follows that we have an FPTAS for CCMDMT.

4.1. *Upper and Lower Bounds for Minimum Delay*

We use a technique similar to [2]. Let $\nu_1 < \nu_2 < \cdots < \nu_k$ be different edge delays, and γ_j be the cost function defined as $\gamma_j(e) = \gamma(e)$ if $\delta(e) \leq \nu_j$, and $\gamma_j(e) = \infty$ otherwise. Let T_j be a minimum cost multicast tree of G for γ_j, and J be the minimum j such that $\gamma_j(T_j) \leq \Gamma$.

By the definition of J, the minimum delay of a cost constrained multicast tree is at least ν_J and at most $(n-1)\nu_J$. Since such J and also T_J can be computed in $\mathcal{O}(n \log n)$ time [2], we have the following.

Theorem 4.1. A pair of upper and lower bounds U and L for the minimum delay of a cost constrained multicast tree satisfying $U/L = n-1$ can be computed in $\mathcal{O}(n \log n)$ time. Moreover, a multicast tree T_J with cost at most Γ and delay at most U can also be computed in $\mathcal{O}(n \log n)$ time. ∎

Given a 2-tree G with source s and destinations D, cost and delay functions γ and δ, and a positive integer Γ, we denote by $\textsc{Comp\_UL}(G, s, D, \gamma, \delta, \Gamma)$ an $\mathcal{O}(n \log n)$

time procedure for computing upper and lower bounds U and L with $U/L \leq n-1$.

4.2. *FPTAS for CCMDMT*

For any $\alpha > 0$, let δ_α be a delay function defined as $\delta_\alpha(e) = \lfloor \alpha\delta(e) \rfloor$ for any $e \in E(G)$. Let T_α be a minimum delay multicast tree with cost at most Γ for δ_α and $\mathrm{OPT}(\delta_\alpha) = \delta_\alpha(T_\alpha)$. Notice that T_1 is a minimum delay multicast tree with cost at most Γ for $\delta = \delta_1$. We denote by P_α a maximum delay path in T_α for δ_α.

By the definition of δ_α, we have

$$\delta(e) \geq \frac{1}{\alpha}\delta_\alpha(e) \quad \text{and} \tag{4.1}$$

$$\delta(e) < \frac{1}{\alpha}(\delta_\alpha(e) + 1) \tag{4.2}$$

for any $e \in E(G)$. If we denote by P_1' a maximum delay path of T_1 for δ_α,

$$\begin{aligned}
\mathrm{OPT}(\delta) &= \sum_{e \in E(P_1)} \delta(e) \\
&\geq \sum_{e \in E(P_1')} \delta(e) \\
&\geq \sum_{e \in E(P_1')} \frac{1}{\alpha}\delta_\alpha(e) && (4.3) \\
&\geq \sum_{e \in E(P_\alpha)} \frac{1}{\alpha}\delta_\alpha(e) \\
&\geq \frac{1}{\alpha}\delta_\alpha(T_\alpha) \\
&= \frac{1}{\alpha}\mathrm{OPT}(\delta_\alpha). && (4.4)
\end{aligned}$$

where (4.3) follows from (4.1).

Moreover, if we set $\alpha = (n-1)/\varepsilon L$, and denote by P_α' a maximum delay path in T_α for δ, we have

$$\begin{aligned}
\delta(T_\alpha) &= \sum_{e \in E(P'_\alpha)} \delta(e) \\
&< \frac{1}{\alpha} \sum_{e \in E(P'_\alpha)} (\delta_\alpha(e) + 1) \qquad (4.5) \\
&\leq \frac{1}{\alpha}|E(P'_\alpha)| + \frac{1}{\alpha} \sum_{e \in E(P'_\alpha)} \delta_\alpha(e) \\
&\leq \frac{n-1}{\alpha} + \frac{1}{\alpha} \sum_{e \in E(P_\alpha)} \delta_\alpha(e) \\
&= \varepsilon L + \frac{1}{\alpha}\mathrm{OPT}(\delta_\alpha) \qquad (4.6) \\
&\leq \varepsilon L + \mathrm{OPT}(\delta) \qquad (4.7) \\
&\leq (1+\varepsilon)\mathrm{OPT}(\delta),
\end{aligned}$$

where inequality (4.5) and (4.7) follow from (4.2) and (4.4), respectively.

Thus, we conclude that Algorithm 3 shown in Fig. 6 is an FPTAS for CCMDMT. Since $\Delta_\alpha = (n-1)U/\varepsilon L$, we have the following by Theorem 3.2.

Input a series-parallel graph G', $s \in V(G')$, $D \subseteq V(G')$, $\gamma' : E(G') \to \mathbb{N}$, $\delta' : E(G') \to \mathbb{N}$, $\Gamma \in \mathbb{Z}^+$, $\varepsilon > 0$.
Output a multicast tree T with cost at most Γ and delay at most $(1+\varepsilon)\mathrm{OPT}(\delta')$.

begin
  $\textsc{Ext}(G', \gamma', \delta')$;
  $\textsc{Comp\_UL}(G, s, D, \gamma, \delta, \Gamma)$;
  $\alpha := (n-1)/\varepsilon L$;
  $\delta_\alpha(e) := \lfloor \alpha\delta(e) \rfloor \ \forall e \in E(G)$;
  $\Delta_\alpha := \alpha U$;
  $\mathrm{BA}(G, s, D, \gamma, \delta_\alpha, \Delta_\alpha)$;
  $\textsc{Min\_Delay}(\Gamma, \mathcal{W}(r, \mathcal{C}(r)))$;
  $\mathrm{MT}(G, s, D, \gamma, \delta_\alpha, \Delta_\alpha, \xi)$;
  return T;
end

Fig. 6. Algorithm 3.

Lemma 4.1. *For a series-parallel graph G with n vertices and a non-negative integer Δ, Algorithm 3 computes a $(1+\varepsilon)$-approximate solution for CCMDMT in*

$\mathcal{O}\left(n(nU/\varepsilon L)^3\right)$ *time if we are given an upper bound* U *and a lower bound* L *of CCMDMT.* ∎

If $U/L = \mathcal{O}(n)$, $\mathcal{O}\left(n(nU/\varepsilon L)^3\right) = \mathcal{O}\left(n^7/\varepsilon^3\right)$. Thus, from Theorem 4.1 and Lemma 4.1, we have the following.

Theorem 4.2. A $(1+\varepsilon)$-approximate solution for CCMDMT can be obtained in $\mathcal{O}(n \log n + n^7/\varepsilon^3)$ time. ∎

5. Concluding Remarks

- The time complexity in Theorem 4.2 can be reduced to $\mathcal{O}(n^4/\epsilon^3 + n^3)$ by adopting a well-known scaling and rounding technique used in [2,3,5,9]. The proof is rather complicated and omitted here.
- It should be noted that our method to obtain FPTAS for CCMDMT cannot apply to DCMCMT in a straightforward way, since Δ can be exponentially large.
- The approximability of DCMCMT and CCMDMT for general graphs, and that of DCMCMT for series-parallel graphs are interesting open problems.

References

1. G. Chen, G. Xue, k-pair delay constrained minimum cost routing in undirected networks, ACM-SIAM Symposium on Discrete Algorithms (2001) 230–231.
2. G. Chen, G. Xue, A PTAS for weight constrained steiner trees in series-parallel graphs, Theoretical Computer Science 304 (2003) 237–247.
3. R. Hassin, Approximation schemes for the restricted shortest path problem, Mathematics of Operations Research 17 (2) (1992) 36–42.
4. V. Kompella, J. Pasquale, G. Polyzos, Multicast routing for multimedia communication, IEEE/ACM Transactions on Networking 1 (1993) 286–292.
5. D. Lorenz, D. Raz, A simple efficient approximation scheme for the restricted path problem, Operations Research Letters 28 (2001) 213–219.
6. M. Parsa, Q. Zhu, J. J. G.-L.-Aceves, An iterative algorithm for delay-constrained minimum-cost multicast, IEEE/ACM Trans. Networking 6 (2001) 213–219.
7. R. Sriram, G. Manimaran, C. Murthy, Algorithms for delay-constrained low-cost multicast tree construction, Computer Communications 21 (1998) 1693–1706.
8. J. Wald, C. Colbourn, Steiner trees, partial 2-trees and minimum IFI networks, Networks 13 (1983) 159–167.
9. A. Warburton, Approximation of pareto optima in multiple-objective shortest path problems, Operations Research 35 (1987) 70–79.
10. H. Youssef, A. A.-Mulhem, S. Sait, M. Tahir, QoS-driven multicast tree generation using tabu search, Computer Communications 25 (2002) 1140–1149.