



The Heterogeneous Rooted Tree Cover Problem

Pengxiang Pan¹, Junran Lichen², Ping Yang¹, and Jianping Li¹(✉)

¹ School of Mathematics and Statistics, Yunnan University,
East Outer Ring South Road, University Town,
Kunming 650504, People's Republic of China
{pengxiang,jianping}@ynu.edu.cn

² School of Mathematics and Physics, Beijing University of Chemical Technology,
No.15, North Third Ring East Road, Chaoyang District,
Beijing 100029, People's Republic of China
J.R.Lichen@buct.edu.cn

Abstract. We consider the heterogeneous rooted tree cover (HRTC) problem. Concretely, given an undirected complete graph $G = (V, E)$ with a root $r \in V$, an edge-weight function $w : E \rightarrow R^+$ satisfying the triangle inequality, a vertex-weight function $f : V \setminus \{r\} \rightarrow R_0^+$, and k construction teams having nonuniform construction speeds $\lambda_1, \lambda_2, \dots, \lambda_k$, we are asked to find k trees for these k construction teams to cover all vertices in V , each tree starting at the same root r , *i.e.*, k trees having a sole common vertex called root r , the objective is to minimize the maximum completion time, where the completion time of each team is the total construction weight of its related tree divided by its construction speed.

In this paper, we first design a $58.3286(1 + \delta)$ -approximation algorithm to solve the HRTC problem in time $O(n^3(1 + \frac{1}{\delta}) + \log(w(E) + f(V \setminus \{r\})))$ for any $\delta > 0$. In addition, we present a $\max\{2\rho, 2 + \rho - \frac{2}{k}\}$ -approximation algorithm for resolving the HRTC problem in time $O(n^2)$, where ρ is the ratio between the maximum and minimum speed of these k teams.

Keywords: Rooted tree cover · Nonuniform speeds · Approximation algorithms · Complexity of algorithms

1 Introduction

The subgraph cover problems, including the cycle cover problem and the tree cover problem, form a much-studied family of combinatorial optimization problems. These problems have wide range of practical applications, such as routings of multi-vehicles [2, 7, 12], nurse station location [4] and data gathering in wireless

This paper is supported by the National Natural Science Foundation of China [Nos. 12361066, 12101593]. Junran Lichen is also supported by Fundamental Research Funds for the Central Universities [No.buctrc202219], and Jianping Li is also supported by Project of Yunling Scholars Training of Yunnan Province [No. K264202011820].

sensor networks [13, 15]. In some applications, the minimization of the latest service completion time at the service locations is more relevant. As a result, there is a growing body of literature on cover problems under the min-max objective.

Considering the vehicles to have nonuniform speeds in routing planning, Gørtz et al. [8] in 2016 proposed the heterogeneous traveling salesman problem, which we refer to as the heterogeneous rooted cycle cover (HRCC) problem. In the HRCC problem, given a complete graph $G = (V, E)$ equipped with an edge-weight function $w : E \rightarrow R^+$ that satisfies the triangle inequality, a root $r \in V$ and k vehicles with nonuniform speeds $\lambda_1, \lambda_2, \dots, \lambda_k$, we are asked to find k cycles for these k vehicles to cover all vertices in V , each vehicle starting at r , *i.e.*, these k cycles having a sole common vertex r , the objective is to minimize the maximum completion time, where the completion time of a vehicle is the total weight of its related cycle divided by its speed. For the HRCC problem, Gørtz et al. [8] in 2016 presented a constant factor approximation algorithm.

The rooted cycle cover (RCC) problem [6], also called the k -traveling salesman problem, which is an important special version of the HRCC problem, where $\lambda_i = 1$ for each $i \in \{1, 2, \dots, k\}$. Employing a splitting strategy, Frederickson et al. [6] in 1978 gave a $(\frac{5}{2} - \frac{1}{k})$ -approximation algorithm to solve the RCC problem. For the version $k = 1$ of the RCC problem, *i.e.*, the metric traveling salesman problem, Christofides [3] provided a famous $3/2$ -approximation algorithm by using an algorithm for solving the Euler tour problem.

In addition, many researches focus on the tree cover problems of graphs, in which vertices are all covered by a set of k trees. Taking the service handling times of vertices into consideration, Nagamochi [10] proposed the rooted tree cover (RTC) problem, which is modelled as follows. Given a complete graph $G = (V, E)$ equipped with an edge-weight function $w : E \rightarrow R^+$ satisfying the triangle inequality, a vertex-weight function $f : V \setminus \{r\} \rightarrow R_0^+$, a root $r \in V$ and k construction teams, it is asked to find k trees for these k teams to cover all vertices in V , each tree starting at the same root r , the objective is to minimize the maximum total weight among these k trees, where the total weight of a tree is the summation of edge weights and vertex weights in that tree, equivalently, the objective is to minimize the maximum completion time, where the completion time of a construction team is the total construction weight of that tree divided by its speed for the case that speeds of these k teams are all same one.

The RTC problem is *NP*-hard [1] even for the case $k = 2$ and $f(\cdot) \equiv 0$. Many research papers have been focused on the development of constant factor approximation algorithms to resolve the RTC problem. Using a tree partition technique, Nagamochi [10] in 2005 presented a $(3 - \frac{2}{k+1})$ -approximation algorithm to resolve the RTC problem. Xu and Wen [14] in 2010 gave a lower bound of $10/9$ for the RTC problem. Moreover, the other relevant results can be found in [5, 9, 11, 16].

In practice, the construction efficiencies or construction speeds of multiple construction teams are often different similar to the vehicle speeds of the HRCC problem. Motivated by the observation and the RTC problem, we address the heterogeneous rooted tree cover (HRTC) problem. Concretely, given an undirected complete graph $G = (V, E; w, f)$ with a root $r \in V$, an edge-weight

function $w : E \rightarrow R^+$ satisfying the triangle inequality, a vertex-weight function $f : V \setminus \{r\} \rightarrow R_0^+$, and k construction teams having nonuniform construction speeds $\lambda_1, \lambda_2, \dots, \lambda_k$, we are asked to find k trees $\mathcal{T} = \{T_i \mid i = 1, 2, \dots, k\}$ for these k construction teams to cover all vertices in V , each starting at the same root r , *i.e.*, k trees having a sole common vertex called root r , the objective is to minimize the maximum completion time, where the completion time of each team is the total construction weight of its related tree divided by its construction speed. In a formulaic way, the min-max objective is written as $\min_{\mathcal{T}} \max\{\frac{w(T_i)+f(T_i)}{\lambda_i} \mid i = 1, 2, \dots, k\}$.

As far as what we have known, the HRTC problem has not been considered in the literature. The aforementioned HRCC problem (without vertex weights) has been studied in [8], but we cannot directly use the algorithms for the HRCC problem to solve the HRTC problem, because the HRTC problem includes the vertex weights. However, modifying the technique in [8], we intend to design first approximation algorithm with constant approximation ratio to solve the HRTC problem. In addition, we shall present second approximation algorithm with lower time complexity to resolve the HRTC problem.

The remainder of this paper is organized as follows. In Sect. 2, we present some terminologies and fundamental lemmas to state descriptions of approximation algorithms; In Sect. 3, we design first constant factor approximation algorithm to solve the HRTC problem; In Sect. 4, we design second approximation algorithm with lower time complexity to resolve the HRTC problem; In Sect. 5, we provide our conclusion and further research.

2 Terminologies and Fundamental Lemmas

All graphs considered in the paper are assumed to be finite, undirected and loopless. Given a graph $G = (V, E)$, to contract a vertex subset $V' \subseteq V$ is to replace these vertices by a single vertex incident to all the edges which were incident in G to any vertex in V' . The resulting graph is denoted by G/V' with vertex set $V \cup \{v'\} \setminus V'$ and edge set $E \cup \{uv' \mid uv \in E, u \in V \setminus V', v \in V'\} \setminus E(G[V'])$, where v' is viewed as a new vertex obtained by contracting the vertex subset V' . For a vertex set V and a set $\mathcal{T} = \{T_i \mid i = 1, 2, \dots, k\}$ of trees (or cycles), if $V \subseteq \bigcup_{i=1}^k V(T_i)$, we say that \mathcal{T} covers V .

For any two sets X_1 and X_2 , $X_1 + X_2$ is a multiset obtained by adding all elements in $X_1 \cap X_2$ to $X_1 \cup X_2$. Especially, for any two graphs $G = (V, E)$ and $G' = (V', E')$, denote $G \cup G' = (V \cup V', E \cup E')$ and $G + G' = (V \cup V', E + E')$, respectively.

In designing a constant factor approximation algorithm for the HRTC problem, we need the following definition, which is obtained by slightly modifying the definition in [8].

Definition 1. *Given an undirected graph $G = (V, E; w, f)$ with two constants $M > 0$ and $\varepsilon > 0$, where $w : E \rightarrow R^+$ is an edge-weight function and $f : V \setminus \{r\} \rightarrow R_0^+$ is a vertex-weight function, let \mathcal{F}_i be a set of trees in G starting at the same vertex r for each integer $i \geq 0$. Then the collection $\{\mathcal{F}_i\}_{i \geq 0} = \bigcup_{i \geq 0} \mathcal{F}_i$ is referred to as $(\alpha, \beta)_{M, \varepsilon}$ -assignable, if it has the following properties*

- (1) $w(F) + f(F) \leq \alpha \cdot (1 + \varepsilon)^i M$, for each tree $F \in \mathcal{F}_i$, $i \geq 0$;
- (2) $\sum_{j \geq i} (w(\mathcal{F}_j) + f(\mathcal{F}_j)) \leq \beta M \cdot \Lambda((1 + \varepsilon)^{i-1})$ for each $i \geq 0$, where $\Lambda((1 + \varepsilon)^{i-1})$ is the sum of speeds that at least $(1 + \varepsilon)^{i-1}$.

For Definition 1, we can regard the sum of edge weights and vertex weights of each tree as a whole weight, and using the ASSIGN algorithm in Gørtz et al. [8], we can obtain the following important result.

Lemma 1. [8] *Given an $(\alpha, \beta)_{M, \varepsilon}$ -assignable collection $\{\mathcal{F}_i\}_{i \geq 0}$ of trees starting at r , we can use the ASSIGN algorithm to assign all trees in $\{\mathcal{F}_i\}_{i \geq 0}$ to k construction teams in time $O(n^3 + \log(w(E) + f(V \setminus \{r\})))$, satisfying that the completion time of each construction team is at most $((1 + \varepsilon)\alpha + \beta)M$, where the completion time of each team is the total construction weight divided by its construction speed, n is the number of vertices.*

3 An Approximation Algorithm with Constant Approximation Ratio

In this section, we consider the heterogeneous rooted tree cover (HRTC) problem. Without loss of generality, we may assume that the considered graph are all connected and $2 \leq k \leq n - 1$.

By Lemma 1, we design the following strategies to solve the HRTC problem.

- (1) Find an $(\alpha, \beta)_{M, \varepsilon}$ -assignable collection of subtrees to cover all vertices;
- (2) Assign subtrees in the above collection to k construction teams to minimize the completion time of any team.

Given an undirected graph $G = (V, E; w, f)$ with two current values M and ε (the precise value to be chosen later), we first give a partition of V that is $\{V_0, V_1, \dots\}$, where $V_0 = \{v \in V \mid w(rv) + f(v) \leq M\}$, and $V_i = \{v \in V \mid (1 + \varepsilon)^{i-1}M < w(rv) + f(v) \leq (1 + \varepsilon)^i M\}$ for each $i \geq 1$. For each $i \geq 0$, let $V_{\leq i} = \bigcup_{j \leq i} V_j$ and $V_{\geq i} = \bigcup_{j \geq i} V_j$. Similarly, we give a partition of E that is $\{E_0, E_1, \dots\}$, where $E_i = \{uv \in E \mid u \in V_{\leq i}, v \in V_i\}$ for each $i \geq 0$. For each $i \geq 0$, let $E_{\leq i} = \bigcup_{j \leq i} E_j$ and $E_{\geq i} = \bigcup_{j \geq i} E_j$.

Now, analyzing the lower bound of the HRTC problem, we obtain the following lemma.

Lemma 2. *Given a complete graph $G = (V, E; w, f)$ as an instance of the HRTC problem, for any constant $M \geq OPT$, we have $w(T_{G/V_{<l}}^{MS}) + f(V_{\geq l}) \leq M \cdot \Lambda((1 + \varepsilon)^{l-1})$ for each integer $l \geq 0$, where OPT is the optimal value to the given instance, $T_{G/V_{<l}}^{MS}$ is a minimum edge-weight spanning tree of $G/V_{<l}$, i.e., a new graph obtained by contracting a vertex set $V_{<l}$, and $\Lambda((1 + \varepsilon)^{l-1})$ is the total of speeds that exceed $(1 + \varepsilon)^{l-1}$.*

Proof. Consider that in an optimal solution to G for the HRTC problem, if any vertex $v \in V_{\geq l}$ can be constructed by a construction team with speed λ' , then

we have $\lambda' > (1 + \varepsilon)^{l-1}$. This is because $w(rv) + f(v) > (1 + \varepsilon)^{l-1}M$ holds for each $v \in V_{\geq l}$, and the construction weight of a construction team with speed λ' is at most $\lambda' \cdot OPT \leq \lambda' \cdot M$, implying $\lambda' \cdot M \geq \lambda' \cdot OPT > (1 + \varepsilon)^{l-1}M$. Since $\max\{w(rv) + f(v), w(rv') + f(v')\} > (1 + \varepsilon)^{l-1}M$ holds for each edge $e = vv' \in E_{\geq l}$, we deduce that any edge in $E_{\geq l}$ must be constructed by some team with speed exceed $(1 + \varepsilon)^{l-1}$.

Let E^* denote the set of edges constructed by teams in the optimal solution. By the above arguments, we have $w(E^* \cap E_{\geq l}) + f(V_{\geq l}) \leq OPT \cdot \Lambda((1 + \varepsilon)^{l-1}) \leq M \cdot \Lambda((1 + \varepsilon)^{l-1})$, meaning $w(E^* \cap E_{\geq l}) + f(V_{\geq l}) \leq M \cdot \Lambda((1 + \varepsilon)^{l-1})$. Clearly, $G[E^*]$ is a spanning tree of G , and $E^* \cap E_{\geq l}$ corresponds to a spanning tree of $G/V_{< l}$. Since $T_{G/V_{< l}}^{MS}$ is a minimum edge-weight spanning tree of $G/V_{< l}$, we obtain $w(T_{G/V_{< l}}^{MS}) + f(V_{\geq l}) \leq w(E^* \cap E_{\geq l}) + f(V_{\geq l})$, implying $w(T_{G/V_{< l}}^{MS}) + f(V_{\geq l}) \leq M \cdot \Lambda((1 + \varepsilon)^{l-1})$. \square

For each $i \geq 0$, let H_i denote the edge subset of G corresponding to a minimum spanning tree of $G[V_{\leq i}]/V_{< i}$. Obviously, $G[\mathcal{H}]$ with $\mathcal{H} = \bigcup_{i \geq 0} H_i$ is a spanning tree of G . Using the similar arguments in [8] to analyze the relation between \mathcal{H} and a minimum edge-weight spanning tree of G , we obtain a result as follows.

Lemma 3. [8] *Given a complete graph $G = (V, E; w, f)$ and a constant $\varepsilon > 0$, a spanning tree $G[\mathcal{H}]$ with $\mathcal{H} = \bigcup_{i \geq 0} H_i$ of G can be constructed to satisfy:*

- *The vertex levels along every root-leaf path are nondecreasing.*
- *For each $i \geq 0$, we have $\sum_{j \geq i} w(H_j) \leq (6 + \frac{6}{\varepsilon}) \cdot w(T_{G/V_{< i}}^{MS})$, where $T_{G/V_{< i}}^{MS}$ is a minimum edge-weight spanning tree of $G/V_{< i}$.*

In Lemma 3, for each $i \geq 0$, it is clear that $\sum_{j \geq i} w(H_j) \leq (6 + \frac{6}{\varepsilon}) \cdot w(T_{G/V_{< i}}^{MS})$ means $\sum_{j \geq i} (w(H_j) + f(V_j)) \leq (6 + \frac{6}{\varepsilon}) \cdot w(T_{G/V_{< i}}^{MS}) + f(V_{\geq i}) \leq (6 + \frac{6}{\varepsilon}) \cdot (w(T_{G/V_{< i}}^{MS}) + f(V_{\geq i}))$. By Lemma 2, we obtain at once that $\sum_{j \geq i} (w(H_j) + f(V_j)) \leq (6 + \frac{6}{\varepsilon}) \cdot (w(T_{G/V_{< i}}^{MS}) + f(V_{\geq i})) \leq (6 + \frac{6}{\varepsilon}) \cdot M \cdot \Lambda((1 + \varepsilon)^{i-1})$, which is stated in the following

Lemma 4. *Given a complete graph $G = (V, E; w, f)$ with two constants $\varepsilon > 0$ and $M \geq OPT$, where OPT is the optimal value to the instance G for the HRTC problem, then the spanning tree $G[\mathcal{H}]$ with $\mathcal{H} = \bigcup_{i \geq 0} H_i$ of G mentioned-above satisfies:*

- *The vertex levels along every root-leaf path are nondecreasing.*
- *For each $i \geq 0$, we have $\sum_{j \geq i} (w(H_j) + f(V_j)) \leq (6 + \frac{6}{\varepsilon}) \cdot M \cdot \Lambda((1 + \varepsilon)^{i-1})$.*

To shorten notation, given each edge $e = uv$ in \mathcal{H} , denote $x_e \in \{u, v\}$ to be a vertex farther away from r in $G[\mathcal{H}]$, and $y_e \in \{u, v\}$ to be a vertex closer to r in $G[\mathcal{H}]$. For each subtree $G' = (V', E')$ of $G[\mathcal{H}]$ mentioned above, we define a new function $f_1(\cdot)$ to be $f_1(G') = \sum_{e \in E'} f_1(x_e)$, implying $f_1(G') = f(G') - f(y_{G'})$, where $y_{G'}$ is a vertex in $G[\mathcal{H}]$ closest to r .

Basing from Lemma 1 to Lemma 4, we design a following algorithm, denoted by the algorithm HRTC₁, to solve the HRTC problem.

Algorithm: HRTC₁

INPUT: An undirected complete graph $G = (V, E; w, f)$ with a root $r \in V$, an edge-weight function $w : E \rightarrow R^+$, a vertex-weight function $f : V \setminus \{r\} \rightarrow R_0^+$, k construction teams having speeds $\lambda_1, \dots, \lambda_k$ respectively, a small fixed constant $\delta > 0$, and two constants $\zeta = 2$ and $\varepsilon = 1.3146$ (to be chosen in Theorem 1);

OUTPUT: A set $\mathcal{T} = \{T_i \mid i = 1, 2, \dots, k\}$ of k trees.

Begin

Step 1 Set $M = \max_{v \in V} \left\{ \frac{w(r,v) + f(v)}{\lambda_{\max}} \right\}$, where $\lambda_{\max} = \max\{\lambda_i \mid i = 1, 2, \dots, k\}$;

Step 2 Using M and ε , we can partition the set V into subsets V_0, V_1, \dots , and the set E into subsets E_0, E_1, \dots as mentioned-above; For convenience, we may actually assume that the number of subsets partitioned is t , i.e., $(1 + \varepsilon)^{t-1} M < \max\{w(rv) + f(v) \mid v \in V\} \leq (1 + \varepsilon)^t M$;

Step 3 If $(w(T_{G/V_{<l}^{MS}}) + f(V_{\geq l})) > M \cdot \Lambda((1 + \varepsilon)^{l-1})$ holds for some $l \in \{1, 2, \dots, t\}$ then

set $M := (1 + \delta)M$, and go to Step 2;

Step 4 Construct a spanning tree $G[\mathcal{H}]$ with $\mathcal{H} = \bigcup_{i \geq 0} H_i$ of G , where H_i is the edge subset of G corresponding to a minimum edge-weight spanning tree of $G[V_{\leq i}]/V_{<i}$; Set $\mathcal{S}_0 := \{H_0\}$;

Step 5 For each $i \in \{1, 2, \dots, t\}$, partition H_i into a set \mathcal{S}_i of subtrees such that each subtree $\eta \in \mathcal{S}_i$ contains exactly one edge $h(\eta)$ from $V_{<i}$ to V_i ; Let $\gamma = \frac{\varepsilon}{(2+\varepsilon)(1+\varepsilon)}$ and $\mathcal{S}_0^m = \mathcal{S}_0$; For each $i \in \{1, 2, \dots, t\}$, set $\mathcal{S}_i^m = \emptyset$ and $\mathcal{S}_i^u = \emptyset$;

Step 6 For all $i \in \{1, 2, \dots, t\}, \eta \in \mathcal{S}_i$ do:

If $(w(\eta) + f_1(\eta)) \geq \gamma \cdot (1 + \varepsilon)^i M$ then

Set $\mathcal{S}_i^m := \mathcal{S}_i^m \cup \{\eta\}$;

Else

Set $\mathcal{S}_i^u := \mathcal{S}_i^u \cup \{\eta\}$;

Step 7 For all $i \in \{1, 2, \dots, t\}, \sigma \in \mathcal{S}_i^u$ do:

Determine a subtree $\pi(\sigma)$ in $\bigcup_{j < i} \mathcal{S}_j$, having $\pi(\sigma) \cap \sigma \neq \emptyset$;

Step 8 For all $i \in \{0, 1, 2, \dots, t\}, \tau \in \mathcal{S}_i^m$ do:

(8.1) Set $\text{Dangle}(\tau) = \{\sigma \in \mathcal{S}_{i+1}^u : \pi(\sigma) = \tau\}$;

(8.2) If the total weight of $(\tau \setminus h(\tau)) \cup \text{Dangle}(\tau)$ is at most $\zeta(1 + \varepsilon)^{i+1} M$, then set $q = 1$ and $F'_1 = (\tau \setminus h(\tau)) \cup \text{Dangle}(\tau)$, and go to Step (8.5);

(8.3) Find an Euler tour in the multigraph $((\tau \setminus h(\tau)) \cup \text{Dangle}(\tau)) + ((\tau \setminus h(\tau)) \cup \text{Dangle}(\tau))$, and transform the tour to a cycle by “short-cutting” previously visited vertices;

(8.4) Split the resulting cycle into maximal paths of total weight, including edge weights and vertex weights, at most $\zeta(1 + \varepsilon)^{i+1} M$ each, denoted by F'_1, F'_2, \dots, F'_q ;

(8.5) For each $j \in \{1, \dots, q\}$, augment F'_j by adding an edge from r to the vertex in F'_j closest to r , to obtain a set of subtrees starting at r , denoted by $\mathcal{F}_i(\tau) = \{F_1, F_2, \dots, F_q\}$;

Step 9 For each $i \in \{0, 1, \dots, t\}$, set $\mathcal{F}_i = \bigcup_{\tau \in \mathcal{S}_i^m} \mathcal{F}_i(\tau)$; Using the ASSIGN algorithm, combine the set $\{\mathcal{F}_i\}_{i \geq 0} = \bigcup_{i \geq 0} \mathcal{F}_i$ into k trees $\mathcal{T} = \{T_i \mid i = 1, 2, \dots, k\}$ corresponding to k construction teams;

Step 10 Output k trees $\mathcal{T} = \{T_i \mid i = 1, 2, \dots, k\}$ corresponding to k teams.
End

Using Step 4, we obtain that each connected component of the subgraph in G corresponding to H_i ($i \geq 1$) is a subtree. Note that such a subtree contains at least one edge from $V_{<i}$ to V_i , it follows that the partition at Step 5 is indeed executed.

Using Steps 6–7 in the algorithm HRTC₁, we can obtain the following

Lemma 5. *For any $i \geq 1$ and $\sigma \in \mathcal{S}_i^u$, there exists $\pi(\sigma) \in \mathcal{S}_{i-1}$. Moreover, $\pi(\sigma) \in \mathcal{S}_{i-1}^m$.*

Proof. For an $\sigma \in \mathcal{S}_i^u$, it is clear that $w(h(\sigma)) + f_1(h(\sigma)) \leq w(\sigma) + f_1(\sigma) < \gamma \cdot (1 + \varepsilon)^i M$. By the definition of \mathcal{S}_i^u , we have $v \in V_{\leq i}$ for every $v \in V(\sigma)$. Let $h(\sigma) = yx$, where $y \in \pi(\sigma)$ and $x \in V_i$. Moreover, we deduce that $y \in V_{i-1}$. Otherwise, we assume $x \in V_i$ and $y \in V_{<i-1}$, it follows that $w(\sigma) + f_1(\sigma) \geq w(yx) + f(x) \geq w(rx) + f(x) - (w(ry) + f(y)) > (1 + \varepsilon)^{i-1} M - (1 + \varepsilon)^{i-2} M > \gamma \cdot (1 + \varepsilon)^i M$, which contradicts $\sigma \in \mathcal{S}_i^u$. Thus, we obtain $y \in V_{i-1}$, implying $\pi(\sigma) \in \mathcal{S}_{i-1}$.

For the second part of the lemma, if $i = 1$, then clearly $\pi(\sigma) = \mathcal{S}_0$ and $\pi(\sigma) \in \mathcal{S}_0^m$. When $i \geq 2$, from the above arguments, we have $\pi(\sigma) \in \mathcal{S}_{i-1}$. Similar to the above arguments, since G is connected, we conclude that there is a vertex $z \in \pi(\sigma)$ satisfying $z \in V_{<i-1}$. Using the triangle inequality twice, we have the following

$$w(zy) + f(y) + w(yx) + f(x) \geq w(zx) + f(x) \geq w(rx) + f(x) - (w(rz) + f(z)).$$

Since $x \in V_i$ and $z \in V_{<i-1}$, we obtain $w(rx) + f(x) - (w(rz) + f(z)) > (1 + \varepsilon)^{i-1} M - (1 + \varepsilon)^{i-2} M = \varepsilon(1 + \varepsilon)^{i-2} M$, meaning $w(rx) + f(x) - (w(rz) + f(z)) > \varepsilon(1 + \varepsilon)^{i-2} M$. Since $\sigma \in \mathcal{S}_i^u$, we have $w(yx) + f(x) \leq w(\sigma) + f_1(\sigma) < \gamma \cdot (1 + \varepsilon)^i M$. Hence, we have the following

$$w(\pi(\sigma)) + f_1(\pi(\sigma)) \geq w(zy) + f(y) > \varepsilon(1 + \varepsilon)^{i-2} M - \gamma \cdot (1 + \varepsilon)^i M = \gamma \cdot (1 + \varepsilon)^{i-1} M.$$

This shows that the subtree $\pi(\sigma) \in \mathcal{S}_{i-1}^m$. □

Employing the similar argument as in [8], we obtain the following two lemmas by executing Step 8.

Lemma 6. *For any $F \in \mathcal{F}_i(\tau)$, we have $w(F) + f(F) \leq (\zeta + 1 + (\zeta + 1)\varepsilon)(1 + \varepsilon)^i M$.*

Proof. For each $F \in \mathcal{F}_i(\tau)$, note that F consists of some subtree F'_j ($1 \leq j \leq q$) and an edge rv'_j connecting r to v'_j , where v'_j is a vertex in F'_j closest to r . Based on the construction of F'_j , we obtain $w(F'_j) + f(F'_j) \leq \zeta(1 + \varepsilon)^{i+1} M$. Since F'_j only contains vertices in $V_{\leq i+1}$, we have $w(rv'_j) \leq w(rv'_j) + f(v'_j) \leq (1 + \varepsilon)^{i+1} M$. Hence, it follows that $w(F) + f(F) = w(F'_j) + f(F'_j) + w(rv'_j) \leq (\zeta + 1 + (\zeta + 1)\varepsilon)(1 + \varepsilon)^i M$. □

Lemma 7. $\sum_{F \in \mathcal{F}_i(\tau)} (w(F) + f(F)) \leq \max\{2 + \frac{2}{\varepsilon}, \frac{4}{\zeta} + 2\} \cdot (w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau)))$.

Proof. We break the analysis into two cases depending on q .

Case 1: $q = 1$, i.e., $\mathcal{F}_i(\tau)$ only contains a subtree F .

If $i = 0$, then τ includes r and $w(F) + f(F) \leq w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau))$. If $i > 0$, it is clear that there exists $u \in \tau$ having $u \in V_{<i}$. Based on the construction of subtree, we obtain $w(F) + f(F) \leq w(rx_{h(\tau)}) + w(F') + f_1(\tau \cup \text{Dangle}(\tau)) \leq w(ry_{h(\tau)}) + w(y_{h(\tau)}x_{h(\tau)}) + w(F') + f_1(\tau \cup \text{Dangle}(\tau)) \leq (1 + \varepsilon)^{i-1}M + w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau))$. Since $\tau \in \mathcal{S}_i^m$, implying $w(\tau) + f_1(\tau) \geq \gamma \cdot (1 + \varepsilon)^i M$, we have $(1 + \varepsilon)^{i-1}M \leq \frac{2+\varepsilon}{\varepsilon} \cdot (w(\tau) + f_1(\tau))$. Thus, we obtain $w(F) + f(F) \leq (1 + \varepsilon)^{i-1}M + w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau)) \leq (\frac{2+\varepsilon}{\varepsilon} + 1) \cdot (w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau))) = (2 + \frac{2}{\varepsilon}) \cdot (w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau)))$, which implies $\sum_{F \in \mathcal{F}_i(\tau)} (w(F) + f(F)) = w(F) + f(F) \leq (2 + \frac{2}{\varepsilon}) \cdot (w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau)))$.

Case 2: $q \geq 2$.

By Step 8, we obtain that $4(w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau))) > (q-1) \cdot \zeta(1+\varepsilon)^{i+1}M$, that is $(1+\varepsilon)^{i+1}M < \frac{4(w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau)))}{\zeta(q-1)}$. Since $V(\tau \cup \text{Dangle}(\tau)) \subseteq V_{\leq i+1}$, each edge added from r to subtree F'_j ($1 \leq j \leq q$) has weight at most $(1 + \varepsilon)^{i+1}M$. Therefore, we conclude that $\sum_{F \in \mathcal{F}_i(\tau)} (w(F) + f(F)) \leq q \cdot (1 + \varepsilon)^{i+1}M + 2w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau)) \leq (\frac{4q}{\zeta(q-1)} + 2) \cdot (w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau))) \leq (\frac{4}{\zeta} + 2) \cdot (w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau)))$.

Combining the two preceding arguments in Cases 1–2, we obtain $\sum_{F \in \mathcal{F}_i(\tau)} (w(F) + f(F)) \leq \max\{2 + \frac{2}{\varepsilon}, \frac{4}{\zeta} + 2\} \cdot (w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau)))$. \square

Applying Lemmas 5–7, we obtain the following

Lemma 8. *If $w(T_{G/V_{<i}}^{MS}) + f(V_{\geq i}) \leq M \cdot \Lambda((1 + \varepsilon)^{i-1})$ holds for each integer $i \geq 0$, then the collection $\{\mathcal{F}_i\}_{i \geq 0}$ obtained at Step 8 is $(\alpha, \beta)_{M, \varepsilon}$ -assignable, where $\alpha = \zeta + 1 + (\zeta + 1)\varepsilon$, $\beta = (6 + \frac{6}{\varepsilon}) \max\{2 + \frac{2}{\varepsilon}, \frac{4}{\zeta} + 2\}$ and $\mathcal{F}_i = \bigcup_{\tau \in \mathcal{S}_i^m} \mathcal{F}_i(\tau)$.*

Proof. We shall prove that the collection $\{\mathcal{F}_i\}_{i \geq 0}$ satisfies the two properties in Definition 1. By Lemma 6, it is clear that the property (1) in Definition 1 holds. Recall that in Lemma 4, $\sum_{j \geq i} (w(H_j) + f(V_j)) \leq (6 + \frac{6}{\varepsilon}) \cdot M \cdot \Lambda((1 + \varepsilon)^{i-1})$ holds for each $i \geq 0$. Now, the proof is completed by showing that $\sum_{j \geq i} (w(\mathcal{F}_j) + f(\mathcal{F}_j)) \leq \max\{2 + \frac{2}{\varepsilon}, \frac{4}{\zeta} + 2\} \cdot \sum_{j \geq i} (w(H_j) + f(V_j))$.

In the algorithm HRTC_1 , we observe that

$$\sum_{j \geq i} (w(\mathcal{S}_j^m) + w(\mathcal{S}_{j+1}^u)) \leq \sum_{j \geq i} w(\mathcal{S}_j) = \sum_{j \geq i} w(H_j).$$

By Lemma 5, $\{\text{Dangle}(\tau) \mid \tau \in \mathcal{S}_j^m\}$ is a partition of \mathcal{S}_{j+1}^u , which means $\sum_{\tau \in \mathcal{S}_j^m} (w(\tau \cup \text{Dangle}(\tau)) + f_1(\tau \cup \text{Dangle}(\tau))) = w(\mathcal{S}_j^m) + w(\mathcal{S}_{j+1}^u) + f_1(\mathcal{S}_j^m \cup \mathcal{S}_{j+1}^u)$. By Lemma 7, we have $w(\mathcal{F}_j) + f(\mathcal{F}_j) = \sum_{\tau \in \mathcal{S}_j^m} (w(\mathcal{F}_j(\tau)) + f(\mathcal{F}_j(\tau))) = \sum_{\tau \in \mathcal{S}_j^m} \sum_{F \in \mathcal{F}_j(\tau)} (w(F) + f(F)) \leq \max\{2 + \frac{2}{\varepsilon}, \frac{4}{\zeta} + 2\} \cdot \sum_{\tau \in \mathcal{S}_j^m} (w(\tau \cup \text{Dangle}(\tau)) +$

$f_1(\tau \cup \text{Dangle}(\tau))$), implying $w(\mathcal{F}_j) + f(\mathcal{F}_j) \leq \max\{2 + \frac{2}{\varepsilon}, \frac{4}{\zeta} + 2\} \cdot (w(\mathcal{S}_j^m) + w(\mathcal{S}_{j+1}^u) + f_1(\mathcal{S}_j^m \cup \mathcal{S}_{j+1}^u))$. By Steps 6–8, for each edge $e \in \mathcal{S}_i^m$ ($i \geq 0$), we see that $x_e \in V_{\geq i}$, and any two subtrees in $\{\mathcal{F}_i\}_{i \geq 0}$ are disjoint except for root r . Thus, we obtain $\sum_{j \geq i} (w(\mathcal{F}_j) + f(\mathcal{F}_j)) \leq \max\{2 + \frac{2}{\varepsilon}, \frac{4}{\zeta} + 2\} \cdot \sum_{j \geq i} (w(\mathcal{S}_j^m) + w(\mathcal{S}_{j+1}^u) + f_1(\mathcal{S}_j^m \cup \mathcal{S}_{j+1}^u)) \leq \max\{2 + \frac{2}{\varepsilon}, \frac{4}{\zeta} + 2\} \cdot \sum_{j \geq i} (w(H_j) + f(V_j))$. \square

Using the above lemmas, we obtain the following result.

Theorem 1. *The algorithm HRTC_1 is a $58.3286(1 + \delta)$ -approximation algorithm to solve the HRTC problem, and it runs in time $O(n^3(1 + \frac{1}{\delta}) + \log(w(E) + f(V \setminus \{r\})))$, where $w(E) = \sum_{e \in E} w(e)$ and $f(V \setminus \{r\}) = \sum_{v \in V \setminus \{r\}} f(v)$, respectively.*

Proof. By Lemma 2, the decision condition in Step 3 does not hold whenever $M \geq \text{OPT}$. Based on the update rule for M , we deduce that Steps 4–9 is executed with $M \leq (1 + \delta) \cdot \text{OPT}$. When fixing $\zeta = 2$ and $\varepsilon = 1.3146$, using Lemma 8, we obtain a $(6.9438, 42.2565)_{M, 1.3146}$ -assignable collection $\{\mathcal{F}_i\}_{i \geq 0}$. Using Lemma 1 at Step 9, we can assign $\{\mathcal{F}_i\}_{i \geq 0}$ into k construction teams in time $O(n^3 + \log(w(E) + f(V \setminus \{r\})))$, such that the completion time of any construction team is at most $58.3286 \cdot M$, which implies $\text{OUT} \leq 58.3286 \cdot M \leq 58.3286(1 + \delta) \cdot \text{OPT}$.

Notice that every step in the algorithm HRTC_1 can be executed in polynomial time. We shall bound the number of iterations. As mentioned above, the algorithm HRTC_1 halts before $M > (1 + \delta)\text{OPT}$, where $(1 + \delta)\text{OPT} \leq (1 + \delta) \cdot \frac{\sum_{v \in V} (w(rv) + f(v))}{\lambda_{\max}} \leq (1 + \delta) \cdot |V| \cdot \frac{\max_{v \in V} \{w(rv) + f(v)\}}{\lambda_{\max}}$. Since M is initialized at $\frac{\max_{v \in V} \{w(r, v) + f(v)\}}{\lambda_{\max}}$, and increased by an $(1 + \delta)$ -factor for each iteration, we deduce that the number of iterations is at most $O(\frac{1}{\delta} \log |V|)$. This implies that Steps 1–3 run in at most time $O(\frac{n^3}{\delta})$. By Lemma 1, it is easy to check that Steps 4–10 execute in time $O(n^3 + \log(w(E) + f(V \setminus \{r\})))$. Hence, the algorithm HRTC_1 can be implemented in time $O(n^3(1 + \frac{1}{\delta}) + \log(w(E) + f(V \setminus \{r\})))$. \square

4 An Approximation Algorithm with Lower Time Complexity

In practice, we observe a fact that $\frac{\lambda_{\max}}{\lambda_{\min}}$ is generally small, where $\lambda_{\max} = \max\{\lambda_i \mid i = 1, 2, \dots, k\}$ and $\lambda_{\min} = \min\{\lambda_i \mid i = 1, 2, \dots, k\}$. Thus, we intend to design a better approximation algorithm to resolve the HRTC problem under the above fact.

Different from the method in [10] for solving the RTC problem, we modify a splitting technique in [6] to design an approximation algorithm to resolve the HRTC problem, which is described as follows.

Algorithm: HRTC_2

INPUT: An undirected complete graph $G = (V, E; w, f)$ with a root $r \in V$, an edge-weight function $w : E \rightarrow R^+$, a vertex-weight function $f : V \setminus \{r\} \rightarrow R_0^+$ and k construction teams having speeds $\lambda_1, \dots, \lambda_k$, respectively;

OUTPUT: A set $\mathcal{T} = \{T_i \mid i = 1, 2, \dots, k\}$ of trees.

Begin

Step 1 Find a minimum edge-weight spanning tree T^{MS} in G ; Determine an Euler tour in $(V, E_{T^{MS}} + E_{T^{MS}})$ traversing each edge exactly once, and use “short-cutting” to transform such tour to a cycle $C = rv_{i_1}v_{i_2} \cdots r$ traversing each vertex $v \in V$ exactly once;

Step 2 Set $w_0 = \max\{w(rv) \mid v \in V\}$; For each edge $uv \in E$, set $w'(uv) = w(uv) + f(u) + f(v)$; For each $i \in \{1, 2, \dots, k\}$, set $\lambda_{1,i} = \sum_{j=1}^i \lambda_j$;

Step 3 For $j = 1$ to $k - 1$ do:

Set $L_j = \frac{\lambda_{1,j}}{\lambda_{1,k}}(w'(C) - 2w_0) + w_0$, find the last vertex $v_{i_{p(j)}}$ such that $w'(C[r, v_{i_{p(j)}}]) \leq L_j$, where $C[r, v_{i_{p(j)}}] = rv_{i_1}v_{i_2} \cdots v_{i_{p(j)}}$;

Step 4 Set $T'_1 = C[r, v_{i_{p(1)}}] = rv_{i_1} \cdots v_{i_{p(1)}}$, $T'_2 = C[v_{i_{p(1)+1}}, v_{i_{p(2)}}], \dots, T'_k = C[v_{i_{p(k-1)+1}}, r]$; For each $j \in \{1, 2, \dots, k\}$, augment T'_j by connecting r to a vertex in T'_j closest to r with edge, where the resulting tree is denoted by T_j ;

Step 5 Output k trees $\mathcal{T} = \{T_i \mid i = 1, 2, \dots, k\}$ corresponding to k teams.

End

Analyzing the lower bound of the optimal value for the HRTC problem, we obtain the following result.

Lemma 9. *Given a complete graph $G = (V, E; w, f)$ as an instance of the HRTC problem, we have $OPT \geq \max\{\frac{f_0}{\lambda_{\max}}, \frac{w_0}{\lambda_{\max}}, \frac{w'(C)}{2\lambda_{1,k}}\}$, where OPT is the optimal value to the given instance, $f_0 = \max\{f(v) \mid v \in V\}$, $w_0 = \max\{w(rv) \mid v \in V\}$ and C is produced at Step 1 of the algorithm HRTC₂.*

Proof. Note that any feasible solution for the instance G cover all vertices in V , it is clear that $OPT \geq \frac{f_0}{\lambda_{\max}}$ and $OPT \geq \frac{w_0}{\lambda_{\max}}$. We shall prove $OPT \geq \frac{w'(C)}{2\lambda_{1,k}}$.

Consider any optimal solution $\mathcal{T}^* = \{T_i^* \mid i = 1, 2, \dots, k\}$ for the instance G . By the construction of C at Step 1, we have $w(T^{MS}) + f(T^{MS}) \geq \frac{w(C)}{2} + f(C)$, where T^{MS} is a minimum edge-weight spanning tree of G . Since all subtrees in \mathcal{T}^* can be merged into a spanning tree, we obtain $\sum_{i=1}^k (w(T_i^*) + f(T_i^*)) \geq w(T^{MS}) + f(T^{MS})$, implying $\sum_{i=1}^k (w(T_i^*) + f(T_i^*)) \geq \frac{w(C)}{2} + f(C)$. Since $OPT = \max\{\frac{w(T_i^*) + f(T_i^*)}{\lambda_i} \mid i = 1, 2, \dots, k\}$, it follows that $OPT \cdot \sum_{i=1}^k \lambda_i \geq \sum_{i=1}^k (w(T_i^*) + f(T_i^*)) \geq \frac{w(C)}{2} + f(C)$, meaning $OPT \geq \frac{w(C) + 2f(C)}{2 \sum_{i=1}^k \lambda_i} = \frac{w(C) + 2f(C)}{2\lambda_{1,k}}$. This implies $OPT \geq \frac{w(C) + 2f(C)}{2\lambda_{1,k}} = \frac{w'(C)}{2\lambda_{1,k}}$. \square

By the algorithm HRTC₂, we obtain the following result.

Theorem 2. *The algorithm HRTC₂ is a $\max\{\frac{2\lambda_{\max}}{\lambda_{\min}}, 2 + \frac{\lambda_{\max}}{\lambda_{\min}} - \frac{2}{k}\}$ -approximation algorithm for resolving the HRTC problem, and it runs in time $O(n^2)$, where n is the number of vertices.*

Proof. Given an instance $G = (V, E; w, f)$ of the HRTC problem, we may assume that $\mathcal{T}^* = \{T_i^* \mid i = 1, 2, \dots, k\}$ is an optimal solution with the optimal value $OPT = \max\{\frac{w(T_i^*) + f(T_i^*)}{\lambda_i} \mid i = 1, \dots, k\}$, and \mathcal{T} is trees outputted by the algorithm HRTC₂ with the output value $OUT = \max\{\frac{w(T_i) + f(T_i)}{\lambda_i} \mid i = 1, \dots, k\}$.

Now, we consider the j^{th} tree T_j ($1 \leq j \leq k$) in \mathcal{T} . Using the algorithm HRTC_2 , we obtain the following

$$\begin{aligned} \frac{w(T_j) + f(T_j)}{\lambda_j} &\leq \frac{w(T'_j) + f(T'_j) + w_0}{\lambda_j} \\ &\leq \max\left\{\frac{f_0}{\lambda_j}, \frac{w'(T'_j)}{\lambda_j}\right\} + \frac{w_0}{\lambda_j} \\ &= \max\left\{\frac{f_0 + w_0}{\lambda_j}, \frac{w'(T'_j) + w_0}{\lambda_j}\right\}. \end{aligned}$$

From Lemma 9, it may be concluded that $\frac{f_0 + w_0}{\lambda_j} = \frac{f_0}{\lambda_j} + \frac{w_0}{\lambda_j} \leq \frac{f_0}{\lambda_{\min}} + \frac{w_0}{\lambda_{\min}} \leq \frac{2\lambda_{\max}}{\lambda_{\min}} \text{OPT}$. On account of the construction of \mathcal{T} in algorithm, we have the following

$$\begin{aligned} \frac{w'(T'_j) + w_0}{\lambda_j} &\leq \frac{\frac{\lambda_j}{\lambda_{1,k}} \cdot (w'(C) - 2w_0) + w_0}{\lambda_j} \\ &= \frac{w_0}{\lambda_j} + \frac{w'(C) - 2w_0}{\lambda_{1,k}} \\ &\leq \frac{w_0}{\lambda_{\min}} + \frac{w'(C)}{\lambda_{1,k}} - \frac{2w_0}{\lambda_{1,k}} \\ &= \frac{\lambda_{\max} \cdot w_0}{\lambda_{\min} \cdot \lambda_{\max}} + \frac{w'(C)}{\lambda_{1,k}} - \frac{2w_0}{\lambda_{1,k}} \\ &\leq \frac{\lambda_{\max} \cdot w_0}{\lambda_{\min} \cdot \lambda_{\max}} + \frac{w'(C)}{\lambda_{1,k}} - \frac{2w_0}{k \cdot \lambda_{\max}} \\ &= \frac{w'(C)}{\lambda_{1,k}} + \left(\frac{\lambda_{\max}}{\lambda_{\min}} - \frac{2}{k}\right) \cdot \frac{w_0}{\lambda_{\max}} \\ &\leq 2\text{OPT} + \left(\frac{\lambda_{\max}}{\lambda_{\min}} - \frac{2}{k}\right) \cdot \text{OPT} \\ &= \left(2 + \frac{\lambda_{\max}}{\lambda_{\min}} - \frac{2}{k}\right) \cdot \text{OPT}, \end{aligned}$$

implying $\frac{w(T_j) + f(T_j)}{\lambda_j} \leq \max\left\{\frac{2\lambda_{\max}}{\lambda_{\min}}, 2 + \frac{\lambda_{\max}}{\lambda_{\min}} - \frac{2}{k}\right\} \cdot \text{OPT}$.

Thus, for each $j \in \{1, \dots, k\}$, we have $\frac{w(T_j) + f(T_j)}{\lambda_j} \leq \max\left\{\frac{2\lambda_{\max}}{\lambda_{\min}}, 2 + \frac{\lambda_{\max}}{\lambda_{\min}} - \frac{2}{k}\right\} \cdot \text{OPT}$ by using the above arguments. This shows that

$$\text{OUT} \leq \max\left\{\frac{2\lambda_{\max}}{\lambda_{\min}}, 2 + \frac{\lambda_{\max}}{\lambda_{\min}} - \frac{2}{k}\right\} \cdot \text{OPT}.$$

The time complexity of the algorithm HRTC_2 can be determined as follows.

(1) Using Prim algorithm for solving the minimum spanning tree problem, Step 1 execute in time $O(n^2)$; (2) Step 2 needs $O(m)$ time to compute w_0 and define $w'(\cdot)$, where $m = |E|$; (3) Step 3 needs time $O(n^2)$ to split a cycle; (4) Step 4 needs time $O(m)$ to construct the trees $\mathcal{T} = \{T_i \mid i = 1, 2, \dots, k\}$. Therefore, the running time of the algorithm HRTC_2 is $O(n^2)$. \square

5 Conclusion and Further Work

In this paper, we consider the heterogeneous rooted tree cover problem (the HRTC problem), and design two approximation algorithms for solving the HRTC problem.

In further work, it is a challenge for us to design some approximation algorithms with constant approximation ratios to solve the HRTC problem in strongly polynomial time, and we shall study other versions of the cover problems with nonuniform speeds.

References

1. Averbakh, I., Berman, O.: A heuristic with worst-case analysis for minimax routing of two travelling salesmen on a tree. *Discret. Appl. Math.* **68**(1–2), 17–32 (1996)
2. Campbell, A.M., Vandenbussche, D., Hermann, W.: Routing for relief efforts. *Transp. Sci.* **42**(2), 127–145 (2008)
3. Christofides, N.: Worst-case analysis of a new heuristic for the travelling salesman problem, Report 388. Carnegie Mellon University, Graduate School of Industrial Administration (1976)
4. Even, G., Garg, N., Koemann, J., Ravi, R., Sinha, A.: Min-max tree covers of graphs. *Oper. Res. Lett.* **32**(4), 309–315 (2004)
5. Farbstain, B., Levin, A.: Min-max cover of a graph with a small number of parts. *Discret. Optim.* **16**, 51–61 (2015)
6. Frederickson, G.N., Hecht, M.S., Kim, C.E.: Approximation algorithms for some routing problems. *SIAM J. Comput.* **7**(2), 178–193 (1978)
7. Golden, B.L., Raghavan, S., Wasil, E.A.: *The Vehicle Routing Problem: Latest Advances and New Challenges*. Springer, New York (2008). <https://doi.org/10.1007/978-0-387-77778-8>
8. Gørtz, I.L., Molinaro, M., Nagarajan, V., Ravi, R.: Capacitated vehicle routing with nonuniform speeds. *Math. Oper. Res.* **41**(1), 318–331 (2016)
9. Nagamochi, H., Okada, K.: Polynomial time 2-approximation algorithms for the minmax subtree cover problem. In: Ibaraki, T., Katoh, N., Ono, H. (eds.) *ISAAC 2003*. LNCS, vol. 2906, pp. 138–147. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-24587-2_16
10. Nagamochi, H.: Approximating the minmax rooted-subtree cover problem. *IEICE Trans. Fundamentals Electron. Commun. Comput. Sci.* **E88-A**(5), 1335–1338 (2005)
11. Schwartz, S.: An overview of graph covering and partitioning. *Discret. Math.* **345**(8), 112884 (2022)
12. Toth, P., Vigo, D.: *Vehicle Routing: Problems, Methods and Applications*. MOS-SIAM, Philadelphia (2014)
13. Wu, W., Zhang, Z., Lee, W., Du, D.Z.: *Optimal Coverage in Wireless Sensor Networks*. Springer, Heidelberg (2020). <https://doi.org/10.1007/978-3-030-52824-9>
14. Xu, Z., Wen, Q.: Approximation hardness of min-max tree covers. *Oper. Res. Lett.* **38**(3), 169–173 (2010)
15. Xu, W., Liang, W., Lin, X.: Approximation algorithms for min-max cycle cover problems. *IEEE Trans. Comput.* **64**(3), 600–613 (2015)
16. Yu, W., Liu, Z.: Better approximability results for min-max tree/cycle/path cover problems. *J. Comb. Optim.* **37**, 563–578 (2019)