Hardness and Approximation for the Star p-Hub Routing Cost Problem in Metric Graphs

Hao-Ping Yeh, Wei Lu, Li-Hsuan Chen, Ling-Ju Hung, Ralf Klasing, Sun-Yuan Hsieh

Hardness and Approximation for the Star p-Hub Routing Cost Problem in Metric Graphs[⋆]

Hao-Ping Yeh[a], Wei Lu[a], Li-Hsuan Chen[b], Ling-Ju Hung[c,∗], Ralf Klasing[d], Sun-Yuan Hsieh[a,e]

[a]*Department of Computer Science and Information Engineering, National Cheng Kung University, No. 1, University Road, Tainan 701, Taiwan*

[b]*ProBrand Technology, Inc., Taoyuan City 330, Taiwan*

[c]*Department of Creative Technologies and Product Design, National Taipei University of Business, No. 100, Sec. 1, Fulong Rd., Pingzhen District, Taoyuan City 32462, Taiwan*

[d]*CNRS, LaBRI, Université de Bordeaux, 351 Cours de la Libération, 33405 Talence cedex, France*

[e]*Institute of Medical Information, Institute of Manufacturing Information and Systems, Center for Innovative FinTech Business Models, and International Center for the Scientific Development of Shrimp Aquaculture, National Cheng Kung University, No. 1, University Road, Tainan 701, Taiwan*

Abstract

Given a metric graph $G = (V, E, w)$, a specific vertex $c \in V$, and an integer p, let T be a depth-2 spanning tree of G rooted at c such that c is adjacent to p vertices called hubs and each of the remaining vertices is adjacent to a hub. The Star p-Hub Routing Cost Problem is to find a spanning tree T of G satisfying the conditions stated above such that the sum of distances between all pairs of vertices in T is minimized. In this paper, we prove that the Star p-Hub Routing Cost Problem is NP-hard. A 3-approximation algorithm running in time $O(n^2)$ is given for solving the same problem where n is the number of vertices in the input graph. Moreover, we give an example to show that the analysis of the approximation ratio cannot be better than $2 - \epsilon$ for any $\epsilon > 0$.

[⋆]The preliminaries of this paper were published in *Proceedings of the 23rd International Computer Symposium (ICS 2018)*, vol. 1013 of Communications in Computer and Information Science, pp. 534-531.

[∗]Corresponding author

Email addresses: `P76081459@mail.ncku.edu.tw` (Hao-Ping Yeh), `p76041166@mail.ncku.edu.tw` (Wei Lu), `g943339@oz.nthu.edu.tw` (Li-Hsuan Chen), `ljhung@ntub.edu.tw` (Ling-Ju Hung), `ralf.klasing@labri.fr` (Ralf Klasing), `hsiehsy@mail.ncku.edu.tw` (Sun-Yuan Hsieh)

1. Introduction

In computer or transportation networks, each node has to send messages or transport commodities to other nodes. Instead of using the direct link between each origin/destination pair, a network with hub-and-spoke architecture uses hubs to connect each origin/destination pair. Hubs are facilities that can provide services such as consolidation and switching of flows between origins and destinations. All hubs are usually assumed to be fully interconnected. Each path from an origin to a destination in a hub-and-spoke network goes through at least one hub. With large amounts of flows passing through hubs, hubs can achieve economies of scale in network utilization. Furthermore, the total number of links can be reduced in a hub-and-spoke network. Hub-and-spoke networks have numerous applications in the airport industry [1], transportation systems [2], and logistics networks [3].

1.1. Hub Location Problems

The hub location problem (HLP) is to design a hub-and-spoke network fulfilling certain constraints and optimizing a predefined objective. HLPs usually consist of a two-level decision process: selecting some nodes to locate hubs and allocating the remaining nodes (called non-hubs) to hubs. The p-Hub Median Problem, a classic HLP, is to locate p hubs in order to minimize the total transportation cost. In 1987, O'Kelly [4] developed the first mathematical formulations of this problem by studying the dataset of airline passenger networks. Since then, numerous HLPs and variants of the p-Hub Median Problem have been studied by researchers [5–15].

Single-allocation and *multi-allocation* are two types of HLPs that differ in how non-hubs are allocated to hubs. If every non-hub can be served by exactly one hub, the problem is classified as single-allocation. If every non-hub can be served by several hubs, the problem is classified as multi-allocation. For more classifications of HLPs, there exist several reviews and surveys on HLPs [5, 7, 12, 14].

1.2. Approximation Results for HLPs

Although there are many results on linear programming-based and heuristic algorithms for solving HLPs, the studies on approximation algorithms of HLPs are still few. We introduce some approximation results for HLPs in this subsection.

Iwasa *et al.* [16] studied the single allocation problem in hub-and-spoke networks. Given the predetermined locations of hubs and the required amount of flow between each pair of nodes, the problem aims to minimize the total transportation cost after allocating non-hubs to hubs. They presented a deterministic 3-approximation algorithm and a randomized 2-approximation algorithm. Chen *et al.* [13] studied the SINGLE ALLOCATION AT MOST p-HUB CENTER ROUTING PROBLEM in Δ_β-metric graphs. Δ_β-metric graphs are weighted complete graphs satisfying the β-triangle inequality. The problem involves locating at most p hubs and allocating each non-hub to a hub in order to minimize the routing cost. For any $\beta > 1/2$, they proved that the problem is NP-hard and gave 2β-approximation algorithms.

The p-HUB CENTER PROBLEM is to locate p hubs and allocate each non-hub to a hub so as to minimize the maximum cost between pairs of nodes. Chen *et al.* [8] studied this problem in metric graphs. They showed that for any $\epsilon > 0$, approximating the p-HUB CENTER PROBLEM to a ratio $4/3 - \epsilon$ is NP-hard. They also proposed a 5/3-approximation algorithm. As for Δ_β-metric graphs, the upper and lower bounds of approximability according to the value of β were provided in [10]. Wang *et al.* [15] studied six variants of the p-HUB MEDIAN PROBLEM and the p-HUB CENTER PROBLEM. Each of the variants has a different objective function. They presented improved hardness and approximation results.

The STAR p-HUB CENTER PROBLEM is similar to the p-HUB CENTER PROBLEM. The difference between them is that hubs in the STAR p-HUB CENTER PROBLEM are connected to a given central node instead of being fully interconnected. It was shown that to approximate the STAR p-HUB CENTER PROBLEM in metric graphs to a ratio $5/4 - \epsilon$ is NP-hard [6]. A 7/2-approximation algorithm was given in the same paper [6]. Chen *et al.* [9] reduced the gap between the upper and lower bounds of approximability for the STAR p-HUB CENTER PROBLEM in metric graphs. In [11], the upper and lower bounds of approximability according to the value of β was presented for the STAR p-HUB CENTER PROBLEM in Δ_β-metric graphs.

1.3. Star p-Hub Routing Cost Problem

We discuss the STAR p-HUB ROUTING COST PROBLEM (SpHRP) in this paper. This problem is to select p nodes as hubs connecting to a given central node and assign each non-hub to a hub such that the routing cost is minimized. We prove that the SpHRP problem is NP-hard and give a 3-approximation algorithm to solve it.

Given an undirected, complete, and weighted graph $G = (V, E, w)$, G is called a *metric graph* if the following three conditions are satisfied, *i.e.,* $w(v, v) = 0$ for any $v \in V$; $w(u, v) = w(v, u)$ for any $u, v \in V$; and for any $u, v, x \in V$, $w(u, v) \leq w(u, x) + w(x, v)$. Given a metric graph $G = (V, E, w)$, a specific vertex $c \in V$, and a positive integer p, the goal of SpHRP is to obtain a spanning tree T of G satisfying the following conditions: T is rooted at c, c is adjacent to exactly p vertices called hubs, and each of the remaining vertices (called non-hubs) is adjacent to exactly one hub. For $u, v \in V$, $d_H(u, v)$ denotes the distance between u, v in graph H. Notice that $d_G(u, v) = w(u, v)$ if G is a metric graph. For $u, v \in V$, $d_T(u, v)$ denotes the length of the path between u, v in tree T. Define $C(T) = \sum_{u \in V} \sum_{v \in V} d_T(u, v)$, which can be designated as the routing cost of tree T. We list the formal definition of the SpHRP in the following.

Star p-Hub Routing Cost Problem (SpHRP)

Input: A metric graph $G = (V, E, w)$, a vertex $c \in V$, and a positive integer p, where $|V| \geq 2p + 1$

Output: A depth-2 spanning tree T^* rooted at c such that c has exactly p children and the routing cost of T^*, $C(T^*)$, is minimized

Here, we assume that the number of non-hubs is greater than or equal to the number of hubs, that is, $|V| \geq 2p + 1$. The assumption $|V| \geq 2p + 1$ is reasonable because in real-world applications, a hub could be a logistics center or an airport, and a non-hub could be a retail store, a customer, or a passenger. The approximation algorithm of the SpHRP can be used to solve some variants of SpHRP. There is a variant of the problem that is to locate at most p hubs instead of locating exactly p hubs. The algorithm of SpHRP can be implemented by enumerating the values of p from 1 to p. One may consider another variant of the problem that is to choose the root and hubs at the same time instead of specifying the root in advance. The algorithm can be implemented by enumerating each vertex as root.

The solution of SpHRP is a star/star network, as the network connecting the central node to hubs is a star and each network connecting a hub and non-hubs allocated to it is also a star. The solution of SpHRP is also a depth-2 tree with a specific root connecting to exactly p hubs. Note that the output of the Star p-Hub Center Problem has the same restricted structure. There are other HLPs requiring this kind of topology as well. Yaman [17] studied a problem similar to the SpHRP where the objective is to minimize the total cost of installing capacitated links. The author presented two formulations and a heuristic algorithm. Yaman and Elloumi [18] provided several mathematical models for the Star p-Hub Median Problem with Bounded Path Lengths. The problem is close to SpHRP, but it considers the upper bound on the path lengths. Tikani *et al.* [19] studied the integrated problem considering seat inventory control and revenue management. They proposed a hybrid optimization method based on genetic algorithm and exact solution.

Problems that minimize the sum of shortest paths between all pairs of vertices have been studied for decades in different contexts. There are some problems on finding a subgraph or a spanning tree fulfilling certain constraints such that the routing cost is minimized [20–23]. An application to computational biology is presented in [23]. Some important results on polynomial time approximation schemes for these problems are in [23, 24]. Compared to these problems, the output of SpHRP has a fixed architecture. It is easier to centralize the services of the network and expand the whole network by connecting the central node to the one in another network.

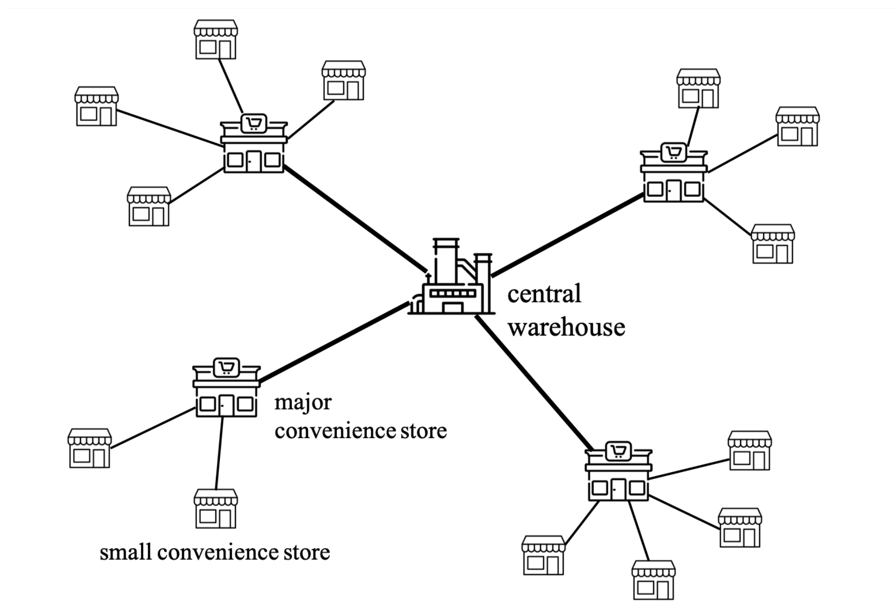

Figure 1: An example of SpHRP with $p = 4$, where the central warehouse is the given central node, four major convenience stores are hubs, and the other small convenience stores are non-hubs.

Fig. 1 displays an example of the SpHRP that can be applied to the transportation network in convenience stores. Commodities are transported from the central warehouse to convenience stores. In addition, convenience stores provide a service that customers can send packages from a convenience store to another. Both the objective and the structure of SpHRP are applicable to this scenario. The remainder of this paper is organized as follows. In Section 2, we prove that SpHRP is NP-hard. In Section 3, we propose a 3-approximation algorithm for the SpHRP. In Section 4, we present an example for the 3-approximation algorithm showing that the approximation ratio is at least $2 - \epsilon$ for any $\epsilon > 0$. We conclude the paper in Section 5.

2. NP-Hardness

In this section, we show the NP-hardness of SpHRP. We define the Restricted Exact Cover by 3-Sets Problem (Rest-X3C) as a restricted version of the well-known NP-hard problem Exact Cover by 3-Sets Problem (X3C) [25].

Exact Cover by 3-Sets Problem (X3C)
Input: A set U, with $|U| = 3q$, and a collection $\mathcal{S}$ of 3-element subsets of U
Output: Is there a subset $\mathcal{S}'$ of $\mathcal{S}$ where every element of U occurs in exactly one member of $\mathcal{S}'$?

Restricted Exact Cover by 3-Sets Problem (Rest-X3C)
Input: A set U, with $|U| = 3q$, and a collection $\mathcal{S}$ of 3-element subsets of U, with $|U| \geq |\mathcal{S}|$
Output: Is there a subset $\mathcal{S}'$ of $\mathcal{S}$ where every element of U occurs in exactly one member of $\mathcal{S}'$?

We prove that Rest-X3C is also an NP-hard problem as follows.

Lemma 1. *Rest-X3C is NP-hard.*

Proof. In the following, we reduce the X3C problem to the Rest-X3C problem. Let $I = (U, \mathcal{S})$ be an input instance of X3C. U is a universal set and $\mathcal{S}$ is a collection of 3-element subsets of U. Let $u_1, u_2, ..., u_{3x}$ be $3x$ elements not in U, where $|U| + 2x \geq |\mathcal{S}|$. Then $I' = (U \cup \{u_1, u_2, ..., u_{3x}\}, \mathcal{S} \cup \{\{u_1, u_2, u_3\}, \{u_4, u_5, u_6\}, ..., \{u_{3x-2}, u_{3x-1}, u_{3x}\}\})$ is an input instance of Rest-X3C, and it is easy to check that I has a solution if and only if I' has a solution. This implies that Rest-X3C is also an NP-hard problem. This completes the proof. □

We prove that SpHRP is an NP-hard problem as follows.

Theorem 1. *SpHRP is NP-hard.*

Proof. We prove that SpHRP is at least as hard as the Rest-X3C. We reduce the Rest-X3C to the SpHRP. Let $(U, \mathcal{S})$ be an input instance of Rest-X3C. U is a universal set and $\mathcal{S}$ is a collection of 3-element subsets of U, where

$|U| = 3q$, $|\mathcal{S}| = m$, and $3q \geq m$. We construct a metric graph $G = (V \cup S \cup \{c\}, E, w)$ of the SpHRP according to $(U, \mathcal{S})$. Let c be the specified vertex. For each element $v \in U$, create a vertex $v \in V$. For each subset $s \in \mathcal{S}$, create a vertex $s \in S$. In this section, we say "s covers v" if $v \in s$. We define the costs of the edges in G as follows.

- For c
  - $w(c, s) = 1$ if $s \in S$.
  - $w(c, v) = 2$ if $v \in V$.
- For $v \in V$
  - $w(v, v') = 2$ if $v' \in V$ and $v \neq v'$.
  - $w(v, s) = 1$ if $s \in S$ and s covers v.
  - $w(v, s') = 3$ if $s' \in S$ and s' does not cover v.
- For $s \in S$
  - $w(s, s') = 2$ if $s' \in S$ and $s \neq s'$.

$w(u,v)$	c	v'	s'
c	0	2	1
v	2	2	$w(v,s')$
s	1	$w(s,v')$	2

Table 1: The costs of edges in G, where $v, v' \in V$, $s, s' \in S$, and $w(v, s')$, $w(s, v')$ are either 1 or 3.

Table 1 presents the edge costs of all edges (u, v) in G. It is not hard to see that G is a metric graph since any three vertices u, v, r in G satisfy $w(u, v) + w(v, r) \geq w(u, r)$. Let $(G, c, p = m)$ be the input of the SpHRP.

If the Rest-X3C has a solution $\mathcal{S}'$, we assume that $\mathcal{S}' = \{s_1, s_2, ..., s_q\}$ without loss of generality. We then construct a solution T (see Fig. 2) of the SpHRP according to $\mathcal{S}'$. Let all vertices in S be hubs. For each $s \in \mathcal{S}'$, we find the corresponding vertex $s \in S$ and connect s to the three vertices in V that are covered by s. Define $R_T(A, B) = \sum_{a \in A} \sum_{b \in B} d_T(a, b)$. We see that the routing cost of T is

$$\begin{aligned} C(T) &= R_T(V,V) + R_T(S,S) + 2R_T(S,V) + 2R_T(\{c\}, S \cup V) \\ &= [3q(3q-1) \cdot 4 - 3q \cdot 2(4-2)] + m(m-1) \cdot 2 + 2[m(3q) \cdot 3 \end{aligned}$$

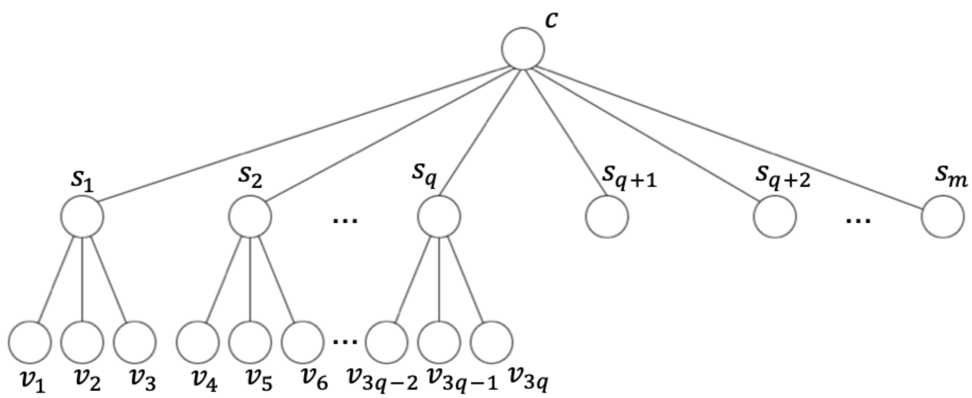

Figure 2: A solution T of the SpHRP after the reduction.

$$
\begin{aligned}
& -3q(3-1)] + 2 \cdot (m + 3q \cdot 2) \\
= \;& (36q^2 - 24q) + (2m^2 - 2m) + (18mq - 12q) + (2m + 12q) \\
= \;& 18mq + 2m^2 + 36q^2 - 24q
\end{aligned}
$$

Let T^* denote an optimal solution of SpHRP. We have $C(T^*) \leq C(T) = 18mq + 2m^2 + 36q^2 - 24q$. Next, we use this inequality to prove Claims 1 to 3.

Claim 1. *In T^*, all vertices in S are hubs and adjacent to c.*

Proof. In T^*, suppose a vertex v_x in V is a hub, and a vertex s_x in S is a non-hub. For a non-hub u in T^*, we use $f^*(u)$ to denote the hub adjacent to u in T^*.

- For $R_{T^*}(V, V)$:

  Only case (a) and case (b) in Fig. 3 can make the distance between two vertices in V be equal to 2; otherwise, the distance between two vertices in V is greater than or equal to 4.

  In case (a), the position of v_j makes $d_{T^*}(v_i, v_j)$ and $d_{T^*}(v_j, v_i)$ be equal to 2. For each vertex v in the position similar to that of v_j, v makes two distances be equal to 2 in $R_{T^*}(V, V)$. Let n_1 be the number of non-hubs $v \in V$ satisfying $f^*(v) \in V$.

  In case (b), for vertices v_l, v_m, v_n, there are six distances equal to 2 in $R_{T^*}(V, V)$. For each vertex v in the position similar to that of v_l, v makes two distances be equal to 2 in $R_{T^*}(V, V)$. Let n_2 be the number

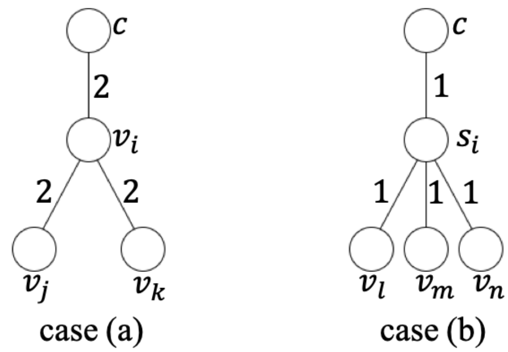

Figure 3: The two cases that can make the distance between two vertices in V be equal to 2, where $v_i, v_j, v_k, v_l, v_m, v_n \in V$, $s_i \in S$, and s_i covers v_l, v_m, v_n.

of non-hubs $v \in V$ satisfying $f^*(v) \in S$ and $d_{T^*}(v, f^*(v)) = 1$. We have

$$\begin{aligned} R_{T^*}(V,V) &\geq 3q(3q-1)4 - n_1 \cdot 2(4-2) - n_2 \cdot 2(4-2) \\ &= 3q(3q-1)4 - (n_1+n_2) \cdot 2(4-2) \\ &\geq 3q(3q-1)4 - 3q \cdot 2(4-2) \quad (\text{since } n_1+n_2 \leq 3q) \\ &= 36q^2 - 24q \end{aligned}$$

- For $R_{T^*}(S,S)$:

$$\begin{aligned} R_{T^*}(S,S) &\geq R_G(S,S) \quad (\text{by the triangle inequality}) \\ &= m(m-1) \cdot 2 \\ &= 2m^2 - 2m \end{aligned}$$

- For $R_{T^*}(S,V)$:

  Only case (c) and case (d) in Fig. 4 can make the distance between a vertex in S and a vertex in V be equal to 1; otherwise, the distance between a vertex in S and a vertex in V is greater than or equal to 3.

  In case (c), the position of s_i makes $d_{T^*}(s_i, v_i)$ be equal to 1. For each vertex s in the position similar to that of s_i, s makes one distance be equal to 1 in $R_{T^*}(S,V)$. Let n_1 be the number of non-hubs $s \in S$ satisfying $f^*(s) \in V$ and $d_{T^*}(s, f^*(s)) = 1$.

  In case (d), the position of v_j makes $d_{T^*}(s_k, v_j)$ be equal to 1. For each vertex v in the position similar to that of v_j, v makes one distance be

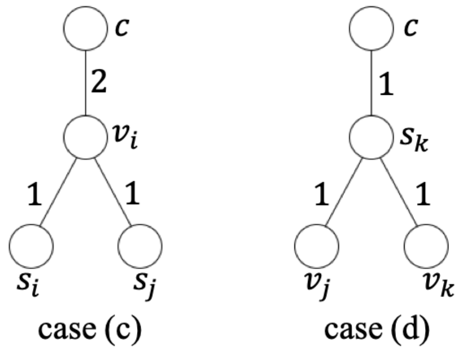

Figure 4: The two cases that can make the distance between a vertex in S and a vertex in V be equal to 1, where $v_i, v_j, v_k \in V$, $s_i, s_j, s_k \in S$, s_i, s_j cover v_i, and s_k covers v_j, v_k.

equal to 1 in $R_{T^*}(S, V)$. Let n_2 be the number of non-hubs $v \in V$ satisfying $f^*(v) \in S$ and $d_{T^*}(v, f^*(v)) = 1$. Note that both n_1 and n_2 are the number of some non-hubs. The number of all non-hubs is $3q$, so $n_1 + n_2 \leq 3q$. We have

$$\begin{aligned} R_{T^*}(S, V) &\geq m \cdot 3q \cdot 3 - n_1(3-1) - n_2(3-1) \\ &= m \cdot 3q \cdot 3 - (n_1 + n_2) \cdot 2 \\ &\geq m \cdot 3q \cdot 3 - 3q \cdot 2 \quad (\text{since } n_1 + n_2 \leq 3q) \\ &= 9mq - 6q \end{aligned}$$

- For $R_{T^*}(\{c\}, V \cup S)$:

$$\begin{aligned} R_{T^*}(\{c\}, V \cup S) &= R_{T^*}(\{c\}, V \cup S \setminus \{s_x\}) + d_{T^*}(c, s_x) \\ &\geq R_G(\{c\}, V \cup S \setminus \{s_x\}) + d_{T^*}(c, s_x) \\ &= 3q \cdot 2 + (m-1) \cdot 1 + d_{T^*}(c, s_x) \\ &\geq 3q \cdot 2 + m - 1 + 3 \quad (\text{since } s_x \text{ is a non-hub}) \\ &= 6q + m + 2 \end{aligned}$$

Hence,

$$\begin{aligned} C(T^*) &= R_{T^*}(V, V) + R_{T^*}(S, S) + 2R_{T^*}(S, V) + 2R_{T^*}(\{c\}, S \cup V) \\ &\geq (36q^2 - 24q) + (2m^2 - 2m) + 2(9mq - 6q) + 2(6q + m + 2) \\ &= 18mq + 2m^2 + 36q^2 - 24q + 4 \\ &> C(T) \end{aligned}$$

We can see that $C(T^*) > C(T)$. This contradicts the fact that T^* is an optimal solution. This completes the proof. $\square$

Claim 2. *In T^*, all vertices in V are adjacent to the vertex in S that covers it.*

Proof. In T^*, suppose a vertex v_x in V is adjacent to a vertex s_y in S that does not cover v_x. We can construct another solution T' (see Fig. 5) for the SpHRP by removing the edge (s_y, v_x) in T^* and adding the edge connecting v_x and a vertex s_x in S that covers v_x. For each vertex $u \in V \cup S \setminus \{s_x, v_x\}$, we see that $d_{T^*}(s_y, u) = d_{T'}(s_y, u)$. We also see that the shortest path from v_x to u goes through s_y in T^*. We have

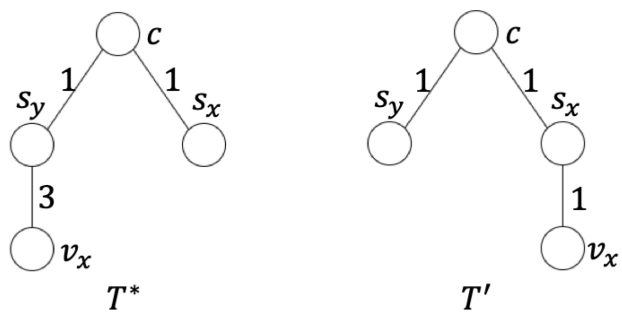

Figure 5: T^* and T' in the proof of Claim 2.

$$
\begin{aligned}
C(T^*) - C(T') &= 2R_{T^*}(\{v_x\}, V \cup S \cup \{c\}) - 2R_{T'}(\{v_x\}, V \cup S \cup \{c\}) \\
&= 2R_{T^*}(\{v_x\}, V \cup S \setminus \{s_x, v_x\}) + 2R_{T^*}(\{v_x\}, \{c, s_x, v_x\}) \\
&\quad -2R_{T'}(\{v_x\}, V \cup S \setminus \{s_x, v_x\}) - 2R_{T'}(\{v_x\}, \{c, s_x, v_x\}) \\
&= 2R_{T^*}(\{v_x\}, V \cup S \setminus \{s_x, v_x\}) + 2(4 + 5 + 0) \\
&\quad -2R_{T'}(\{v_x\}, V \cup S \setminus \{s_x, v_x\}) - 2(2 + 1 + 0) \\
&\geq 2R_{T^*}(\{s_y\}, V \cup S \setminus \{s_x, v_x\}) \\
&\quad +2[(3q + m - 2)d_{T^*}(v_x, s_y)] \\
&\quad -2R_{T'}(\{s_y\}, V \cup S \setminus \{s_x, v_x\}) \\
&\quad -2[(3q + m - 2)d_{T'}(v_x, s_y)] + 12 \\
&= 12
\end{aligned}
$$

We can see that $C(T^*) > C(T')$. This contradicts the fact that T^* is an optimal solution. This completes the proof. $\square$

Claim 3. *In T^*, each hub is either adjacent to three non-hubs in V or not ajacent to any non-hubs in V.*

Proof. According to Claim 2, we can see that the maximum number of vertices in V that are adjacent to each hub is three. For any $v_1, v_2 \in V$, if v_1 and v_2 are adjacent to the same vertex in S, then $d_{T^*}(v_1, v_2) = 2$; otherwise, $d_{T^*}(v_1, v_2) = 4$. Therefore, $\min_{v \in V}(R_{T^*}(\{v\}, V)) = (3q-3)4 + 2 \cdot 2$.

- Case 1:

  In T^*, suppose a vertex s_x in S has only two children v_x, v_y that belong to V. Note that $R_{T^*}(\{v_x\}, V) = R_{T^*}(\{v_y\}, V) = (3q-2)4 + 2$.

$$\begin{aligned} C(T^*) - C(T) &= R_{T^*}(V,V) - R_T(V,V) \\ &= R_{T^*}(V \setminus \{v_x, v_y\}, V) + R_{T^*}(\{v_x\}, V) \\ &\quad + R_{T^*}(\{v_y\}, V) - R_T(V,V) \\ &\geq (3q-2)[(3q-3)\cdot 4 + 2\cdot 2] + 2[(3q-2)\cdot 4 + 2] \\ &\quad - (36q^2 - 24q) \\ &= 4 \end{aligned}$$

  Thus, we obtain that $C(T^*) > C(T)$. This contradicts the fact that T^* is an optimal solution. Therefore, in T^* there is no $s_x \in S$ having exactly two children in V.

- Case 2:

  In T^*, suppose a vertex s_x in S has only one child v_x that belongs to V. Note that $R_{T^*}(\{v_x\}, V) = (3q-1)4$.

$$\begin{aligned} C(T^*) - C(T) &= R_{T^*}(V,V) - R_T(V,V) \\ &= R_{T^*}(V \setminus \{v_x\}, V) + R_{T^*}(\{v_x\}, V) - R_T(V,V) \\ &\geq (3q-1)[(3q-3)\cdot 4 + 2\cdot 2] + (3q-1)\cdot 4 \\ &\quad - (36q^2 - 24q) \\ &= 4 \end{aligned}$$

  Thus, we see that $C(T^*) > C(T)$. This contradicts the fact that T^* is an optimal solution. Therefore, in T^* there is no $s_x \in S$ having exactly one child in V.

This completes the proof of the claim. □

Claim 4. *The SpHRP has an optimal solution T^* with $C(T^*) \leq 18mq + 2m^2 + 36q^2 - 24q$ if and only if Rest-X3C has a solution.*

Proof. If the Rest-X3C has a solution $\mathcal{S}'$, we can construct a solution T for the SpHRP problem according to $\mathcal{S}'$ in the following way. Let all $s_x \in \mathcal{S}$ be hubs adjacent to c in T and for each $s_i \in \mathcal{S}'$ let the three elements in s_i be non-hubs adjacent to s_i (see Fig. 2). We see that

$$C(T^*) \leq C(T) = 18mq + 2m^2 + 36q^2 - 24q.$$

If the SpHRP has an optimal solution T^* with $C(T^*) \leq 18mq + 2m^2 + 36q^2 - 24q$, then Claims 1 to 3 hold. According to Claims 1 to 3, T^* has q hubs with three children and $m - q$ hubs with no child. Then, we can create a set $\mathcal{S}'$ containing the elements of $\mathcal{S}$ which correspond to the hubs with children in T^*. Because every element of U occurs in exactly one member of $\mathcal{S}'$, $\mathcal{S}'$ is a solution of the Rest-X3C. This completes the proof. □

We have proven that we can obtain the input of the SpHRP from the input of the Rest-X3C in polynomial time and obtain the output of the Rest-X3C from the output of the SpHRP in polynomial time. If there exists a polynomial time algorithm that solves the SpHRP, then the Rest-X3C can be solved in polynomial time. However, Rest-X3C is a NP-hard problem. This implies that SpHRP is also an NP-hard problem. This completes the proof of Theorem 1. □

3. Approximation Algorithm

In this section, we give a 3-approximation algorithm[1] for the SpHRP. The algorithm is kind of intuitive. It first finds the vertex that has the minimal cost to the other vertices as hub 1; then it chooses $p - 1$ vertices closest to the root as the remaining hubs. All the other vertices are connected to hub 1.

Let $(G = (V, E, w), c, p)$ be the input of the SpHRP, where $|V| = n$. Let T be the output of the proposed algorithm, $h_1, h_2, ..., h_p$ represent each hub in T, and $l_1, l_2, ..., l_{n-p-1}$ represent each non-hub in T. Let T^* denote an optimal solution of the SpHRP. Let $h_1^*, h_2^*, ..., h_p^*$ represent each hub in T^*, where $h_1^*, h_2^*, ..., h_p^*$ are sorted by the distance to c in T^* increasingly. Let $l_1^*, l_2^*, ..., l_{n-p-1}^*$ represent each non-hub in T^*, where $l_1^*, l_2^*, ..., l_{n-p-1}^*$ are sorted by the distance to c in T^* increasingly.

[1]The proposed algorithm improves the approximation ratio of the previous result in *ICS 2018*.

Algorithm 1: Approximation algorithm for SpHRP (G, c, p)

Step 1: Find c_s, where $c_s = \arg\min_{u \in V \setminus \{c\}} \left(\sum_{v \in V} d_G(u, v) \right)$.
Step 2: Let vertex c_s be h_1 and connect c to h_1 in T.
Step 3: Select $p-1$ vertices $\{h_2, h_3, ..., h_p\}$ closest to c from $V \setminus \{c, h_1\}$ and connect them to c in T.
Step 4: Connect all vertices in $V \setminus \{c, h_1, h_2, ..., h_p\}$ to h_1 in T.
Step 5: Return T.

Before we start to prove that Algorithm 1 is a 3-approximation algorithm, we prove two technical lemmas first.

Lemma 2. $C(T^*) \geq (n-1) \sum_{v \in V} d_G(c_s, v) + \sum_{u \in V} d_{T^*}(c, u)$

Proof.

$$\begin{aligned}
C(T^*) &= \sum_{u \in V} \sum_{v \in V} d_{T^*}(u, v) \\
&= \sum_{u \in V \setminus \{c\}} \sum_{v \in V} d_{T^*}(u, v) + \sum_{v \in V} d_{T^*}(c, v) \\
&\geq \sum_{u \in V \setminus \{c\}} \sum_{v \in V} d_G(u, v) + \sum_{v \in V} d_{T^*}(c, v) \\
&\quad \text{(using the triangle inequality)} \\
&\geq (n-1) \min_{u \in V \setminus \{c\}} \Big(\sum_{v \in V} d_G(u, v) \Big) + \sum_{v \in V} d_{T^*}(c, v) \\
&= (n-1) \sum_{v \in V} d_G(c_s, v) + \sum_{v \in V} d_{T^*}(c, v) \\
&\quad \text{(due to the selection of } c_s \text{ in Algorithm 1)}
\end{aligned}$$

This completes the proof. □

Lemma 3. $C(T^*) \geq 2(2n-2-p) \sum_{i=2}^{p} d_G(c, h_i)$

Proof. For a non-hub l in T^*, we use $f^*(l)$ to denote the hub adjacent to l in T^*. Note that $|V| \geq 2p+1$. There are at least p non-hubs in T^*. Define a set $L = \{l_2^*, l_3^*, ..., l_p^*\}$. We use H_i to denote the set that contains all the non-hubs adjacent to h_i^* in T^*. Define $k_i = |H_i|$ and $k_i' = |H_i \cap L|$. For a hub h_i^* in T^*, we define $k'(h_i^*) = k_i'$.

We need the following two claims to prove this lemma.

Claim 5. $(k'_i+1)(n-k'_i-1) \le (k_i+1)(n-k_i-1)$

Proof. Note that $n-p-1 \ge k_i$ given that the vertex c and all the hubs cannot be the children of h^*_i in T^*. Furthermore, note that $p-1 \ge k'_i$ given that $|L| = p-1$. Hence, we can obtain that $n \ge k_i+p+1 \ge k_i+k'_i+2$. In the following, we prove that $(k_i+1)(n-k_i-1)-(k'_i+1)(n-k'_i-1) \ge 0$.

$$\begin{aligned}
& (k_i+1)(n-k_i-1)-(k'_i+1)(n-k'_i-1) \\
= \;& (k_in-k_i^2-k_i+n-k_i-1)-(k'_in-{k'_i}^2-k'_i+n-k'_i-1) \\
= \;& n(k_i-k'_i)-(k_i^2-{k'_i}^2)-2(k_i-k'_i) \\
= \;& (k_i-k'_i)(n-k_i-k'_i-2) \\
\ge \;& 0 \quad (k_i \ge k'_i, \text{ given the definitions of } k_i \text{ and } k'_i)
\end{aligned}$$

This completes the proof the claim. □

Claim 6. $\sum_{i=1}^{p}[(2n-2)k'_i - 2{k'_i}^2 - 2k'_i] \cdot d_G(c,h^*_i) = \sum_{i=2}^{p}[(2n-2) - 2k'(f^*(l^*_i)) - 2] \cdot d_G(c, f^*(l^*_i))$

Proof. We see that $k'_1+k'_2+...+k'_p = p-1$ since $k'_1, k'_2, ..., k'_p$ only count these $p-1$ non-hubs: $l^*_2, l^*_3, ..., l^*_p$.

$$\begin{aligned}
& \sum_{i=1}^{p}[(2n-2)k'_i - 2{k'_i}^2 - 2k'_i] \cdot d_G(c,h^*_i) \\
= \;& \sum_{i=1}^{p} k'_i[(2n-2) - 2k'_i - 2] \cdot d_G(c,h^*_i) \\
= \;& \sum_{i=1}^{p}\sum_{j=1}^{k'_i}[(2n-2) - 2k'_i - 2] \cdot d_G(c,h^*_i) \\
= \;& \sum_{i=2}^{p}[(2n-2) - 2k'(f^*(l^*_i)) - 2] \cdot d_G(c, f^*(l^*_i)) \\
& \text{(we see this summation from the point of view of non-hubs)}
\end{aligned}$$

This completes the proof the claim. □

By Claim 5 and 6, we can prove Lemma 3 in the following. Note that the method below of calculating the routing cost is different from the method in Section 2. For each edge in T^*, we obtain the value by multiplying the number of occurrences of this edge calculated in the routing cost and the cost of this edge. Adding up the value obtained from each edge in T^*, we get the routing cost of T^*.

$$
\begin{aligned}
C(T^*) =\ & 2\sum_{i=1}^{p}(k_i+1)(n-k_i-1)d_G(c,h_i^*) + 2(n-1)\sum_{i=1}^{n-p-1} d_G(f^*(l_i^*),l_i^*) \\
\geq\ & 2\sum_{i=1}^{p}(k_i'+1)(n-k_i'-1)d_G(c,h_i^*) \\
& + 2(n-1)\sum_{i=1}^{n-p-1} d_G(f^*(l_i^*),l_i^*) \quad \text{(by Claim 5)} \\
\geq\ & 2(n-1)\sum_{i=2}^{p} d_G(c,h_i^*) + \sum_{i=1}^{p}[(2n-2)k_i' - 2k_i'^2 - 2k_i']d_G(c,h_i^*) \\
& + 2(n-1)\sum_{i=2}^{p} d_G(f^*(l_i^*),l_i^*) \\
& \text{(drop terms } 2(n-1)d_G(c,h_1^*),\ 2(n-1)d_G(f^*(l_1^*),l_1^*), \\
& \text{and } 2(n-1)\sum_{i=p+1}^{n-p-1} d_G(f^*(l_i^*),l_i^*)) \\
\geq\ & 2(n-1)\sum_{i=2}^{p} d_G(c,h_i) + \sum_{i=2}^{p}[(2n-2) - 2k'(f^*(l_i^*)) - 2]d_G(c,f^*(l_i^*)) \\
& + 2(n-1)\sum_{i=2}^{p} d_G(f^*(l_i^*),l_i^*) \\
& \text{(due to the selection of } h_2,...,h_p \text{ in Algorithm 1, and Claim 6)} \\
\geq\ & 2(n-1)\sum_{i=2}^{p} d_G(c,h_i) + \sum_{i=2}^{p}[(2n-2) - 2k'(f^*(l_i^*)) - 2]d_G(c,h_i) \\
\geq\ & \sum_{i=2}^{p}[(4n-4) - 2(p-1) - 2]d_G(c,h_i) \\
& \text{(the maximum output of } k'() \text{ is } p-1) \\
=\ & 2(2n-2-p)\sum_{i=2}^{p} d_G(c,h_i)
\end{aligned}
$$

This completes the proof of the lemma. □

By using Lemmas 2 and 3, we now show that Algorithm 1 is a 3-approximation algorithm.

Theorem 2. *There is a 3-approximation algorithm for the SpHRP problem running in $O(n^2)$ time, where n is the number of vertices in the input graph.*

Proof. It is easy to see that in time $O(n^2)$, Algorithm 1 returns a feasible solution of the SpHRP.

By using Lemmas 2 and 3, we now prove that the solution T satisfies the approximation ratio 3 by showing that $C(T) \leq 3C(T^*)$.

$$
\begin{aligned}
C(T) = {} & 2p(n-p)d_G(h_1, c) + 2(n-1)\left[\sum_{i=1}^{n-p-1} d_G(h_1, l_i) + \sum_{i=2}^{p} d_G(c, h_i)\right] \\
= {} & 2(n-1)d_G(h_1, c) + 2(n-1-p)(p-1)d_G(h_1, c) \\
& + 2(n-1)\left[\sum_{i=1}^{n-p-1} d_G(h_1, l_i) + \sum_{i=2}^{p} d_G(c, h_i)\right] \\
\leq {} & 2(n-1)d_G(h_1, c) + 2(n-1-p)\sum_{i=2}^{p}\left[d_G(h_1, h_i) + d_G(h_i, c)\right] \\
& + 2(n-1)\left[\sum_{i=1}^{n-p-1} d_G(h_1, l_i) + \sum_{i=2}^{p} d_G(c, h_i)\right] \\
& \text{(using the triangle inequality)} \\
= {} & 2(n-1)\sum_{v \in V} d_G(h_1, v) - 2p\sum_{i=2}^{p} d_G(h_1, h_i) \\
& + 2(2n-2-p)\sum_{i=2}^{p} d_G(c, h_i) \\
\leq {} & 2C(T^*) - 2\sum_{u \in V} d_{T^*}(c, u) - 2p\sum_{i=2}^{p} d_G(h_1, h_i) + C(T^*) \\
& \text{(using Lemmas 2 and 3)} \\
\leq {} & 3C(T^*)
\end{aligned}
$$

This completes the proof of Theorem 2. □

4. Tightness of the Proposed Approximation Algorithm

In this section, we present an example for Algorithm 1 showing that the approximation ratio is at least $2 - \epsilon$ for any $\epsilon > 0$.

Lemma 4. *For any $\epsilon > 0$, the approximation ratio of Algorithm 1 is at least $2 - \epsilon$.*

Proof. Let $\epsilon > 0$. Define $V = \{c, v_1, v_2, v_{1,1}, v_{1,2}, ..., v_{1,x}, v_{2,1}, v_{2,2}, ..., v_{2,x}\}$, where $x = \lceil 3/\epsilon \rceil$. We construct a metric graph $G = (V, E, w)$. Let $(G, c, p = 2)$ be the input of the SpHRP. Define $a = \lceil 3/\epsilon \rceil$ and $b = 1$. We define the cost of each edge in G as follows.

- For c
  - $w(c, v_1) = a$.
  - $w(c, v_2) = a$.
  - $w(c, v_{1,i}) = a + b$ if $1 \leq i \leq x$.
  - $w(c, v_{2,i}) = a + b$ if $1 \leq i \leq x$.
- For v_1
  - $w(v_1, v_2) = 2a$.
  - $w(v_1, v_{1,i}) = b$ if $1 \leq i \leq x$.
  - $w(v_1, v_{2,i}) = 2a + b$ if $1 \leq i \leq x$.
- For v_2
  - $w(v_2, v_{1,i}) = 2a + b$ if $1 \leq i \leq x$.
  - $w(v_2, v_{2,i}) = b$ if $1 \leq i \leq x$.
- For $v_{1,i}$ with $1 \leq i \leq x$
  - $w(v_{1,i}, v_{1,j}) = 2b$ if $1 \leq j \leq x$ and $j \neq i$.
  - $w(v_{1,i}, v_{2,j}) = 2a + 2b$ if $1 \leq j \leq x$.
- For $v_{2,i}$ with $1 \leq i \leq x$
  - $w(v_{2,i}, v_{2,j}) = 2b$ if $1 \leq j \leq x$ and $j \neq i$.

It is easy to verify that G is a metric graph since any three vertices u, v, r in G satisfy $w(u, v) + w(v, r) \geq w(u, r)$.

We can get a solution T_1 (see Fig. 6) of the input instance $(G, c, p = 2)$ of the SpHRP. The routing cost of T_1 is

$$\begin{aligned} C(T_1) = &\ 2 \cdot 1 \cdot (2x + 2) \cdot b \cdot 2x + 2 \cdot (x + 1) \cdot (x + 2) \cdot a \cdot 2 \\ = &\ (4x^2 + 12x + 8)a + (8x^2 + 8x)b \end{aligned}$$

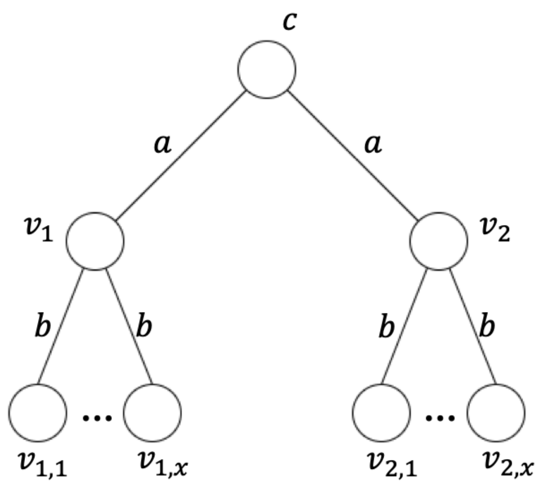

Figure 6: The solution T_1 of the example.

Let T^* denote an optimal solution of the SpHRP. We have $C(T^*) \leq C(T_1) = (4x^2 + 12x + 8)a + (8x^2 + 8x)b$.

Let T_2 denote the output of Algorithm 1 (see Fig. 7). The routing cost of T_2 is

$$\begin{aligned} C(T_2) = &\ 2 \cdot 1 \cdot (2x + 2) \cdot b \cdot x + 2 \cdot 1 \cdot (2x + 2) \cdot a \cdot 1 + 2 \cdot 2 \cdot (2x + 1) \cdot a \cdot 1 \\ &+ 2 \cdot 1 \cdot (2x + 2) \cdot (2a + b) \cdot x \\ = &\ (8x^2 + 20x + 8)a + (8x^2 + 8x)b \end{aligned}$$

We obtain that

$$\frac{C(T_2)}{C(T^*)} \geq \frac{C(T_2)}{C(T_1)}.$$

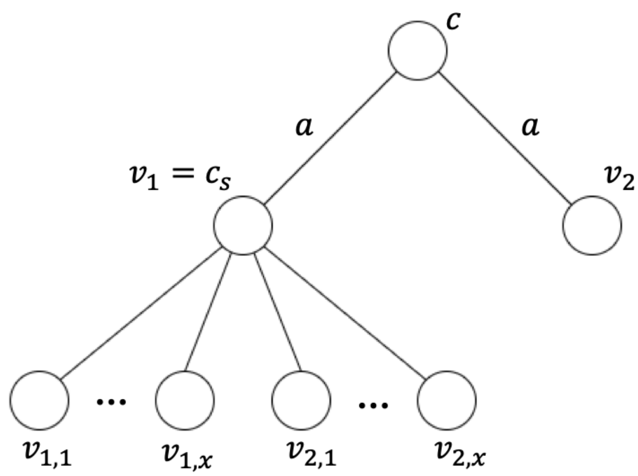

Figure 7: The output T_2 of the proposed approximation algorithm.

Moreover,

$$\begin{aligned}
\frac{C(T_2)}{C(T_1)} &= \frac{(8x^2+20x+8)a+(8x^2+8x)b}{(4x^2+12x+8)a+(8x^2+8x)b} \\
&= \frac{8x^3+28x^2+16x}{4x^3+20x^2+16x} \text{ (using } a = x \text{ and } b = 1) \\
&= 2 - \frac{12x^2+16x}{4x^3+20x^2+16x} \\
&\geq 2 - \frac{(12x^2+16x)\cdot\epsilon}{(4x^2\cdot\frac{3}{\epsilon}+20x\cdot\frac{3}{\epsilon}+16\cdot\frac{3}{\epsilon})\cdot\epsilon} \quad \text{(since } x = \left\lceil\frac{3}{\epsilon}\right\rceil \geq \frac{3}{\epsilon}) \\
&= 2 - \frac{12x^2+16x}{12x^2+60x+48}\cdot\epsilon \geq 2-\epsilon
\end{aligned}$$

Therefore, the approximation ratio is at least $2-\epsilon$ for any $\epsilon > 0$. This completes the proof. □

By Lemma 4, we cannot see whether the analysis of Algorithm 1 given in Section 3 is tight or not.

5. Conclusion

In this paper, we introduced and analyzed the STAR p-HUB ROUTING COST PROBLEM (SpHRP). We showed that the SpHRP is NP-hard. A 3-

approximation algorithm was proposed to solve the same problem running in time $O(n^2)$. We also presented an example for the 3-approximation algorithm showing that the approximation ratio is at least $2 - \epsilon$ for any $\epsilon > 0$. So far, due to the gap between 3 and $2 - \epsilon$ for any $\epsilon > 0$, it is still open whether the analysis of the 3-approximation algorithm given in this paper is tight or not. If the analysis is tight, there should exist a counterexample which achieves approximation ratio $3 - \epsilon$ for any $\epsilon > 0$. Otherwise, there should be a more elegant analysis. An appealing future research direction is to determine whether there exists an α-approximation algorithm running in polynomial time where $1 < \alpha < 3$, or to prove that for any $\epsilon > 0$, it is NP-hard to approximate the SpHRP to a ratio $c - \epsilon$ for some constant $c > 1$. Other future direction is to study the variants of SpHRP, including the at most p variant and a variant with discount factor (with the reduction on the costs between hubs and root).

References

[1] C.-C. Lin, J.-Y. Lin, Y.-C. Chen, The capacitated p-hub median problem with integral constraints: An application to a chinese air cargo network, Applied Mathematical Modelling 36 (6) (2012) 2777–2787.

[2] A. D. Vasconcelos, C. D. Nassi, L. A. Lopes, The uncapacitated hub location problem in networks under decentralized management, Computers & Operations Research 38 (12) (2011) 1656–1666.

[3] R. Ishfaq, C. R. Sox, Design of intermodal logistics networks with hub delays, European Journal of Operational Research 220 (3) (2012) 629–641.

[4] M. E. O'Kelly, A quadratic integer program for the location of interacting hub facilities, European Journal of Operational Research 32 (3) (1987) 393–404.

[5] R. Z. Farahani, M. Hekmatfar, A. B. Arabani, E. Nikbakhsh, Hub location problems: A review of models, classification, solution techniques, and applications, Computers & Industrial Engineering 64 (4) (2013) 1096–1109.

[6] H. Liang, The hardness and approximation of the star p-hub center problem, Operations Research Letters 41 (2) (2013) 138–141.

[7] I. Contreras, Hub location problems, in: Location science, Springer, 2015, pp. 311–344.

[8] L.-H. Chen, D.-W. Cheng, S.-Y. Hsieh, L.-J. Hung, C.-W. Lee, B. Y. Wu, Approximation algorithms for single allocation k-hub center problem, in: Proceedings of the 33rd Workshop on Combinatorial Mathematics and Computation Theory (CMCT 2016), 2016, pp. 13–18.

[9] L.-H. Chen, D.-W. Cheng, S.-Y. Hsieh, L.-J. Hung, C.-W. Lee, B. Y. Wu, Approximation algorithms for the star k-hub center problem in metric graphs, in: Computing and Combinatorics - 22nd International Conference, COCOON 2016, Ho Chi Minh City, Vietnam, August 2-4, 2016, Proceedings, Vol. 9797 of Lecture Notes in Computer Science, Springer, 2016, pp. 222–234. `doi:10.1007/978-3-319-42634-1\_18`.

[10] L.-H. Chen, S.-Y. Hsieh, L.-J. Hung, R. Klasing, The approximability of the p-hub center problem with parameterized triangle inequality, in: Y. Cao, J. Chen (Eds.), Computing and Combinatorics - 23rd International Conference, COCOON 2017, Hong Kong, China, August 3-5, 2017, Proceedings, Vol. 10392 of Lecture Notes in Computer Science, Springer, 2017, pp. 112–123. `doi:10.1007/978-3-319-62389-4\_10`.

[11] L.-H. Chen, D.-W. Cheng, S.-Y. Hsieh, L.-J. Hung, R. Klasing, C.-W. Lee, B. Y. Wu, Approximability and inapproximability of the star p-hub center problem with parameterized triangle inequality, Journal of Computer and System Sciences 92 (2018) 92–112.

[12] S.-Y. Hsieh, S.-S. Kao, A survey of hub location problems, Journal of Interconnection Networks 19 (01) (2019) 1940005.

[13] L.-H. Chen, S.-Y. Hsieh, L.-J. Hung, R. Klasing, Approximation algorithms for the p-hub center routing problem in parameterized metric graphs, Theoretical Computer Science 806 (2020) 271–280.

[14] S. A. Alumur, J. F. Campbell, I. A. Contreras, B. Y. Kara, V. Marianov, M. E. O'Kelly, Perspectives on modeling hub location problems, Eur. J. Oper. Res. 291 (1) (2021) 1–17. `doi:10.1016/j.ejor.2020.09.039`.

[15] X. Wang, G. Chen, Y. Chen, G. Lin, Y. Wang, A. Zhang, Improved hardness and approximation results for single allocation hub location problems, Theor. Comput. Sci. 864 (2021) 10–19. `doi:10.1016/j.tcs.2021.02.003`.

[16] M. Iwasa, H. Saito, T. Matsui, Approximation algorithms for the single allocation problem in hub-and-spoke networks and related metric labeling problems, Discrete Applied Mathematics 157 (9) (2009) 2078–2088.

[17] H. Yaman, Star p-hub median problem with modular arc capacities, Computers & Operations Research 35 (9) (2008) 3009–3019.

[18] H. Yaman, S. Elloumi, Star p-hub center problem and star p-hub median problem with bounded path lengths, Computers & operations research 39 (11) (2012) 2725–2732.

[19] H. Tikani, M. Honarvar, Y. Zare Mehrjerdi, Joint optimization of star p-hub median problem and seat inventory control decisions considering a hybrid routing transportation system, International Journal of Supply and Operations Management 3 (3) (2016) 1442–1465.

[20] T. C. Hu, Optimum communication spanning trees, SIAM Journal on Computing 3 (3) (1974) 188–195.

[21] D. S. Johnson, J. K. Lenstra, A. R. Kan, The complexity of the network design problem, Networks 8 (4) (1978) 279–285.

[22] C.-W. Lin, B. Y. Wu, On the minimum routing cost clustered tree problem, Journal of Combinatorial Optimization 33 (3) (2017) 1106–1121.

[23] B. Y. Wu, G. Lancia, V. Bafna, K.-M. Chao, R. Ravi, C. Y. Tang, A polynomial-time approximation scheme for minimum routing cost spanning trees, SIAM Journal on Computing 29 (3) (2000) 761–778.

[24] S. V. Ravelo, C. E. Ferreira, A PTAS for the metric case of the optimum weighted source–destination communication spanning tree problem, Theoretical Computer Science 771 (2019) 9–22.

[25] M. R. Garey, D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences), first edition Edition, W. H. Freeman, 1979.