On the Complexity of the Star p-hub Center Problem with Parameterized Triangle Inequality

Li-Hsuan Chen[1], Sun-Yuan Hsieh[1], Ling-Ju Hung[1(✉)], Ralf Klasing[2], Chia-Wei Lee[1], and Bang Ye Wu[3]

[1] Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan 701, Taiwan
{clh100p,hunglc}@cs.ccu.edu.tw, hsiehsy@mail.ncku.edu.tw, cwlee@csie.ncku.edu.tw

[2] CNRS, LaBRI, Université de Bordeaux, 351 Cours de la Libération, 33405 Talence Cedex, France
ralf.klasing@labri.fr

[3] Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi 62102, Taiwan
bangye@cs.ccu.edu.tw

Abstract. A complete weighted graph $G = (V, E, w)$ is called Δ_β-metric, for some $\beta \geq 1/2$, if G satisfies the β-triangle inequality, *i.e.,* $w(u, v) \leq \beta \cdot (w(u, x) + w(x, v))$ for all vertices $u, v, x \in V$. Given a Δ_β-metric graph $G = (V, E, w)$ and a center $c \in V$, and an integer p, the Δ_β-Star p-Hub Center Problem (Δ_β-SpHCP) is to find a depth-2 spanning tree T of G rooted at c such that c has exactly p children and the diameter of T is minimized. The children of c in T are called hubs. For $\beta = 1$, Δ_β-SpHCP is NP-hard. (Chen *et al.*, *COCOON 2016*) proved that for any $\varepsilon > 0$, it is NP-hard to approximate the Δ_β-SpHCP to within a ratio $1.5 - \varepsilon$ for $\beta = 1$. In the same paper, a $\frac{5}{3}$-approximation algorithm was given for Δ_β-SpHCP for $\beta = 1$. In this paper, we study Δ_β-SpHCP for all $\beta \geq \frac{1}{2}$. We show that for any $\varepsilon > 0$, to approximate the Δ_β-SpHCP to a ratio $g(\beta) - \varepsilon$ is NP-hard and we give $r(\beta)$-approximation algorithms for the same problem where $g(\beta)$ and $r(\beta)$ are functions of β. If $\beta \leq \frac{3-\sqrt{3}}{2}$, we have $r(\beta) = g(\beta) = 1$, *i.e.,* Δ_β-SpHCP is polynomial time solvable. If $\frac{3-\sqrt{3}}{2} < \beta \leq \frac{2}{3}$, we have $r(\beta) = g(\beta) = \frac{1+2\beta-2\beta^2}{4(1-\beta)}$. For $\frac{2}{3} \leq \beta \leq 1$, $r(\beta) = \min\{\frac{1+2\beta-2\beta^2}{4(1-\beta)}, 1 + \frac{4\beta^2}{5\beta+1}\}$. Moreover, for $\beta \geq 1$,

Parts of this research were supported by the Ministry of Science and Technology of Taiwan under grants MOST 105–2221–E–006–164–MY3, MOST 103–2218–E–006–019–MY3, and MOST 103-2221-E-006-135-MY3. L.-H. Chen, L.-J. Hung (corresponding author), and C.-W. Lee are supported by the Ministry of Science and Technology of Taiwan under grants MOST 105–2811–E–006–071, –046, and –022, respectively.

R. Klasing–Part of this work was done while Ralf Klasing was visiting the Department of Computer Science and Information Engineering at National Cheng Kung University. This study has been carried out in the frame of the "Investments for the future" Programme IdEx Bordeaux - CPU (ANR-10-IDEX-03-02). Research supported by the LaBRI under the "Projets émergents" program.

D. Fotakis et al. (Eds.): CIAC 2017, LNCS 10236, pp. 152–163, 2017.
DOI: 10.1007/978-3-319-57586-5_14

we have $r(\beta) = \min\{\beta + \frac{4\beta^2-2\beta}{2+\beta}, 2\beta+1\}$. For $\beta \geq 2$, the approximability of the problem (*i.e.,* upper and lower bound) is linear in β.

1 Introduction

The *hub location problems* have various applications in transportation and telecommunication systems. Variants of hub location problems have been defined and well-studied in the literatures (see the two survey papers [1,15]). Suppose that we have a set of demand nodes that want to communicate with each other through some hubs in a network. A *single allocation hub location problem* requests each demand node can only be served by exactly one hub. Conversely, if a demand node can be served by several hubs, then this kind of hub location problem is called *multi-allocation*. Classical hub location problems ask to minimize the total cost of all origin-destination pairs (see *e.g.,* [27]). However, minimizing the total routing cost would lead to the result that the poorest service quality is extremely bad. In this paper, we consider a single hub location problem with min-max criterion, called Δ_β-Star p-Hub Center Problem which is different from the classic hub location problems. The min-max criterion is able to avoid the drawback of minimizing the total cost.

A complete weighted graph $G = (V, E, w)$ is called Δ_β-metric, for some $\beta \geq 1/2$, if the distance function $w(\cdot, \cdot)$ satisfies $w(v, v) = 0$, $w(u, v) = w(v, u)$, and the β-triangle inequality, *i.e.,* $w(u, v) \leq \beta \cdot (w(u, x) + w(x, v))$ for all vertices $u, v, x \in V$. (If $\beta > 1$ then we speak about *relaxed triangle inequality*, and if $\beta < 1$ we speak about *sharpened triangle inequality.*) Let u, v be two vertices in a tree T. Use $d_T(u, v)$ to denote the distance between u, v in T. Define $D(T) = \max_{u,v \in T} d_T(u, v)$ called the diameter of T. We give the definition of the Δ_β-Star p-Hub Center Problem as follows.

Δ_β-Star p-Hub Center Problem (Δ_β-SpHCP).

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

Output: A depth-2 spanning tree T^* rooted at c (called the central hub) such that c has exactly p children (called hubs) and the diameter of T^*, $D(T^*)$, is minimized.

Here, we assume that the number of non-hubs is at least as many as the number of hubs, *i.e.,* $|V| \geq 2p + 1$. The assumption $|V| \geq 2p + 1$ is reasonable because in real applications, a hub could be a post office or an airport, and a non-hub could be a mail post, a customer, or a passenger.

The Δ_β-SpHCP problem is a general version of the original Star p-Hub Center Problem (SpHCP) since the original problem assumes the input graph to be a metric graph, *i.e.,* $\beta = 1$. We use SpHCP to denote the Δ_β-SpHCP for $\beta = 1$. Yaman and Elloumi [28] showed that SpHCP is NP-hard and gave two integer programming formulations for the same problem. Liang [24] showed that SpHCP does not admit a $(1.25 - \varepsilon)$-approximation algorithm for any $\varepsilon > 0$ unless $P = NP$ and gave a 3.5-approximation algorithm. Recently, Chen *et al.* [17] reduced the gap between the upper and lower bounds of approximability of SpHCP. They showed that for any $\varepsilon > 0$, to approximate SpHCP to a ratio $1.5 - \varepsilon$ is NP-hard and gave 2-approximation and $\frac{5}{3}$-approximation algorithms for SpHCP.

The Single Allocation p-Hub Center Problem was introduced in [14, 26] which is similar to SpHCP with min-max criterion and well-studied in [16, 18, 23, 25]. The difference between the two problems is that the Single Allocation p-Hub Center Problem assumes that hubs are fully interconnected. Thus, for the Single Allocation p-Hub Center Problem, the communication between hubs is not necessary to go through a specified central hub c.

If $\beta = 1$, Δ_β-SpHCP is NP-hard and even NP-hard to have a $(1.5 - \varepsilon)$-approximation algorithm for any $\varepsilon > 0$ [17]. In this paper, we investigate the complexity of Δ_β-SpHCP parameterized by β-triangle inequality. The motivation of this research for $\beta < 1$ is to investigate whether there exists a large subclasses of input instances of Δ_β-SpHCP that can be solved in polynomial time or admit polynomial-time approximation algorithms with a reasonable approximation ratio. For $\beta \geq 1$, it is an interesting issue to see whether there exists a polynomial-time approximation algorithm with an approximation ratio linear in β.

The well-known concept of *stability of approximation* [10, 12, 22] is used in our study. The idea behind this concept is to find a parameter (characteristic) of the input instances that captures the hardness of particular inputs. An approximation algorithm is called *stable* with respect to this parameter, if its approximation ratio grows with this parameter but not with the size of the input instances. A nice example is the Traveling Salesman Problem (TSP) that does not admit any polynomial-time approximation algorithm with an approximation ratio bounded by a polynomial in the size of the input instance, but is $\frac{3}{2}$-approximable for metric input instances. Here, one can characterize the input instances by their "distance" to metric instances. This can be expressed by the β-triangle inequality for any $\beta \geq \frac{1}{2}$.

In a sequence of papers [2, 3, 5, 9–11, 13], it was shown that one can partition the set of all input instances of TSP into infinitely many subclasses according to the degree of violation of the triangle inequality, and for each subclass one can guarantee upper and lower bounds on the approximation ratio. Similar studies were performed for the problem of constructing 2-connected spanning subgraphs of a given complete graph whose edge weights obey the β-triangle inequality [6], and for the problem of finding, for a given positive integer $k \geq 2$ and an edge-weighted graph G, a minimum k-edge- or k-vertex-connected spanning subgraph [7, 8], demonstrating that for these problems the β-triangle inequality can serve as a measure of hardness of the input instances.

In Table 1, we list the main results of this paper. We prove that for any $\varepsilon > 0$, to approximate Δ_β-SpHCP to a ratio $g(\beta) - \varepsilon$ is NP-hard where $\beta \geq \frac{3-\sqrt{3}}{2}$ and $g(\beta)$ is a function of β. We give $r(\beta)$-approximation algorithms for Δ_β-SpHCP. If $\beta \leq \frac{3-\sqrt{3}}{2}$, we have $r(\beta) = g(\beta) = 1$, *i.e.*, Δ_β-SpHCP is polynomial time solvable. If $\frac{3-\sqrt{3}}{2} < \beta \leq \frac{2}{3}$, we have $r(\beta) = g(\beta)$. For $\frac{2}{3} \leq \beta \leq 1$, $r(\beta) =$

Table 1. The main results where Δ_β-SpHCP cannot be approximated within $g(\beta)$ and has an $r(\beta)$-approximation algorithm.

β	Lower bound $g(\beta)$	Upper bound $r(\beta)$
$[\frac{1}{2}, \frac{3-\sqrt{3}}{2}]$	1	1
$(\frac{3-\sqrt{3}}{2}, \frac{2}{3}]$	$\frac{1+2\beta-2\beta^2}{4(1-\beta)}$	$\frac{1+2\beta-2\beta^2}{4(1-\beta)}$
$[\frac{2}{3}, 0.7737\ldots]$	$\frac{5\beta+1}{4}$	$\frac{1+2\beta-2\beta^2}{4(1-\beta)}$
$[0.7737\ldots, 1]$	$\frac{5\beta+1}{4}$	$1+\frac{4\beta^2}{5\beta+1}$
$[1, 2]$	$\beta+\frac{1}{2}$	$\beta+\frac{4\beta^2-2\beta}{2+\beta}$
$[2, \infty)$	$\beta+\frac{1}{2}$	$2\beta+1$

$\min\{\frac{1+2\beta-2\beta^2}{4(1-\beta)}, 1+\frac{4\beta^2}{5\beta+1}\}$ and $g(\beta) = \frac{5\beta+1}{4}$. Moreover, for $\beta \geq 1$, we have $r(\beta) = \min\{\beta + \frac{4\beta^2-2\beta}{2+\beta}, 2\beta+1\}$ and $g(\beta) = \beta + \frac{1}{2}$. For $\beta \geq 2$, the approximability of the problem (*i.e.*, upper and lower bound) is linear in β.

For a vertex v in a tree T, we use $N_T(v)$ to denote the set of vertices adjacent to v in T and $N_T[v] = N_T(v) \cup \{v\}$. Let $f(v)$ be the parent of v in T and $f(v) = v$ if v is the root of T. Let T^* be an optimal solution of Δ_β-SpHCP in a given β-metric graph $G = (V, E, w)$. For a non-hub x in T^*, we use $f^*(x)$ to denote the hub in T^* that is adjacent to x. We use $\tilde{T}$ to denote the best solution among all solutions in $\mathcal{T}$ where $\mathcal{T}$ is the collection of all solutions satisfying that all non-hubs are adjacent to the same hub for Δ_β-SpHCP in a given β-metric graph $G = (V, E, w)$.

We close this section with the following theorem. Due to the limitation of space, we omit the proof.

Theorem 1. *Let $\beta > \frac{3-\sqrt{3}}{2}$. For any $\varepsilon > 0$, to approximate Δ_β-SpHCP to a factor $g(\beta) - \varepsilon$ is NP-hard where*

(i) $g(\beta) = \frac{1+2\beta-2\beta^2}{4(1-\beta)}$ *if* $\frac{3-\sqrt{3}}{2} < \beta \leq \frac{2}{3}$;
(ii) $g(\beta) = \frac{5\beta+1}{4}$ *if* $\frac{2}{3} \leq \beta \leq 1$;
(iii) $g(\beta) = \beta + \frac{1}{2}$ *if* $\beta \geq 1$.

2 Polynomial Time Algorithms

In this section, we show that for $\frac{1}{2} \leq \beta \leq \frac{3-\sqrt{3}}{2}$, Δ_β-SpHCP can be solved in polynomial time. Besides, we give polynomial time approximation algorithms for Δ_β-SpHCP for $\beta > \frac{3-\sqrt{3}}{2}$. For $\frac{3-\sqrt{3}}{2} < \beta \leq \frac{2}{3}$, our approximation algorithm achieves the factor that closes the gap between the upper and lower bounds of approximability for Δ_β-SpHCP.

Due to the limitation of space, we omit some proofs in this section.

Lemma 1. *Let $\frac{1}{2} \leq \beta < 1$. Then the following statements hold.*

(i) There exists a solution $\tilde{T}$ satisfying that all non-hubs are adjacent to the same hub and $D(\tilde{T}) \leq \max\{1, \frac{1+2\beta-2\beta^2}{4(1-\beta)}\} \cdot D(T^*)$.
(ii) There exists a polynomial time algorithm to compute a solution T such that $D(T) = D(\tilde{T})$.

According to Lemma 1, we obtain the following results.

Lemma 2. *Let $\frac{1}{2} \leq \beta \leq 0.7737\ldots$. Then the following statements hold.*

1. *If $\beta \leq \frac{3-\sqrt{3}}{2}$, then Δ_β-SpHCP can be solved in polynomial time.*
2. *If $\frac{3-\sqrt{3}}{2} < \beta \leq 0.7737\ldots$, there is a $\frac{1+2\beta-2\beta^2}{4(1-\beta)}$-approximation algorithm for Δ_β-SpHCP.*

Proof. Let T^* denote an optimal solution of the Δ_β-SpHCP problem. According to Lemma 1, there is a polynomial time algorithm for Δ_β-SpHCP to compute a solution T such that $D(T) \leq \max\{1, \frac{1+2\beta-2\beta^2}{4(1-\beta)}\} \cdot D(T^*)$.

If $\beta \leq \frac{3-\sqrt{3}}{2}$, $D(T) \leq \max\{1, \frac{1+2\beta-2\beta^2}{4(1-\beta)}\} \cdot D(T^*) = D(T^*)$.

If $\frac{3-\sqrt{3}}{2} < \beta \leq 0.7737\ldots$,

$$D(T) \leq \max\{1, \frac{1+2\beta-2\beta^2}{4(1-\beta)}\} \cdot D(T^*) = \frac{1+2\beta-2\beta^2}{4(1-\beta)} \cdot D(T^*).$$

This completes the proof. □

Lemma 3. *Let $0.7737\ldots \leq \beta \leq 1$. Then, there is a $(1 + \frac{4\beta^2}{5\beta+1})$-approximation algorithm for Δ_β-SpHCP.*

Proof. It is not hard to see that Algorithm 1 runs in polynomial time. Let T^* be an optimal solution of Δ_β-SpHCP. In this lemma, we show that for $0.7737\ldots \leq \beta \leq 1$, Algorithm 1 returns a solution T such that $D(T) \leq (1 + \frac{4\beta^2}{5\beta+1}) \cdot D(T^*)$. Let ℓ be the largest edge cost in T^* with one end vertex as a hub and the other end vertex as a non-hub. Note that both Algorithm APX1 and Algorithm APX2 guess all possible edges (y, z) to be the longest edge in T^* with y as a hub and z as a non-hub. Let T_1 and T_2 be the best solutions returned by Algorithm APX1 and Algorithm APX2, respectively.

Claim 1. $D(T_1) \leq D(T^*) + 4\beta\ell$.

Proof of Claim. We first show that for any two hubs u, v in T_1, $d_{T_1}(u, v) = w(u, c) + w(v, c) \leq D(T^*)$. Let T^* be an optimal solution of Δ_β-SpHCP. Let $f^*(u)$ and $f^*(v)$ be the parents of u and v in T^* respectively.

If $f^*(u) \neq f^*(v)$, there are three cases.

Algorithm 1. Approximation algorithm for Δ_β-SpHCP (G, c).

(i) Run Algorithm APX1.
(ii) Run Algorithm APX2.
(iii) Return the best solution found by Algorithms APX1 and APX2.

Algorithm APX1
Guess the correct edge (y, z) where $w(y, z) = \ell$ is the largest edge cost in an optimal solution T^* with y as a hub and z as a non-hub. Let $U := V \setminus \{c\}$ and $h_1 = y$. Let T_1 be the tree found by the following steps and H be the set of children of c in T_1. Initialize $H = \emptyset$.

(i) Add edge (h_1, c) in the tree T, let $H := H \cup \{h_1\}$, and let $U := U \setminus \{h_1\}$.
(ii) For $x \in U$, if $w(h_1, x) \leq \ell$, add edges (x, h_1) in T and let $U := U \setminus \{x\}$.
(iii) While $i = |H| + 1 \leq p$ and $U \neq \emptyset$,
  – choose $v \in U$, let $h_i = v$, add edge (h_i, c) in T, let $U := U \setminus \{v\}$, and let $H := H \cup \{h_i\}$;
  – for $x \in U$, if $w(x, h_i) \leq 2\beta\ell$, then add edge (x, h_i) in T and $U := U \setminus \{x\}$.
(iv) If $|H| < p$ and $U = \emptyset$, we change the shape of T by selecting $p - |H|$ vertices closest to c from the second layer to be the children of c, call the new tree T_1; otherwise let $T_1 := T$.

Algorithm APX2
Guess the correct edge (y, z) where $w(y, z) = \ell$ is the largest edge cost in an optimal solution T^* with y as a hub and z as a non-hub. Let T_2 be the tree found by the following steps.

(i) Let y be the child of c in T_2.
(ii) Pick $(p - 1)$ vertices $\{v_1, v_2, \ldots, v_{p-1}\}$ closest to c from $U \setminus \{y, z\}$. Let $N_{T_2}(c) = \{y, v_1, v_2, \ldots, v_{p-1}\}$.
(iii) Let all vertices in $U \setminus \{v_1, v_2, \ldots, v_{p-1}, y\}$ be the children of y.

– Suppose that $f^*(u) = c$ and $f^*(v) \neq u$. Then

$$\begin{aligned} d_{T_1}(u, v) &= w(u, c) + w(v, c) \leq w(u, c) + w(c, f^*(v)) + w(f^*(v), v) \\ &= d_{T^*}(u, v) \leq D(T^*). \end{aligned}$$

– Suppose that $f^*(u) = c$ and $f^*(v) = u$. Since $w(u, v) \leq 2\beta\ell$, v is selected as a hub in Step (iv) of Algorithm APX1. Since in Step (iv), the algorithm select $(p - |H|)$ vertices closest to c from the second layer as hubs, there exists y' which is a hub in T^* and a non-hub in T_1 satisfying $w(y', c) \geq w(v, c)$. Thus,

$$d_{T_1}(u, v) = w(u, c) + w(v, c) \leq w(u, c) + w(y', c) = d_{T^*}(u, y') \leq D(T^*).$$

– Suppose that $f^*(u) \neq c$. Then

$$\begin{aligned} d_{T_1}(u, v) &= w(u, c) + w(v, c) \\ &\leq w(u, f^*(u)) + w(f^*(u), c) + w(c, f^*(v)) + w(f^*(v), v) \\ &= d_{T^*}(u, v) \leq D(T^*). \end{aligned}$$

If $f^*(u) = f^*(v) = c$, $d_{T_1}(u, v) = d_{T^*}(u, v) \leq D(T^*)$.

If $f^*(u) = f^*(v) \neq c$, then at most one of u, v is selected as a hub in Step (iii) of Algorithm APX1 since $w(u, v) \leq 2\beta\ell$, or both u and v are selected as hubs in Step (iv).

Suppose that u is selected as a hub in Step (iii) and and v is selected as a hub in Step (iv). We see that in Step (iv), the algorithms select $(p-|H|)$ vertices closest to c from the second layer as hubs. Thus, there exists y' which is a hub in T^* and a non-hub in T_1 satisfying $w(y', c) \geq w(v, c)$. We obtain that

$$d_{T_1}(u, v) = w(u, c) + w(v, c) \leq d_{T^*}(u, c) + w(y', c) = d_{T^*}(u, y') \leq D(T^*).$$

Suppose that both u, v are selected as hubs in Step (iv). We see that in Step (iv), the algorithm selects $(p - |H|)$ vertices closest to c from the second layer as hubs. Thus, there exist y_1, y_2 which are hubs in T^* and non-hubs in T_1 satisfying $w(y_1, c) \geq w(u, c)$ and $w(y_2, c) \geq w(v, c)$. We obtain that

$$d_{T_1}(u, v) = w(u, c) + w(v, c) \leq w(y_1, c) + w(y_2, c) = d_{T^*}(y_1, y_2) \leq D(T^*).$$

Notice that each non-hub v in T_1 is adjacent to a hub $f(v)$ in T_1 if $w(v, f(v)) \leq 2\beta\ell$.

Thus, for u, v in T_1, $d_{T_1}(u, v) \leq D(T^*) + 4\beta\ell$ and $D(T_1) \leq D(T^*) + 4\beta\ell$. This completes the proof of the claim. ■

Claim 2. $D(T_2) \leq \max\{D(T^*), (D(T^*) - \ell) + \beta(D(T^*) - \ell)\}$.

Proof of Claim. Let T^* be an optimal solution. For a vertex v, use $f^*(v)$ to denote the parent of v in T^*. Notice that Algorithm APX2 guesses all possible edges (y, z) to be a longest edge in T^* with one end vertex as a hub and the other end vertex as a non-hub. In the following we assume that $w(y, z) = \ell$ is the largest edge cost in T^* with y as a hub and z as a non-hub. Since Algorithm APX2 picks $(p - 1)$ vertices closest to c, y is a hub in both T^* and T_2, and $w(y, z) = \ell$, we see that for any hub v in T_2, $d_{T_2}(v, y) \leq D(T^*) - \ell$.

For two non-hubs u, v in T_2, we have the following three cases.

- $f^*(u) = f^*(v) = y$, we see that $d_{T_2}(u, v) = d_{T^*}(u, v) \leq D(T^*)$.
- $f^*(u) = y$ and $f^*(v) \neq y$, we see that

$$d_{T_2}(u, v) = w(u, y) + w(v, y) \leq \ell + \beta \cdot d_{T^*}(v, y) \leq \ell + \beta \cdot (D(T^*) - \ell) \leq D(T^*).$$

- $f^*(u) \neq y$ and $f^*(v) \neq y$, we see that

$$\begin{aligned} d_{T_2}(u, v) &= w(u, y) + w(v, y) \leq \beta \cdot d_{T^*}(u, y) + \beta \cdot d_{T^*}(v, y) \\ &\leq 2\beta(D(T^*) - \ell) \leq (D(T^*) - \ell) + \beta(D(T^*) - \ell). \end{aligned}$$

For a non-hub u and a hub v in T_2, there are two cases.

- If $f^*(u) = y$, we see that

$$d_{T_2}(u, v) = w(u, y) + d_{T_2}(v, y) \leq \ell + D(T^*) - \ell = D(T^*).$$

– If $f^*(u) \neq y$, we see that

$$d_{T_2}(u,v) = w(u,y) + d_{T_2}(v,y) \leq \beta \cdot (D(T^*) - \ell) + (D(T^*) - \ell).$$

For two hubs u, v in T_2, $u \neq y$ and $v \neq y$, we see that $d_{T_2}(u,v) \leq D(T^*)$ since y is a hub in T^* and Algorithm APX2 picks the other $(p-1)$ vertices closest to c as hubs.

Thus, $D(T_2) \leq \max\{D(T^*), (D(T^*) - \ell) + \beta(D(T^*) - \ell)\}$. This completes the proof of the claim. ■

Notice that if $\frac{\ell}{D(T^*)} \geq \frac{\beta}{1+\beta}$, $D(T_2) = D(T^*)$. Thus, the worst case approximation ratio happens when $\frac{\ell}{D(T^*)} < \frac{\beta}{1+\beta}$.

If $\frac{\ell}{D(T^*)} < \frac{\beta}{1+\beta}$, $D(T_2) \leq D(T^*) - \ell + \beta(D(T^*) - \ell)$. We see that the approximation ratio of Algorithm 1 is $r(\beta) = \min\{\frac{D(T_1)}{D(T^*)}, \frac{D(T_2)}{D(T^*)}\}$. The worst case approximation ratio of Algorithm 1 happens when $D(T_1) = D(T_2)$, *i.e.*,

$$D(T^*) + 4\beta\ell = (D(T^*) - \ell) + \beta \cdot (D(T^*) - \ell)$$

We obtain that $\frac{\ell}{D(T^*)} = \frac{\beta}{5\beta+1}$. Thus,

$$\begin{aligned} r(\beta) &= \min\{\tfrac{D(T_1)}{D(T^*)}, \tfrac{D(T_2)}{D(T^*)}\} \leq \min\{1 + \tfrac{4\beta^2}{5\beta+1}, 1 - \tfrac{\beta}{5\beta+1} + \beta(1 - \tfrac{\beta}{5\beta+1})\} \\ &= 1 + \tfrac{4\beta^2}{5\beta+1}. \end{aligned}$$

This completes the proof. □

In Lemma 4, we prove that if $1 \leq \beta \leq 2$, Algorithm 1 is a $(\beta + \frac{4\beta^2-2\beta}{2+\beta})$-approximation algorithm for Δ_β-SpHCP.

Lemma 4. *Let $1 \leq \beta \leq 2$. Then, there is a $(\beta + \frac{4\beta^2-2\beta}{2+\beta})$-approximation algorithm for Δ_β-SpHCP.*

If $\beta \geq 2$, we give Algorithm 2 to solve Δ_β-SpHCP and prove that Algorithm 2 is a $(2\beta + 1)$-approximation algorithm in Lemma 5.

Lemma 5. *Let $\beta \geq 2$. Then, there is a $(2\beta + 1)$-approximation algorithm for Δ_β-SpHCP.*

Proof. Let T^* be an optimal solution of Δ_β-SpHCP. Let (c, q) be the longest edge incident to c in T^*, $w(c,q) = \ell_0$, *i.e.*, $\ell_0 = \max_{v \in N_{T^*}(c)}\{w(v,c)\}$. Let ℓ_1 and ℓ_2 be the largest and second largest edge costs in T^* with one end vertex as a hub and the other end vertex as a non-hub. Note that it is possible that $\ell_1 = \ell_2$. Our algorithm is presented as Algorithm 2. Line 1 of Algorithm 2 guesses the values of ℓ_0, ℓ_1 and ℓ_2. We certainly do not know their exact values. However, since each of them has only polynomially many possible values, we can run the algorithm for all of their possible values and take the best solution. Therefore, in the following we assume that we know ℓ_0, ℓ_1 and ℓ_2. It is easy to see that $D(T^*) \geq \ell_0 + \ell_1$ and $D(T^*) \geq \ell_1 + \ell_2$.

Algorithm 2. Approximation algorithm for Δ_β-SpHCP (G, c).

1. Guess the correct values of ℓ_0, ℓ_1 and ℓ_2. Their meanings are provided in the proof.
2. $H \leftarrow \{v \in V \setminus \{c\} \mid w(v, c) \leq \ell_0\}$.
3. Create an instance $\mathcal{J}$ of the k-center problem with forbidden centers, in which $V \setminus \{c\}$ is the set of input vertices, H is the set of allowed centers, $k = p$, and the distance function (satisfying the β-triangle inequality) is the restriction of w to $V \setminus \{c\}$.
4. Apply the greedy approximation algorithm for the k-center problem with forbidden centers (Algorithm 3), to obtain an approximate solution of $\mathcal{J}$. Assume that $H^* \subseteq H$ is the set of centers opened in the solution.
5. **return** the solution that opens H^* as the set of p hubs and assigns each vertex in $V \setminus \{c\}$ to its nearest hub in H^*.

Algorithm 3. Approximation algorithm for k-center with forbidden centers.

1. // Let C be the input vertex set, $C' \subseteq C$ be the set of allowed centers, and w be the distance function on C satisfying the β-triangle inequality. Assume w.l.o.g. that $k \leq |C'|$.
2. $R \leftarrow C$; $S \leftarrow \emptyset$.
3. **while** $R \neq \emptyset$ and $|S| < k$ **do**
4.   Choose an arbitrary vertex $v \in C' \cap R$.
5. $B(v) \leftarrow \{u \in R \mid w(u, v) \leq \beta(\ell_1 + \ell_2)\}$.
6. $R \leftarrow R \setminus B(v)$; $S \leftarrow S \cup \{v\}$.
7. **end**
8. **if** $|S| < k$ and $R = \emptyset$ **then**
9.   select an arbitrary vertex set $S' \subseteq (C' \setminus S)$ of size $k - |S|$; $S \leftarrow S \cup S'$.
10. **return** S

Let T denote the solution returned by Algorithm 2. We next prove that Algorithm 2 is indeed a $(2\beta + 1)$-approximation algorithm for Δ_β-SpHCP by establishing an upper bound of $D(T)$. According to our choice of ℓ_0, the set H defined in line 2 contains all hub nodes in the optimal solution $N_{T^*}(c)$, *i.e.*, $N_{T^*}(c) \subseteq H$. In Line 3, we create an instance $\mathcal{J}$ of the k-center problem with forbidden centers. This problem is defined as follows: The input consists of a set C of demand points in a space satisfying the β-triangle inequality, a set $C' \subseteq C$ of allowed centers, and an integer k. The goal is to open k centers in C' such that the maximum distance between any vertex in C and its nearest center among the k opened centers is minimized. This problem is a generalization of the ordinary k-center problem (in which $C' = C$), and is a special case of the k-supplier problem (in which C' may not be a subset of C) [19–21]. There is a simple greedy approximation algorithm for this problem, which is presented in Algorithm 3. Its analysis is standard and is similar to that of the traditional k-center problem (see [19–21]), and thus is omitted here.

Hence, by applying the greedy approximation algorithm (Algorithm 3) to implement line 4 of Algorithm 2, we obtain a solution H^* of $\mathcal{J}$ with objective value at most $\beta(\ell_1 + \ell_2)$, that is,

$$\max_{v \in V \setminus \{c\}} \min_{h \in H^*} w(v,h) \le \beta(\ell_1 + \ell_2). \tag{1}$$

In Line 5 of Algorithm 2, a solution is returned that opens H^* as the set of p hubs. For each $v \in V \setminus (H^* \cup \{c\})$, let $f'(v) := \arg\min_{h \in H^*} w(v,h)$; *i.e.*, $f'(v)$ is the hub in H^* assigned to v in the solution returned by the algorithm. Let ℓ'_1 and ℓ'_2 be the largest value and second-largest value in the multiset $\{w(v, f'(v)) \mid v \in V \setminus \{c\}\}$. By inequality (1), we have $\ell'_1 + \ell'_2 \le 2\beta(\ell_1 + \ell_2)$.

Let $x, y \in V \setminus \{c\}$ be the nodes achieving the maximum path length in T, *i.e.*, $d_T(x,y) = D(T)$. It suffices to show that $D(T) \le (2\beta + 1) \cdot D(T^*)$.

If $f'(x) = f'(y)$, then $D(T) = w(x, f'(x)) + w(y, f'(y)) \le \ell'_1 + \ell'_2$.

If $f'(x) \ne f'(y)$, then

$$D(T) = w(x, f'(x)) + w(f'(x), c) + w(f'(y), c) + w(y, f'(y)) \le \ell'_1 + 2\ell_0 + \ell'_2$$

where we use $w(h,c) \le \ell_0$ for all $h \in H$ by our choice of H. Combine with the fact that $D(T^*) \ge \ell_0 + \ell_1$ and $D(T^*) \ge \ell_1 + \ell_2$, we always have

$$\begin{aligned} D(T) &\le 2\ell_0 + \ell'_1 + \ell'_2 \\ &\le 2\ell_0 + 2\beta(\ell_1 + \ell_2) \\ &\le 2(\ell_0 + \ell_1) + (2\beta - 1)(\ell_1 + \ell_2) \quad (\text{using } \ \ell_2 \le \ell_1) \\ &\le 2 \cdot D(T^*) + (2\beta - 1) \cdot D(T^*) \\ &= (2\beta + 1) \cdot D(T^*), \end{aligned}$$

which indicates that Algorithm 2 is a $(2\beta + 1)$-approximation algorithm for Δ_β-SpHCP. This completes the proof. □

We close this section with the following theorem.

Theorem 2. *Let $\beta \ge \frac{1}{2}$. There exists a polynomial time $r(\beta)$-approximation algorithm for Δ_β-SpHCP where*

(i) $r(\beta) = 1$ *if* $\beta \le \frac{3-\sqrt{3}}{2}$;
(ii) $r(\beta) = \frac{1+2\beta-2\beta^2}{4(1-\beta)}$ *if* $\frac{3-\sqrt{3}}{2} < \beta \le 0.7737\ldots$;
(iii) $r(\beta) = 1 + \frac{4\beta^2}{5\beta+1}$ *if* $0.7737\ldots \le \beta \le 1$;
(iv) $r(\beta) = \beta + \frac{4\beta^2-2\beta}{2+\beta}$ *if* $1 \le \beta \le 2$;
(v) $r(\beta) = 2\beta + 1$ *if* $\beta \ge 2$.

3 Conclusion

In this paper, we have studied Δ_β-SpHCP for all $\beta \ge \frac{1}{2}$. We showed that for any $\varepsilon > 0$, to approximate Δ_β-SpHCP to a ratio $g(\beta) - \varepsilon$ is NP-hard where

$g(\beta) = \frac{1+2\beta-2\beta^2}{4(1-\beta)}$ if $\frac{3-\sqrt{3}}{2} < \beta \le \frac{2}{3}$; $g(\beta) = \frac{5\beta+1}{4}$ if $\frac{2}{3} < \beta \le 1$; $g(\beta) = \beta + \frac{1}{2}$ if $\beta \ge 1$. Moreover, we gave $r(\beta)$-approximation algorithms for the same problem. If $\beta \le \frac{3-\sqrt{3}}{2}$, we have $r(\beta) = g(\beta) = 1$, *i.e.*, Δ_β-SpHCP is polynomial time solvable for $\beta \le \frac{3-\sqrt{3}}{2}$. If $\frac{3-\sqrt{3}}{2} < \beta \le \frac{2}{3}$, we have $r(\beta) = g(\beta) = \frac{1+2\beta-2\beta^2}{4(1-\beta)}$. For $\frac{2}{3} \le \beta \le 1$, $r(\beta) = \min\{\frac{1+2\beta-2\beta^2}{4(1-\beta)}, 1 + \frac{4\beta^2}{5\beta+1}\}$. For $\beta \ge 1$, we have $r(\beta) = \min\{\beta + \frac{4\beta^2-2\beta}{2+\beta}, 2\beta+1\}$. In the future work, it is of interest to extend the range of β for Δ_β-SpHCP such that the gap between the upper and lower bounds of approximability can be reduced.

References

1. Alumur, S.A., Kara, B.Y.: Network hub location problems: the state of the art. Eur. J. Oper. Res. **190**, 1–21 (2008)
2. Andreae, T.: On the traveling salesman problem restricted to inputs satisfying a relaxed triangle inequality. Networks **38**, 59–67 (2001)
3. Andreae, T., Bandelt, H.-J.: Performance guarantees for approximation algorithms depending on parameterized triangle inequalities. SIAM J. Discret. Math. **8**, 1–16 (1995)
4. Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccamela, A., Protasi, M.: Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties. Springer, Heidelberg (1999)
5. Bender, M.A., Chekuri, C.: Performance guarantees for the TSP with a parameterized triangle inequality. Inf. Process. Lett. **73**, 17–21 (2000)
6. Böckenhauer, H.-J., Bongartz, D., Hromkovič, J., Klasing, R., Proietti, G., Seibert, S., Unger, W.: On the hardness of constructing minimal 2-connected spanning subgraphs in complete graphs with sharpened triangle inequality. In: Agrawal, M., Seth, A. (eds.) FSTTCS 2002. LNCS, vol. 2556, pp. 59–70. Springer, Heidelberg (2002). doi:10.1007/3-540-36206-1_7
7. Böckenhauer, H.-J., Bongartz, D., Hromkovič, J., Klasing, R., Proietti, G., Seibert, S., Unger, W.: On k-edge-connectivity problems with sharpened triangle inequality. In: Petreschi, R., Persiano, G., Silvestri, R. (eds.) CIAC 2003. LNCS, vol. 2653, pp. 189–200. Springer, Heidelberg (2003). doi:10.1007/3-540-44849-7_24
8. Böckenhauer, H.-J., Bongartz, D., Hromkovič, J., Klasing, R., Proietti, G., Seibert, S., Unger, W.: On k-connectivity problems with sharpened triangle inequality. J. Discret. Algorithms **6**(4), 605–617 (2008)
9. Böckenhauer, H.-J., Hromkovič, J., Klasing, R., Seibert, S., Unger, W.: Approximation algorithms for the TSP with sharpened triangle inequality. Inf. Process. Lett. **75**, 133–138 (2000)
10. Böckenhauer, H.-J., Hromkovič, J., Klasing, R., Seibert, S., Unger, W.: Towards the notion of stability of approximation for hard optimization tasks and the traveling salesman problem. In: Bongiovanni, G., Petreschi, R., Gambosi, G. (eds.) CIAC 2000. LNCS, vol. 1767, pp. 72–86. Springer, Heidelberg (2000). doi:10.1007/3-540-46521-9_7
11. Böckenhauer, H.-J., Hromkovič, J., Klasing, R., Seibert, S., Unger, W.: An improved lower bound on the approximability of metric TSP and approximation algorithms for the TSP with sharpened triangle inequality. In: Reichel, H., Tison, S. (eds.) STACS 2000. LNCS, vol. 1770, pp. 382–394. Springer, Heidelberg (2000). doi:10.1007/3-540-46541-3_32

12. Böckenhauer, H.-J., Hromkovič, J., Seibert, S.: Stability of approximation. In: Gonzalez, T.F. (ed.) Handbook of Approximation Algorithms and Metaheuristics. Chapman & Hall/CRC, Boca Raton (2007). Chapter 31
13. Böckenhauer, H.-J., Seibert, S.: Improved lower bounds on the approximability of the traveling salesman problem. RAIRO - Theor. Inform. Appl. **34**, 213–255 (2000)
14. Campbell, J.F.: Integer programming formulations of discrete hub location problems. Eur. J. Oper. Res. **72**, 387–405 (1994)
15. Campbell, J.F., Ernst, A.T.: Hub location problems. In: Drezner, Z., Hamacher, H.W. (eds.) Facility Location: Applications and Theory, pp. 373–407. Springer, Berlin (2002)
16. Chen, L.-H., Cheng, D.-W., Hsieh, S.-Y., Hung, L.-J., Lee, C.-W., Wu, B.Y.: Approximation algorithms for single allocation k-hub center problem. In: Proceedings of the 33rd Workshop on Combinatorial Mathematics and Computation Theory (CMCT 2016), pp. 13–18 (2016)
17. Chen, L.-H., Cheng, D.-W., Hsieh, S.-Y., Hung, L.-J., Lee, C.-W., Wu, B.Y.: Approximation algorithms for the star k-hub center problem in metric graphs. In: Dinh, T.N., Thai, M.T. (eds.) COCOON 2016. LNCS, vol. 9797, pp. 222–234. Springer, Cham (2016). doi:10.1007/978-3-319-42634-1_18
18. Ernst, A.T., Hamacher, H., Jiang, H., Krishnamoorthy, M., Woeginger, G.: Uncapacitated single and multiple allocation p-hub center problem. Comput. Oper. Res. **36**, 2230–2241 (2009)
19. Gonzalez, T.F.: Clustering to minimize the maximum intercluster distance. Theor. Comput. Sci. **38**, 293–306 (1985)
20. Hochbaum, D.S. (ed.): Approximation Algorithms for NP-hard Problems. PWS Publishing Company, Pacific Grove (1996)
21. Hochbaum, D.S., Shmoys, D.B.: A unified approach to approximation algorithms for bottleneck problems. J. ACM **33**, 533–550 (1986)
22. Hromkovič, J.: Stability of approximation algorithms for hard optimization problems. In: Pavelka, J., Tel, G., Bartošek, M. (eds.) SOFSEM 1999. LNCS, vol. 1725, pp. 29–47. Springer, Heidelberg (1999). doi:10.1007/3-540-47849-3_2
23. Kara, B.Y., Tansel, B.Ç.: On the single-assignment p-hub center problem. Eur. J. Oper. Res. **125**, 648–655 (2000)
24. Liang, H.: The hardness and approximation of the star p-hub center problem. Oper. Res. Lett. **41**, 138–141 (2013)
25. Meyer, T., Ernst, A., Krishnamoorthy, M.: A 2-phase algorithm for solving the single allocation p-hub center problem. Comput. Oper. Res. **36**, 3143–3151 (2009)
26. O'Kelly, M.E., Miller, H.J.: Solution strategies for the single facility minimax hub location problem. Pap. Reg. Sci. **70**, 376–380 (1991)
27. Todosijević, R., Urošević, D., Mladenović, N., Hanafi, S.: A general variable neighborhood search for solving the uncapacitated r-allocation p-hub median problem. Optimization Letters **1**, 1–13 (2015). doi:10.1007/s11590-015-0867-6
28. Yaman, H., Elloumi, S.: Star p-hub center problem and star p-hub median problem with bounded path lengths. Comput. Oper. Res. **39**, 2725–2732 (2012)