Theoretical Computer Science 411 (2010) 1038–1044

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

The pos/neg-weighted 1-median problem on tree graphs with subtree-shaped customers[☆]

Yukun Cheng [a,b], Liying Kang [a,*], Changhong Lu [c]

[a] Department of Mathematics, Shanghai University, Shanghai 200444, PR China

[b] School of Mathematics and Statistics, Zhejiang University of Finance and Economics, Hangzhou 310018, PR China

[c] Department of Mathematics, East China Normal University, Shanghai, 200241, PR China

ARTICLE INFO

Article history:
Received 30 April 2009
Received in revised form 27 October 2009
Accepted 7 November 2009
Communicated by D.-Z. Du

Keywords:
Location theory
Median problem
Subtree-shaped customers
Tree graphs

ABSTRACT

In this paper we consider the pos/neg-weighted median problem on a tree graph where the customers are modeled as continua subtrees. We address the discrete and continuous models, i.e., the subtrees' boundary points are all vertices, or possibly inner points of an edge, respectively. We consider two different objective functions. If we minimize the overall *sum of the minimum weighted distances* of the subtrees from the facilities, there exists an optimal solution satisfying a generalized vertex optimality property, e.g., there is an optimal solution such that all facilities are located at vertices or the boundary points of the subtrees. Based on this property we devise a polynomial time algorithm for the pos/neg-weighted 1-median problem on a tree with subtree-shaped customers.

1. Introduction

In location theory the p-median problem plays an important role. It can be stated as follows: Let $G = (V, E)$ be an undirected graph with a length function $l : E \to R_+$ and nonnegative vertex weights. We want to locate p facilities on edges or vertices of G so as to minimize the overall sum of the weighted distances of the vertices to the respective closest facility. The problem possesses the so-called vertex optimality property (see for example Mirchandani [9]) which asserts that there exists an optimal solution such that all facilities are located at vertices of Kariv and Hakimi [8] showed that the classical p-median problem in graphs is NP-hard. For the case where the underlying graph is a tree, these authors designed an $O(p^2n^2)$ algorithm. Then Tamir [13] improved the time complexity of the p-median problem on trees to $O(pn^2)$. In 2005, Benkoczi and Bhattacharya [2] designed an $O(n \log^{p+2} n)$ time algorithm to solve the p median problem on trees.

If obnoxious facilities are considered by allowing the negative vertex weights, i.e., facilities which should be desirably located far away from the clients, one obtains the so-called obnoxious facility location problems. An obnoxious facility could be, e.g., a nuclear plant, a military depot or a garbage dump. A more general problem, namely the semi-obnoxious case where positive and negative weights are allowed, was considered by Burkard and Krarup [6]. They developed a linear time algorithm for the 1-median problem on a cactus.

Burkard, Çela and Dollani [4] considered the 2-median problem on trees for the semi-obnoxious case and called this kind of location problem the pos/neg facility location problem. They proposed two different objective functions. One tries to minimize the sum over all vertices of the minimal weight times the distance (MWD), whereas the second one minimizes

☆ Research was partially supported by the NNSF of China (No. 10971131), the ShuGuang Plan of Shanghai Education Development Foundation (No. 06SG42) and Key Disciplines of Shanghai Municipality (S30104).

* Corresponding author. Tel.: +86 21 66135652.
E-mail address: lykang@shu.edu.cn (L. Kang).

0304-3975/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2009.11.009

the sum over all vertices v of the weighted minimum distance from v to the facilities (WMD). In particular, they show that in general networks with pos/neg weights the vertex optimality does not hold. In the case of trees they showed that this property holds for model (MWD). Based on this observation they developed in the case $p = 2$ an $O(n^2)$ algorithm for trees. For the (WMD) problem they found an algorithm that runs in $O(n^3)$ time on a tree. These results were recently improved by Benkoczi [1] and Benkoczi, Breton and Bhattacharya [3]. Using the spine tree decomposition they showed that the 2-median problem on a tree with positive and negative vertex weights can be solved in $O(n \log n)$ time for the MWD model. For the WMD model they developed an $O(nh \log^2 n)$ time algorithm where h represents the height of the tree. Burkard and Fathali [5] gave an $O(n^3)$ algorithm for the 3-median problem on trees with positive and negative weights in the MWD model.

In the last decade there has been a great interest in locating "extensive" servers (new facilities), like paths or subtrees of a network, which cannot be represented as points of the network or points in a continuous space (see, for example, the surveys by Plastria [11] and Díaz-Báñez et al. [7]). In this paper we study and focus on location problems with extensive customers (see the paper by Nickel et al. [10], Puerto et al., [12] and the references therein). For motivation purposes consider a network with communication lines, where each line is used exclusively to connect a pair of points. To check whether a line properly works it is sufficient to reach any point on that line, and transmit to each one of that line's endpoints. From a given point the server can test the functionality of all lines passing through that point. In this model each communication line is viewed as an extensive customer on the network and the servers are the checkpoints that we need to choose on the network so that we cover all the extensive customers (lines). The optimization problem is to find the minimum number of checkpoints needed to cover all lines.

Motivated by the above example of extensive customers, we consider the median problems on a tree graph where the demand originated at continua (extensive) connected facilities: subtrees. That is, we consider a finite collection of subsets $S_i, i \in I = \{1, 2, \dots, m\}$, where $S_i \subseteq V$. S_i is viewed as a set of demand points. Let $n = |V|$. When a demand is originated at some subset S_i, a server located at x must travel and visit each of the customers in S_i and return to its home base. The total travel distance of the server will be $2d(x, T_i) + 2L(T_i)$, where T_i is the subtree induced by S_i, $d(x, T_i)$ is the distance between x and its closest point in T_i, and $L(T_i)$ is the sum of the edge-lengths of T_i. The transportation cost of S_i is assumed to be a linear function of $2d(x, T_i) + 2L(T_i)$. Specifically, for $i \in I$, let w_i and k_i be a pair of reals. Then the transportation cost is $w_i(d(x, T_i) + k_i)$, where the terms k_i and w_i are referred to as the addend and weight of the extensive customer T_i, respectively. Since $\sum_{i=1}^{m} w_i k_i$ is a constant, our optimization problem is related only to $\sum_{i=1}^{m} w_i d(x, T_i)$. Due to the form of the distance function which depends on T_i we refer to the extensive T_i customer instead of S_i customer.

For more general case, the weight of extensive customer T_i can be positive or negative. The pos/neg-weighted p-median problem of the above extensive customer model is to locate p points (servers) on the tree network, minimizing the sum over all extensive customers of the minimal weight times the distance from T_i to the facilities (MWD) or minimizing the sum over all extensive customers T_i of the weighted minimum distance from T_i to the facilities (WMD). By the definitions of MWD and WMD models, it is easy to get that the general median problem is equivalent to this problem without addends. Thus in this paper we only consider the two models without addends.

In this paper, we shall propose a polynomial time algorithm for finding the one median on a tree with subtree-shaped customers. The rest of the paper is organized as follows. First, we introduce some notations, definitions and preliminaries which will be used throughout the paper. In Section 3, we first consider the discrete case and propose a property of this case for the MWD model. Based on this property, we devise a polynomial time algorithm for the pos/neg-weighted 1-median problem. Finally, we consider the continuous case by constructing a modified tree.

2. Notation, definitions and preliminaries

Let $T = (V, E)$ be an undirected tree with vertex set $V(T) = \{v_1, v_2, \dots, v_n\}$. Each edge (v_i, v_j) has a nonnegative length $l(v_i, v_j)$. In particular, edge (v_i, v_j) is identified as a closed interval $[0, l(v_i, v_j)]$ so that we can refer to its interior points. Let $A(T)$ be the continuum set of points on the edges of T. Consequently, a point x in $A(T)$ is either a vertex $v \in V(T)$ or an interior point of some edge $(v_i, v_j) \in E$. For any point x on edge (v_i, v_j), we define its distance $d(x, v_i) = t$ and its distance $d(x, v_j) = l(v_i, v_j) - t$ where $0 \le t \le l(v_i, v_j)$. Let $P(x, y)$ be the unique path connecting the points x and y in $A(T)$. Distance $d(x, y)$ denotes the length of path $P(x, y)$. For any pair of closed compact subsets $X, Y \subseteq A(T)$, $d(X, Y) = \min\{d(x, y) | x \in X, y \in Y\}$. If $d(X, Y) > 0$, then there exists a point $x' \in X$ and a point $y' \in Y$ such that $d(X, Y) = d(x', y')$. Here, we define path $P(x', y')$ just is the path between subsets X and Y, i.e., $P(X, Y) = P(x', y')$. If $d(X, Y) = 0$, then $X \cap Y \neq \emptyset$. Specially if $X = \{x\}$, then $d(X, Y) = d(x, Y) = \min\{d(x, y) | y \in Y\}$. A subtree is a compact and connected subset of $A(T)$. A subtree is discrete if its relative boundary points are all vertices. Otherwise we call it continuous.

We suppose that T is a rooted tree with root v_1. The level number $level(v_i)$ of vertex v_i, $i = 1, 2, \dots, n$, is the number of vertices on path $P(v_i, v_1)$. For a given rooted tree T rooted at v_1 we number the vertices in the following way: if $level(v_i) < level(v_j)$, then $i < j$; if $level(v_i) = level(v_j)$ and v_i lies on the left of v_j, then $i < j$ (see Fig. 1). If vertex v_i is adjacent to v_j and $level(v_j) = level(v_i) + 1$, then v_i is called the *parent* of v_j and v_j is called a *son* of v_i. Vertex v_i is a leaf if it has no sons. For any vertex v_j, $v_{par(j)}$ denotes the parent of vertex v_j and $Son(v_i)$ denotes the set of v_i's sons. A vertex v_h is a *descendant* of v_i (and v_i is an ancestor of v_h) if v_i is on the path $P(v_h, v_1)$. If we delete edge (v_i, v_j) with $v_i = v_{par(j)}$, the rooted tree T will be decomposed into two subtrees. We denote one subtree containing v_j by T_{v_j} and the other one containing

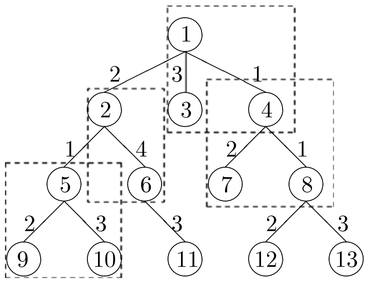

Fig. 1. A rooted tree T and the subtrees labeled by the dashboxes.

vertices v_i and v_1 by $T^c_{v_j}$. Obviously, vertex set $V(T_{v_j})$ is the set of v_j's descendants and T_{v_j} is rooted at v_j. An example is shown in Fig. 1, where the number in each cycle is the index of the vertex and the number on each edge is the weight of the edge.

In this paper we consider the following pos/neg-weighted median location problems. Given a collection of m subtrees $\mathcal{T} = \{T_i\}, i \in I = \{1, 2, \ldots, m\}$. Each subtree $T_i,\ i \in I$, has a weight w_i which can be positive or negative. For each subtree T_i, $i \in I$, we denote z_i to be the closest vertex in T_i to v_1, the root of tree T. Similarly, we call z_i the root of subtree T_i. A subtree is discrete if its relative boundary points are vertices. Assume that p facilities are located at p points in $A(T)$. We identify each facility with its location and denote by $X_p = \{x_1, x_2, \ldots, x_p\}$ the set of p facilities. Similar to the models in [4], we present two different objective functions as follows:

The sum of the *minimum weighted distances* (MWD) between the subtrees and the facility set X_p:

$$F_{\mathrm{MWD}}(X_p) = \sum_{i=1}^{m} \min\nolimits_{1\le j\le p}(w_i d(x_j, T_i)). \tag{1}$$

The MWD p-median problem is to find a set X^* of p points in $A(T)$ such that

$$F_{\mathrm{MWD}}(X^*) = \min\nolimits_{X_p\subseteq A(T)} F_{\mathrm{MWD}}(X_p).$$

And the sum of the *weighted minimum distances* (WMD) between the subtrees and the facility set X_p:

$$F_{\mathrm{WMD}}(X_p) = \sum_{i=1}^{m} w_i\ (\min\nolimits_{1\le j\le p} d(x_j, T_i)). \tag{2}$$

The WMD p-median problem is to find a set X^* of p points in $A(T)$ such that

$$F_{\mathrm{WMD}}(X^*) = \min\nolimits_{X_p\subseteq A(T)} F_{\mathrm{WMD}}(X_p).$$

Note that if $p = 1$ or all weights $w_i,\ i \in I$, are positive, $F_{\mathrm{MWD}}(X_p) = F_{\mathrm{WMD}}(X_p)$ for all $X_p \subseteq A(T)$, i.e., MWD p-median problem is equivalent to WMD p-median problem.

3. The pos/neg-weighted median problem

In this section we first discuss the median problem if the subtrees are all discrete. For each subtree T_i, $i \in I = \{1, 2, \ldots, m\}$, let $E(T_i)$, $V(T_i)$ and L^*_i be the edge set, the vertex set and the boundary vertex set of T_i, respectively, and let $|E(T_i)| = \varepsilon_i$, $|L^*_i| = n^*_i$. A good property in [4] of the MWD p-median problem on a tree if all customers are located at vertices is the vertex optimality property: there exists an optimal solution which locates all facilities at vertices in $V(T)$. Similarly, if we consider the MWD p-median problem on a tree with all discrete subtree-shaped customers, vertex optimality property holds:

Lemma 1. *The vertex optimality property holds for the* MWD *p-median problem on a tree with all discrete subtree-shaped customers.*

Proof. Let $X^* = \{x_1, x_2, \ldots, x_p\}$ be an optimal solution of the MWD p-median problem containing the smallest number of non-vertices. If all x_i are vertices, then we are done. Otherwise we suppose that there exists an interior point x_k on edge (v_r, v_s) with $d(v_r, x_k) = t_0, (0 < t_0 < l(v_r, v_s))$. Define the collections

$$\begin{aligned}
\mathcal{T}_k &= \{T_i \in \mathcal{T} \mid w_i d(T_i, x_k) = \min\nolimits_{1\le j\le p} w_i d(T_i, x_j)\},\\
\mathcal{T}_{r,k} &= \{T_i \in \mathcal{T}_k \mid d(T_i, x_k) > 0 \text{ and } v_r \in V(P(T_i, x_k))\},\\
\mathcal{T}_{s,k} &= \{T_i \in \mathcal{T}_k \mid d(T_i, x_k) > 0 \text{ and } v_s \in V(P(T_i, x_k))\},\\
\mathcal{T}_{0,k} &= \{T_i \in \mathcal{T}_k \mid d(T_i, x_k) = 0\}.
\end{aligned}$$

Note that the collection $\mathcal{T}_k$ contains all subtrees that are served by facility x_k. Since T_i, $i \in I$, is discrete, there is a vertex $u \in V(T_i)$ such that $P(T_i, x_k) = P(u, x_k)$ and $d(T_i, x_k) = d(u, x_k)$ if $d(T_i, x_k) > 0$ and edge $(v_r, v_s) \in E(T_i)$ if $d(T_i, x_k) = 0$.

In the following we discuss a function $f : [0, l(v_r, v_s)] \to R$ with an argument t, where $t = 0$ corresponds to vertex v_r and $t = l(v_r, v_s)$ corresponds to v_s. Moreover, the interior point y on edge (v_r, v_s) with $d(v_r, y)$ corresponds to $t = d(v_r, y)$ and $0 < t < l(v_r, v_s)$. Clearly, for any $T_i \in \mathcal{T}_{r,k}$, $d(T_i, y) > 0$ and $v_r \in V(P(T_i, y))$; For any $T_i \in \mathcal{T}_{s,k}$, $d(T_i, y) > 0$ and $v_s \in V(P(T_i, y))$ and for any $T_i \in \mathcal{T}_{0,k}$, $d(T_i, y) = 0$. Let f be the function given by

$$\begin{aligned} f(t) &= \sum_{T_i \in \mathcal{T}_{r,k}} w_i(d(T_i, v_r) + t) + \sum_{T_i \in \mathcal{T}_{s,k}} w_i(d(T_i, v_s) + l(v_r, v_s) - t) \\ &= \left(\sum_{T_i \in \mathcal{T}_{r,k}} w_i - \sum_{T_i \in \mathcal{T}_{s,k}} w_i \right) t + \sum_{T_i \in \mathcal{T}_{r,k}} w_i d(T_i, v_r) + \sum_{T_i \in \mathcal{T}_{s,k}} w_i(d(T_i, v_s) + l(v_r, v_s)). \end{aligned}$$

Because $d(T_i, y) = 0$ for any $T_i \in \mathcal{T}_{0,k}$, T_i dose not contribute to the function $f(t)$. Obviously, $f(t)$ is a monotone linear function and $\min_{0 \le t \le l(v_r, v_s)} f(t) = \min\{f(0), f(l(v_r, v_s))\}$. Without loss of generality, we suppose that $f(t)$ achieves its minimum at v_r, i.e., $t = 0$. Now we set $\widetilde{X} = (X^* \setminus \{x_k\}) \cup \{v_r\}$ and

$$\begin{aligned} F_{\mathrm{MWD}}(X^*) &= f(t_0) + \sum_{T_i \in \mathcal{T} \setminus \mathcal{T}_k} \min_{1 \le j \le p, j \ne k} w_i d(T_i, x_j) \\ &\ge f(0) + \sum_{T_i \in \mathcal{T} \setminus \mathcal{T}_k} \min_{1 \le j \le p, j \ne k} w_i d(T_i, x_j) \\ &\ge F_{\mathrm{MWD}}(\widetilde{X}). \end{aligned}$$

Note that $\widetilde{X}$ is also an optimal solution with strictly less non-vertices than X^*. It contradicts our assumption. □

For WMD p-median problem it is easy to get that the median problem on a tree if all customers are located at vertices is a special case of the median problem on a tree with all discrete subtree-shaped customers. Unfortunately from [4] we know that the vertex optimality property dose not hold for the former problem. Thus the WMD p-median problem on a tree with all discrete subtree-shaped customers dose not possess this nice property.

Since each subtree is discrete, the root z_i of subtree T_i is a vertex in $V(T)$. By numbering the vertices stated in Section 2, we can get the following observation easily.

Observation 2. *The subtree T_i's root z_i is the vertex with the smallest index in $V(T_i)$.*

For any edge e and vertex v, let the subcollection $\mathcal{T}(e) = \{T_i \in \mathcal{T} \mid e \in E(T_i)\}$ and $\mathcal{T}(v) = \{T_i \in \mathcal{T} \mid z_i = v\}$. Clearly for any subtree $T_i \in \mathcal{T}(e)$, T_i has edge e and for any subtree $T_i \in \mathcal{T}(v)$, T_i's root is v. We denote the sum of weight of $T_i \in \mathcal{T}(e)$ and $T_i \in \mathcal{T}(v)$ by $W(e)$ and $W(v)$ respectively, i.e., $W(e) = \sum_{T_i \in \mathcal{T}(e)} w_i$ and $W(v) = \sum_{T_i \in \mathcal{T}(v)} w_i$.

Observation 3. *For any vertex v_j subtree $T_i \subseteq T_{v_j}$ if and only if $z_i \in V(T_{v_j})$.*

Note that for any vertex v_j and any subtree T_i if $T_i \subseteq T_{v_j}$, then $d(T_i, v_j) = d(z_i, v_j)$. Let $D(v_j)$ be the sum of weight of subtree $T_i \subseteq T_{v_j}$, i.e., $D(v_j) = \sum_{T_i \subseteq T_{v_j}} w_i$. For convenience, we denote $F(T', v_j) = \sum_{T_i \subseteq T'} w_i d(T_i, v_j)$, where T' is a subtree of T. In order to solve the pos/neg-weighted 1-median problem, we should find a vertex v^* such that $F(T, v^*) = min_{v_j \in V(T)} F(T, v_j)$. Next we propose a polynomial time algorithm for finding the medians.

Algorithm A.
Input: A given weighted tree T rooted at v_1, $\{T_i\}_{i \in I}$ with $E(T_i)$, $V(T_i)$ and L_i^*
Output: Find the medians of T.
Step 1: For every edge $e \in E(T)$, compute $W(e)$;
Step 2: For every vertex $v \in V(T)$, compute $W(v)$;
Step 3: Compute $D(v_j) = \sum_{u \in Son(v_j)} D(u) + W(v_j)$ in a bottom-up fashion;
Step 4: Compute $F(T_{v_j}, v_j) = \sum_{u \in Son(v_j)} (F(T_u, u) + D(u) d(u, v_j))$ in a bottom-up fashion;
Step 5: compute $F(T, v_j) = F(T, v_{par(j)}) + [D(v_1) - 2D(v_j) - W(v_j, v_{par(j)})] d(v_j, v_{par(j)})$ in a top-down fashion;
Step 6: Find the vertices with the smallest $F(T, v_j)$ which are the medians.

Now we shall explain the correctness of Algorithm A.

Lemma 4. *For any vertex $v_j \in V(T)$, $D(v_j) = \sum_{u \in Son(v_j)} D(u) + W(v_j)$.*

Proof. If v_j is a leaf, then $Son(v_j) = \emptyset$ and $V(T_{v_j}) = \{v_j\}$. So for any subtree $T_i \subseteq T_{v_j}$, we can get $V(T_i) = \{v_j\}$ and $z_i = v_j$. So $D(v_j) = \sum_{T_i \subseteq T_{v_j}} w_i = \sum_{z_i = v_j} w_i = W(v_j)$.

If v_j is not a leaf, we suppose that this claim holds for any vertex $u \in Son(v_j)$. For any subtree $T_i \subseteq T_{v_j}$, then $z_i = v_j$ or there is a vertex $u \in Son(v_j)$ such that $T_i \subseteq T_u$. So we can get

$$\begin{aligned} D(v_j) &= \sum_{T_i \subseteq T_{v_j}} w_i = \sum_{u \in Son(v_j)} \sum_{T_i \subseteq T_u} w_i + \sum_{z_i = v_j} w_i \\ &= \sum_{u \in Son(v_j)} D(u) + W(v_j). \end{aligned}$$

This completes the proof. □

Lemma 5. *For any vertex $v_j \in V(T)$ and T_{v_j}, $F(T_{v_j}, v_j) = \sum_{u \in Son(v_j)} (F(V_u, u) + D(u) l(u, v_j))$.*

Proof. If v_j is a leaf, then we get $Son(v_j) = \emptyset$. For any subtree $T_i \subseteq T_{v_j}$, $V(T_i) = \{v_j\}$. So $F(T_{v_j}, v_j) = \sum_{T_i \subseteq T_{v_j}} w_i d(T_i, v_j) = 0$.

If v_j is not a leaf, we suppose this claim holds for any vertex $u \in Son(v_j)$. So for any subtree $T_i \subseteq T_{v_j}$, there are two cases:
Case 1. There is a vertex $u \in Son(v_j)$ such that $T_i \subseteq T_u$. Clearly, $d(T_i, v_j) = d(z_i, v_j) = d(z_i, u) + l(u, v_j)$.
Case 2. $z_i = v_j$. So $d(T_i, v_j) = d(z_i, v_j) = 0$.
Thus we can get

$$\begin{aligned} F(T_{v_j}, v_j) &= \sum_{u \in Son(v_j)} \sum_{T_i \subseteq T_u} w_i d(T_i, v_j) + \sum_{z_i = v_j} w_i d(T_i, v_j) \\ &= \sum_{u \in Son(v_j)} \sum_{T_i \subseteq T_u} w_i (d(z_i, u) + l(u, v_j)) \\ &= \sum_{u \in Son(v_j)} \sum_{T_i \subseteq T_u} w_i d(z_i, u) + \sum_{u \in Son(v_j)} \left(\sum_{T_i \subseteq T_u} w_i \right) l(u, v_j) \\ &= \sum_{u \in Son(v_j)} (F(T_u, u) + D(u) l(u, v_j)). \quad \square \end{aligned}$$

Lemma 6. *For any vertex $v_j \in V(T)$,*

$$F(T, v_j) = F(T, v_{par(j)}) + [D(v_1) - 2D(v_j) - W(v_j, v_{par(j)})] l(v_j, v_{par(j)}).$$

Proof. According to the relation between vertex v_j and subtree $T_i \in \mathcal{T}$, we can decompose $\mathcal{T}$ into three disjoint subcollections:

$$\begin{aligned} \mathcal{T}^1(v_j) &= \{T_i \in \mathcal{T} \mid T_i \subseteq T_{v_j}\}; \\ \mathcal{T}^2(v_j) &= \{T_i \in \mathcal{T} \mid T_i \subseteq T^c_{v_j}\}; \\ \mathcal{T}^3(v_j) &= \{T_i \in \mathcal{T} \mid edge\ (v_j, v_{par(j)}) \in E(T_i)\} = \mathcal{T}(v_j, v_{par(j)}). \end{aligned}$$

For any $T_i \in \mathcal{T}^1(v_j)$, we get that path $P(T_i, v_{par(j)})$ passes through vertex v_j. So

$$d(T_i, v_{par(j)}) = d(T_i, v_j) + l(v_j, v_{par(j)}). \tag{3}$$

For any $T_i \in \mathcal{T}^2(v_j)$, path $P(T_i, v_j)$ passes through vertex $v_{par(j)}$. So

$$d(T_i, v_j) = d(T_i, v_{par(j)}) + l(v_j, v_{par(j)}). \tag{4}$$

For any $T_i \in \mathcal{T}^3(v_j)$, edge $(v_j, v_{par(j)}) \in E(T_i)$. So

$$d(T_i, v_j) = d(T_i, v_{par(j)}) = 0. \tag{5}$$

From (3)–(5), we can get

$$\begin{aligned} F(T, v_j) &= \sum_{T_i \in \mathcal{T}} w_i d(T_i, v_j) \\ &= \sum_{k=1}^{3} \sum_{T_i \in \mathcal{T}^k(v_j)} w_i d(T_i, v_j) \\ &= \sum_{T_i \in \mathcal{T}^1(v_j)} w_i (d(T_i, v_{par(j)}) - l(v_j, v_{par(j)})) + \sum_{T_i \in \mathcal{T}^2(v_j)} w_i (d(T_i, v_{par(j)}) + l(v_j, v_{par(j)})) + \sum_{T_i \in \mathcal{T}^3(v_j)} w_i \cdot 0 \\ &= \sum_{T_i \in \mathcal{T}} w_i d(T_i, v_{par(j)}) - \left(\sum_{T_i \in \mathcal{T}^1(v_j)} w_i - \sum_{T_i \in \mathcal{T}^2(v_j)} w_i \right) l(v_j, v_{par(j)}). \end{aligned} \tag{6}$$

edge	(v_1, v_2)	(v_1, v_3)	(v_1, v_4)	(v_2, v_5)	(v_2, v_6)	(v_4, v_7)
$W(e_i)$	0	2	2	0	-3	-2
edge	(v_4, v_8)	(v_5, v_9)	(v_5, v_{10})	(v_6, v_{11})	(v_8, v_{12})	(v_8, v_{13})
$W(e_i)$	-2	4	4	0	0	0

Fig. 2. The value of $W(e)$ for every edge $e \in E(T)$.

vertex	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}
$W(v_i)$	2	-3	0	-2	4	0	0	0	0	0	0	0	0
$D(v_i)$	1	1	0	-2	4	0	0	0	0	0	0	0	0
$F(T_{v_i}, v_i)$	4	4	0	0	0	0	0	0	0	0	0	0	0
$F(T, v_i)$	4	2	1	7	-5	18	13	10	-11	-14	21	12	13

Fig. 3. The computation of $W(v_i)$, $D(v_i)$, $F(T_{v_i}, v_i)$ and $F(T, v_i)$.

From the definitions of $\mathcal{T}^1(v_j)$ and $\mathcal{T}^3(v_j)$,

$$\sum_{T_i \in \mathcal{T}^1(v_j)} w_i = \sum_{T_i \subseteq T_{v_j}} w_i = D(v_j),$$

$$\sum_{T_i \in \mathcal{T}^3(v_j)} w_i = \sum_{T_i \in \mathcal{T}(v_j, v_{par(j)})} w_i = W(v_j, v_{par(j)}).$$

Since $D(v_1) = \sum_{T_i \in \mathcal{T}} w_i = D(v_j) + \sum_{T_i \in \mathcal{T}^2(v_j)} w_i + W(v_j, v_{par(j)})$, we get

$$\sum_{T_i \in \mathcal{T}^2(v_j)} w_i = D(v_1) - D(v_j) - W(v_j, v_{par(j)}).$$

As an immediate consequence, this claim holds. $\square$

From the definition of $\mathcal{T}(v)$, it is easy to get that $\mathcal{T}(v_i) \cap \mathcal{T}(v_j) = \emptyset$, $(v_i \neq v_j)$, and $\bigcup_{v_j \in V} \mathcal{T}(v_j) = \mathcal{T}$. Thus we can traverse each vertex v and get $\{W(v)\}_{v \in V}$ in $O(m+n)$ time. Since there are ε_i edges in T_i, subtree T_i shall be computed ε_i times when we compute $\{W(e)\}_{e \in E}$. So we can traverse each edge e and get $\{W(e)\}_{e \in E}$ in $O(\sum_{i=1}^m \varepsilon_i + n)$ time. In step 3 and 4 we can get $\{D(v_j)\}_{v_j \in V(T)}$ and $\{F(T_{v_j}, v_j)\}_{v_j \in V(T)}$ in $O(n)$ time through a bottom-up tree traversal respectively. Obviously, $F(T_{v_1}, v_1) = F(T, v_1)$. If $F(T, v_1)$ is obtained, we can compute $\{F(T, v_j)\}_{v_j \in V(T)}$ in $O(n)$ time through a top-down tree traversal. At last step 6 take $O(n)$ time to find the medians of T. Therefore, we obtain the following theorem.

Theorem 7. *The pos/neg-weighted 1-median problem on a tree with discrete subtree-shaped customers can be solved in* $\max\{O(\sum_{i=1}^m \varepsilon_i + n),\ O(m+n)\}$ *time.*

Here we give an example in Fig. 1 to illustrate Algorithm A and the resulting values are shown in Figs. 2 and 3 with the optimal solution v_{10}.

For the continuous case, i.e., at least there is a subtree in $\mathcal{T}$ which is not discrete. Based on tree graph $T = (V, E)$, we construct a new tree graph $T' = (V', E')$, V' is obtained by adding all the points in $\bigcup_{i=1}^m L_i^*$ which are not in V to V, the weights of these new vertices are large enough, and connect the new vertices to their adjacent vertices in V, the edge-lengths for the edges that are incident with these new points are corresponding updated. Note that this preprocessing can be done in $O(\sum_{i=1}^m n_i^*)$ time and we get a modified tree graph $T' = (V', E')$ with $|V'| = O(n + \sum_{i=1}^m n_i^*)$. On this modified tree, all subtrees are discrete. Thus in order to solve the pos/neg-weighted 1-median problem of the continuous case, we shall apply Algorithm A and get

Corollary 8. *The pos/neg-weighted 1-median problem of the continuous case can be solved in* $\max\{O(n + \sum_{i=1}^m n_i^* + \sum_{i=1}^m \varepsilon_i),\ O(n + \sum_{i=1}^m n_i^* + m)\}$ *time.*

Acknowledgements

The authors are grateful to the referees for their valuable comments, which have led to improvements in the presentation of the paper.

References

[1] R. Benkoczi, Cardinality constrained facility location problems in trees, Ph.D. Thesis, Simon Fraser University, 2004.

[2] R. Benkoczi, B. Bhattacharya, A new template for solving p-median problems for trees in sub-quadratic time, in: G.S. Brodal, S. Leonardi (Eds.), ESA 2005, in: LNCS, vol. 3669, Springer, Berlin Heidelberg, 2005, pp. 271–282.
[3] R. Benkoczi, D. Breton, B. Bhattacharya, Efficient computation of 2-medians in a tree network with positive/negative weights, Discrete Mathematics 306 (2006) 1505–1516.
[4] R.E. Burkard, E. Çela, H. Dollani, 2-median in trees with pos/neg weights, Discrete Applied Mathematics 105 (2000) 51–71.
[5] R.E. Burkard, J. Fathali, A polynomial method for the pos/neg weighted 3-medians problem on a tree, Mathematical Methods of Operations Research 65 (2007) 229–238.
[6] R.E. Burkard, J. Krarup, A linear algorithm for the pos/neg-weighted median problem on a cactus, Computing 60 (1998) 193–215.
[7] J.M. Díaz-Báñez, J.A. Mesa, A. Schöbel, Continuous location of dimensional structures, European Journal of Operational Research 152 (2004) 22–44.
[8] O. Kariv, S.L. Hakimi, An algorithmic approach to network location problems. Part II p-medians, SIAM Journal on Applied Mathematics 37 (1979) 539–560.
[9] P. Mirchandani, The p-median problem and generalizations, in: P. Mirchandani, R. Francis (Eds.), Discrete Location Theory, Wiley, New York, 1990, pp. 55–117.
[10] S. Nickel, J. Puerto, A.M. RodríguezChía, An approach to location models involving sets as existing facilities, Mathematics of Operations Research 28 (2003) 693–715.
[11] F. Plastria, Continuous covering location problems, in: Z. Drezner, H.W. Hamacher (Eds.), Facility Location: Applications and Theory, Springer, Berlin, Heidelberg, New York, 2002, (Chapter 2).
[12] J. Puerto, A. Tamir, J.A. Mesa, D. Pérez-Brito, Center location problems on trees graphs with subtree-shaped customers, Discrete Applied Mathematics (2008) doi:10.1016/j.dam.2007.11.022.
[13] A. Tamir, An $O(pn^2)$ algorithm for the p-median and related problems on tree graphs, Operations Research Letters 19 (1996) 59–64.