Journal of Southeast University (English Edition) Vol. 29, No. 3, pp. 346 - 351 Sept. 2013 ISSN 1003—7985

2-median location improvement problems under weighted l_1 norm and l_∞ norm on trees

Yang Liping Guan Xiucui

(Department of Mathematics, Southeast University, Nanjing 211189, China)

Abstract: This paper focuses on the 2-median location improvement problem on tree networks, and the problem is to modify the weights of edges at the minimum cost such that the overall sum of the weighted distance of the vertices to the respective closest one of two prescribed vertices in the modified network is upper bounded by a given value. l_1 norm and l_∞ norm are used to measure the total modification cost. These two problems have a strong practical application background and important theoretical research value. It is shown that such problems can be transformed into a series of sum-type and bottleneck-type continuous knapsack problems, respectively. Based on the property of the optimal solution, two $O(n^2)$ algorithms for solving the two problems are proposed, where n is the number of vertices on the tree.

Key words: 2-median; network improvement problem; tree; knapsack problem; l_1 norm; l_∞ norm

doi: 10.3969/j.issn.1003 - 7985.2013.03.021

The network location problems consider how to find the best locations for p critical facilities in a given network to provide rapid and effective service. Generally speaking, the best locations for p critical facilities are to minimize the overall sum of the weighted distance of the vertices to the respective closest facility, which are known as network p-medians. However, the surrounding environment may have changed with the development of the economy, so the old network median facilities may not be in the optimal location. As moving an existing facility may be more expensive than improving the facility network, three classes of inverse network location problems are proposed. The first class is the inverse network location problem[1-3], which is to modify the vertex/edge weights at the minimum cost such that the given p vertices become the p-medians (centers). The second class is the reverse network location problem[4-5], which is to modify the vertex/edge weights within a certain budget to minimize the sum of the weighted distances of p-medians (centers) in the modified network. The third class is the network location improvement problem[6-7], which is to modify the vertex/edge weights at the minimum cost such that the sum of the weighted distances of p-medians (centers) in the modified network is upper bounded by a given value.

For the inverse 1-median location problems under l_1 norm by modifying the vertex weights on trees, Burkard et al.[1] proposed an $O(n\log n)$ greedy algorithm. Guan et al.[2-3] solved the problem under l_∞ norm and bottleneck-Hamming distance in $O(n)$ time, but the problem under weighted sum-Hamming distance was shown to be NP-hard.

For the reverse median location problems, Berman et al.[4] considered the problem on a tree by shortening the edges to minimize the weights distance of the shortest path and showed that it can be solved in strongly polynomial time. Burkard et al.[5] transformed both the reverse 2-median problems on trees and the reverse 1-median problems on special cacti to the reverse 3-median problems on paths, and proposed an $O(n\log n)$ algorithm.

For the 1-median location improvement problem, Bai et al.[6-7] proved that the problem under the Hamming distance on a general graph and a cycle are all NP-hard, and proposed a pseudo polynomial time algorithm based on the dynamic programming method.

In this paper, we consider the 2-median location improvement problems on a tree under weighted l_1 and l_∞ norms by reducing the lengths of edges, which are simply denoted by problems $2\mathrm{MLIP}_1$ and $2\mathrm{MLIP}_\infty$, respectively. Two polynomial time algorithms are presented to solve them.

1 Problem Formulation

In this section, we first construct the mathematical models for problems $2\mathrm{MLIP}_1$ and $2\mathrm{MLIP}_\infty$, then transform them into some sum-type and bottleneck-type continuous knapsack problems, respectively.

Let $T=(V, E)$ be a tree network, and $|V|=n$. Let l_i and l_i^* denote the length and the reduced length of edge e_i, respectively. The reduction $x_i = l_i - l_i^*$ of e_i is upper bounded by u_i, where $0 \leqslant u_i < l_i$. The distance $d^l(v_i, v_j)$ means the length of a shortest path from v_i to v_j with respect to l_i. Denote c_i the cost of reducing one unit length of e_i. Assume that s and t are the two prescribed vertices,

Received 2012-08-04.

Biographies: Yang Liping (1987—), female, graduate; Guan Xiucui (corresponding author), female, associate professor, xcguan@163.com.

Foundation item: The National Natural Science Foundation of China (No. 10801031).

Citation: Yang Liping, Guan Xiucui. 2-median location improvement problems under weighted l_1 norm and l_∞ norm on trees[J]. Journal of Southeast University (English Edition), 2013, 29(3): 346 - 351. [doi: 10.3969/j.issn.1003 - 7985.2013.03.021]

and the problem is to reduce the lengths of edges such that the sum of the weighted distances of the vertices to the closest vertices in s and t is no more than the given upper bound U. The objective is to minimize the cost measured by weighted l_1 norm and l_∞ norm. The mathematical models of problems 2MLIP_1 and 2MLIP_∞ can be respectively formulated as

$$\min \sum_{e_j \in E} c_j x_j$$

$$\text{s. t.} \quad \sum_{v_i \in V} w_i \min_{v \in \{s,t\}} d^{l^*}(v_i, v) \leqslant U$$

$$0 \leqslant x_j \leqslant u_j,\ e_j \in E$$

$$\min \max_{e_j \in E} c_j x_j$$

$$\text{s. t.} \quad \sum_{v_i \in V} w_i \min_{v \in \{s,t\}} d^{l^*}(v_i, v) \leqslant U$$

$$0 \leqslant x_j \leqslant u_j,\ e_j \in E$$

Let $P[s,t]$ be the unique path from vertex s to t, and say $P[s,t] = \{e_1, e_2, \dots, e_{p-1}\}$, $E - P[s,t] = \{e_p, e_{p+1}, \dots, e_{n-1}\}$.

Definition 1 k is called as a critical index in path $P[s,t]$ with respect to the modified edge length l^*, if $\min_{v \in \{s,t\}} d^{l^*}(v, v_k) = d^{l^*}(s, v_k)$, $\min_{v \in \{s,t\}} d^{l^*}(v, v_{k+1}) = d^{l^*}(t, v_{k+1})$, where $(v_k, v_{k+1}) \in P[s,t]$.

If all the edges of path $P[s,t]$ are deleted from the tree T, then T is divided into p disjoint subtrees. The subtree rooted at the only vertex v_k of $P[s,t]$ is denoted by T_k. Furthermore, we write $r(v) = v_k$ if $v \in T_k$ and $v_k \in P[s,t]$. Now a new weight $\tilde{w}_k$ for vertex $v_k \in P[s,t]$ is introduced by

$$\tilde{w}_k = \sum_{v \in T_k} w_v \tag{1}$$

For a fixed index $k = 1, 2, \dots, p$, a new weight vector for edge $e_i \in T$ is introduced by

$$W_i^k = \begin{cases} \sum_{j=i+1}^{k} \tilde{w}_j & i = 1, 2, \dots, k-1 \\ 0 & i = k \text{ or } i = p \\ \sum_{j=k+1}^{i} \tilde{w}_j & i = k+1, \dots, p-1 \\ \sum_{\{v:\, e_i \in P[v, r(v)]\}} w_v & i = p+1, \dots, n-1 \end{cases} \tag{2}$$

Lemma 1 When k is the critical index, the weighted distance of 2-medians s and t with respect to l^* can be calculated by

$$\sum_{v_i \in V} w_i \min_{v \in \{s,t\}} d^{l^*}(v_i, v) = \sum_{i=1}^{n-1} (l_i - x_i) W_i^k \tag{3}$$

Proof Note that $d^{l^*}(v_i, r(v_i)) = 0$ if $v_i \in P[s,t]$. Hence, for vertices $v_i \in V$ and $v \in \{s,t\}$, $d^{l^*}(v_i, v) = d^{l^*}(v_i, r(v_i)) + d^{l^*}(r(v_i), v)$, so

$$\sum_{v_i \in V} w_i \min_{v \in \{s,t\}} d^{l^*}(v_i, v) = \sum_{v_i \in V} w_i d^{l^*}(v_i, r(v_i)) + \sum_{v_i \in V} w_i \min_{v \in \{s,t\}} d^{l^*}(r(v_i), v)$$

Obviously, the weighted distance of 2-median can be divided into 2 terms. The first term is the weighted distance of 1-median on trees, and the second term is the weighted distance of 2-median on paths. Simplify the two terms based on the critical index k.

$$\sum_{v_i \in V} w_i d^{l^*}(v_i, r(v_i)) = \sum_{v_i \in V} \left(w_i \sum_{e_i \in P[v_i, r(v_i)]} (l_i - x_i) \right) = \sum_{v_i \notin p[s,t]} \left(\sum_{\{v_i:\, e_i \in P[v_i, r(v_i)]\}} w_i \right) (l_i - x_i) = \sum_{i=p+1}^{n-1} (l_i - x_i) W_i^k \tag{4}$$

$$\begin{aligned} &\sum_{v_i \in V} w_i \min_{v \in \{s,t\}} d^{l^*}(r(v_i), v) = \\ &\sum_{j=1}^{p} \min_{v \in \{s,t\}} \Big(\sum_{v_i \in T_j} w_i \Big) d^{l^*}(v_j, v) = \sum_{j=1}^{p} \tilde{w}_j \min_{v \in \{s,t\}} d^{l^*}(v_j, v) = \\ &\sum_{j=1}^{k} (\tilde{w}_j d^{l^*}(s, v_j)) + \sum_{j=k+1}^{p} (\tilde{w}_j d^{l^*}(t, v_j)) = \\ &\sum_{j=1}^{k} \left(\tilde{w}_j \sum_{i=1}^{j-1} (l_i - x_i) \right) + \sum_{j=k+1}^{p} \left(\tilde{w}_j \sum_{i=j}^{p} (l_i - x_i) \right) = \\ &\sum_{r=1}^{k} (l_r - x_r) \sum_{j=r+1}^{k} \tilde{w}_j + \sum_{r=k+1}^{p} (l_r - x_r) \sum_{j=k+1}^{r} \tilde{w}_j = \\ &\sum_{r=1}^{k} (l_r - x_r) W_r^k + \sum_{r=k+1}^{p} (l_r - x_r) W_r^k = \\ &\sum_{i=1}^{p} (l_i - x_i) W_i^k \end{aligned} \tag{5}$$

So the weighted distance of 2-median can be denoted by the sum of Eqs. (4) and (5).

Problems 2MLIP_1 and 2MLIP_∞ with the critical index k can be stated respectively as the following problems P_{1k} and $P_{\infty k}$:

$$\min \sum_{i=1}^{n-1} c_i x_i$$

$$\text{s. t.} \quad \sum_{i=1}^{n-1} x_i W_i^k \geqslant \sum_{i=1}^{n-1} l_i W_i^k - U$$

$$0 \leqslant x_j \leqslant u_j,\ e_j \in E$$

$$\min \max_{1 \leqslant i \leqslant n-1} c_i x_i$$

$$\text{s. t.} \quad \sum_{i=1}^{n-1} x_i W_i^k \geqslant \sum_{i=1}^{n-1} l_i W_i^k - U$$

$$0 \leqslant x_j \leqslant u_j,\ e_j \in E$$

Problems P_{1k} and $P_{\infty k}$ are well-known continuous knapsack and bottleneck-type continuous knapsack problems, respectively. So problems 2MLIP_1 and 2MLIP_∞ can be transformed into some continuous knapsack and bottleneck-type knapsack problems corresponding to the critical index k, respectively.

2 $O(n^2)$ Algorithm for 2MLIP$_1$

In this section, some properties of optimal solutions for problem 2MLIP$_1$ are proved, and an $O(n^2)$ algorithm of problem 2MLIP$_1$ is devised based on the algorithms of continuous knapsack problems.

Obviously, if $\sum_{j=1}^{n-1}(l_j - u_j)W_j^k > U$, then the problem has no feasible solution. If $\sum_{j=1}^{n-1} l_j W_j^k \leqslant U$, then the optimal solution is $x=0$. Without loss of generality, we assume that $\sum_{j=1}^{n-1}(l_j - u_j)W_j^k \leqslant U < \sum_{j=1}^{n-1} l_j W_j^k$.

Lemma 2 When k' is not critical index k_0, then $\sum_{i=1}^{p} \tilde{w}_i \min_{v\in\{s,t\}} d^{l^*}(v, v_i) \leqslant \sum_{i=1}^{p} \tilde{w}_i d^{l^*}(st, i, k')$, where

$$d^{l^*}(st, i, k') = \begin{cases} d^{l^*}(s, v_i) & i \leqslant k' \\ d^{l^*}(t, v_i) & i \geqslant k'+1 \end{cases}$$

Proof Assume that k_0 is the critical index on path $P[s,t]$ with respect to l^*, and $\min_{v\in\{s,t\}} d^{l^*}(v, v_{k_0}) = d^{l^*}(s, v_{k_0})$, $\min_{v\in\{s,t\}} d^{l^*}(v, v_{k_0+1}) = d^{l^*}(t, v_{k_0+1})$. Let

$$\min_{v\in\{s,t\}} d^{l^*}(v, v_i) = \begin{cases} d^{l^*}(s, v_i) & 1 \leqslant i \leqslant k_0 \\ d^{l^*}(t, v_i) & k_0+1 \leqslant i \leqslant p \end{cases}$$

We only need to show that $\min_{v\in\{s,t\}} d^{l^*}(v, v_i) \leqslant d^{l^*}(st, i, k')$ when $k' \geqslant k_0$. In the case $k' \leqslant k_0$, we can prove it similarly. Next we consider the following three cases when $k' \geqslant k_0$.

1) If $i \leqslant k_0 \leqslant k'$, then $\min_{v\in\{s,t\}} d^{l^*}(v, v_i) = d^{l^*}(s, v_i) = d^{l^*}(st, i, k')$.

2) If $k_0 \leqslant i \leqslant k'$, then $\min_{v\in\{s,t\}} d^{l^*}(v, v_i) = d^{l^*}(t, v_i) \leqslant d^{l^*}(s, v_i) = d^{l^*}(st, i, k')$.

3) If $k_0 \leqslant k' \leqslant i$, then $\min_{v\in\{s,t\}} d^{l^*}(v, v_i) = d^{l^*}(t, v_i) = d^{l^*}(st, i, k')$.

Therefore, $\sum_{i=1}^{p} \tilde{w}_i \min_{v\in\{s,t\}} d^{l^*}(v, v_i) \leqslant \sum_{i=1}^{p} \tilde{w}_i d^{l^*}(st, i, k')$ holds for $k' \geqslant k_0$ and similarly for $k' < k_0$.

Lemma 3 Let the feasible regions of problems 2MLIP$_1$ and P_{1k} be D and D_k, respectively. Then $D = \bigcup_{k=1}^{p} D_k$.

Proof Assume that $D' = \bigcup_{k=1}^{p} D_k$. First, we show that $D \subseteq D'$. Suppose that x is a feasible solution in D. Then the critical index k can be determined by the modified length vector $l^* = l - x$. Then we know that x is a feasible solution in D_k for some critical index k by Lemma 1, thus $D \subseteq D'$.

Now we show that $D' \subseteq D$. For all $x \in D'$, there must exist $1 \leqslant k \leqslant p-1$ satisfying $x \in D_k$, then $\sum_{i=1}^{n-1}(l_i - x_i)W_i^k \leqslant U$.

1) If k is the critical index, then we have $\sum_{v_i\in V} w_i \min_{v\in\{s,t\}} d^{l^*}(v_i, v) = \sum_{i=1}^{n-1}(l_i - x_i)W_i^k \leqslant U$ by Lemma 1.

2) If k is not the critical index, we can obtain the following results:

$$\begin{aligned}
&\sum_{v_i\in V} w_i \min_{v\in\{s,t\}} d^{l^*}(v_i, v) = \\
&\quad \sum_{v_i\in V} w_i d^{l^*}(v_i, r(v_i)) + \sum_{v_i\in V} w_i \min_{v\in\{s,t\}} d^{l^*}(r(v_i), v) = \\
&\quad \sum_{i=p+1}^{n-1}(l_i - x_i)W_i^k + \sum_{i=1}^{p} \tilde{w}_i \min_{v\in\{s,t\}} d^{l^*}(v_i, v) \leqslant \\
&\quad \sum_{i=p+1}^{n-1}(l_i - x_i)W_i^k + \sum_{i=1}^{p} \tilde{w}_i d^{l^*}(st, i, k) = \\
&\quad \sum_{i=p+1}^{n-1}(l_i - x_i)W_i^k + \sum_{i=1}^{k} \tilde{w}_i d^{l^*}(s, v_i) + \sum_{i=k+1}^{p} \tilde{w}_i d^{l^*}(t, v_i) = \\
&\quad \sum_{i=p+1}^{n-1}(l_i - x_i)W_i^k + \sum_{i=1}^{p}(l_i - x_i)W_i^k = \\
&\quad \sum_{i=1}^{n-1}(l_i - x_i)W_i^k \leqslant U
\end{aligned}$$

The second equality holds by Eq. (4). The inequality holds based on Lemma 2. The fifth equality holds by Eq. (5).

Thus, whether k is the critical index or not, we have $x \in D$, and therefore $D' \subseteq D$.

Note that the objection functions of problems 2MLIP$_1$ and P_{1k} are the same, so the optimal value of 2MLIP$_1$ is the minimum one among $p-1$ optimal values of continuous knapsack problems P_{1k}, which can easily lead to the following two theorems:

Theorem 1 The 2-median location improvement problems on trees under weighted l_1 norm can be solved by $p-1$ continuous knapsack problems P_{1k}, where $2 \leqslant p \leqslant n-1$.

Theorem 2 The 2-median location improvement problems on trees under weighted l_∞ norm can be solved by $p-1$ bottleneck-type continuous knapsack problems $P_{\infty k}$, where $2 \leqslant p \leqslant n-1$.

Next we will transform problem P_{1k} into the standard continuous knapsack form and propose an $O(n^2)$ algorithm to solve problem 2MLIP$_1$.

Let $\tilde{x}_i = 1 - \dfrac{x_i}{u_i}$, $\tilde{p}_i^k = u_i W_i^k$, $\tilde{b}_k = \sum_{i=1}^{n-1} \tilde{p}_i^k - \sum_{i=1}^{n-1} l_i W_i^k - U$, $\tilde{c}_i = c_i u_i$. Then, the objective function of problem P_{1k} can be written as $\min \sum_{i=1}^{n-1} c_i x_i = \sum_{i=1}^{n-1} \tilde{c}_i - \max \sum_{i=1}^{n-1} \tilde{c}_i \tilde{x}_i$.

Thus, we only need to solve the following continuous knapsack problem CP$_{1k}$ to solve problem P_{1k}:

$$\max \sum_{i=1}^{n-1} \tilde{c}_i \tilde{x}_i$$

$$\text{s. t.} \quad \sum_{i=1}^{n-1} \tilde{p}_j^k \tilde{x}_i \leqslant \tilde{b}_k$$

$$0 \leqslant \tilde{x}_i \leqslant 1, 1 \leqslant i \leqslant n-1$$

Let $u_j^k = \dfrac{\tilde{c}_j}{\tilde{p}_j^k}$ and assume that $u_1^k \geqslant u_2^k \geqslant \ldots \geqslant u_{n-1}^k$.

Definition 2 e_r is named as the critical edge if $r = \min\left\{j: \sum_{i=1}^{j} \tilde{p}_i^k \geqslant \tilde{p}_k\right\}$.

The greedy idea[8] can lead to the following theorem:

Theorem 3 The optimal solution x of problem CP_{1k} is

$$\bar{x}_j = \begin{cases} 1 & j = 1, 2, \ldots, r-1 \\ \dfrac{\tilde{b}_k - \sum_{i=1}^{r-1} \tilde{p}_i^k}{\tilde{p}_r^k} & j = r \\ 0 & j = r+1, \ldots, n \end{cases}$$

The corresponding optimal solution of P_{1k} is $x_j^* = u_j(1 - x_j)$, $j = 1, 2, \ldots, n$ and its optimal value is

$$f_{1k} = \sum_{i=r}^{n-1} \tilde{c}_i - \tilde{c}_r \frac{\tilde{b}_k - \sum_{i=1}^{r-1} \tilde{p}_i^k}{\tilde{p}_r^k} \tag{6}$$

where e_r is the critical edge.

So the key to solve problem CP_{1k} is to find the critical edge e_r.

Balas and Zemel[9] proposed an $O(n)$ algorithm to find such a critical item. The detailed algorithm for problem $2MLIP_1$ is stated as follows:

Algorithm 1

Input the weight vector $\boldsymbol{w}$ of vertices, the length vector $\boldsymbol{l}$, the upper bound vector $\boldsymbol{u}$ and the cost vector $\boldsymbol{c}$ of edges, and two prescribed vertices s, t.

Search the path $P[s, t] = \{e_1, \ldots, e_{p-1}\}$ by BFS, and compute the weight W_i^k of edges by Eq. (2), where $p \leqslant i \leqslant n-1$.

Initialize $f := \infty$, $y := 0$

For $k = 1$ to $p-1$ do

Compute the weight W_i^k of edges by Eq. (2), where $1 \leqslant i \leqslant p-1$.

If $\sum_{j=1}^{n-1} (l_j - u_j) W_j^k > U$, break (There is no feasible solution for problem P_{1k});

else if $\sum_{j=1}^{n-1} l_j W_j^k \leqslant U$, the optimal solution of P_{1k} is $y_1 := 0$, and its optimal value is $f_1 := 0$.

else call the critical item procedure in Ref. [4] to solve problem P_{1k}. Let $f_1 := f_{1k}$ (The optimal value of problem P_{1k}), and $y_1 := x^*$ (The optimal solution of problem P_{1k}).

If $f \leqslant f_1$, then $f := f_1$; $y := y_1$;

Output the optimal value f of problem $2MLIP_1$ and its optimal solution y.

Now we analyze the time complexity of Algorithm 1. It runs at most n iterations and the main work in each iteration is to call the critical item procedure in $O(n)$ operations[9]. So the time complexity of Algorithm 1 is $O(n^2)$.

3 $O(n^2)$ Algorithm for $2MLIP_\infty$

Similar to Section 2, some properties of the optimal solution for problem $2MLIP_\infty$ are proved and an $O(n^2)$ algorithm is proposed based on the algorithm of the bottleneck-type continuous knapsack problems.

Assume that $U' = \sum_{i=1}^{n-1} l_i W_i^k - U$. We first consider the unbounded problem $P'_{\infty k}$, and its mathematical model can be formulated as follows:

$$\min \max_{1 \leqslant i \leqslant n-1} c_i x_i$$

$$\text{s. t.} \quad \sum_{i=1}^{n-1} x_i W_i^k \geqslant U'$$

To minimize the maximum cost, we need to make $c_i x_i$ equal for any edge e_i. Assume that the optimal objective value of $P'_{\infty k}$ is Q, then $x_i = \dfrac{Q}{c_i}$ and $\sum_{i=1}^{n-1} x_i W_i^k = \sum_{i=1}^{n-1} \dfrac{Q}{c_i} W_i^k = Q \sum_{i=1}^{n-1} \dfrac{W_i^k}{c_i} \geqslant U'$. Hence $Q \geqslant Q_0 := U' / \sum_{i=1}^{n-1} \dfrac{W_i^k}{c_i}$. Obviously, $Q = Q_0$ is the optimal value of $P'_{\infty k}$, and $x_i = Q_0 / c_i$ is its optimal solution.

Theorem 4 If problem $P_{\infty k}$ is feasible, and assume that $E_B := \{e_i \in E \mid \tilde{c}_i < f_{\infty k}\}$, then the optimal value and the optimal solution respectively are

$$f_{\infty k} = \frac{U' - \sum_{e_i \in E_B} u_i W_i^k}{\sum_{e_i \notin E_B} W_i^k / c_i} \tag{7}$$

$$x_i = \begin{cases} u_i & e_i \in E_B \\ \dfrac{f_{\infty k}}{c_i} & e_i \notin E_B \end{cases} \tag{8}$$

Proof Note that $f_{\infty k}$ is the optimal value of problem $P_{\infty k}$, then $c_i x_i = f_{\infty k}$ for any $e_i \notin E_B$, and $x_i = f_{\infty k}/c_i \leqslant u_i$. It is easy to know that $f_{\infty k}$ is also the optimal value of the unbounded case:

$$\min \max_{e_i \notin E_B} c_i x_i$$

$$\text{s. t.} \quad \sum_{e_i \notin E_B} x_i W_i^k \geqslant U' - \sum_{e_i \in E_B} u_i W_i^k$$

And hence the optimal value $f_{\infty k}$ is calculated by Eq. (7).

Now we show that x is an optimal solution of problem $P_{\infty k}$. Obviously $0 \leqslant x_i \leqslant u_i$, $i = 1, 2, \ldots, n-1$. We have

$$\sum_{i=1}^{n-1} x_i W_i^k = \sum_{e_i \in E_B} x_i W_i^k + \sum_{e_i \notin E_B} x_i W_i^k = \sum_{e_i \in E_B} u_i W_i^k + \sum_{e_i \notin E_B} \frac{f_{\infty k}}{c_i} W_i^k =$$

$$\sum_{e_i \in E_B} u_i W_i^k + \frac{U' - \sum_{e_i \in E_B} u_i W_i^k}{\sum_{e_i \notin E_B} \frac{W_i^k}{c_i}} \sum_{e_i \notin E_B} \frac{W_i^k}{c_i} =$$

$$\sum_{e_i \in E_B} u_i W_i^k + U' - \sum_{e_i \in E_B} u_i W_i^k = U'$$

Therefore, x is the feasible solution, whose object value is the optimal value. Hence it is an optimal solution of problem $P_{\infty k}$.

Next we present the main idea to solve problem $P_{\infty k}$. Note that $\tilde{c}_i = c_i u_i$, $i = 1, 2, \ldots, n-1$. Sort the values $\tilde{c}_i$ in a non-decreasing way, and say $\tilde{c}_{\tau(1)} < \tilde{c}_{\tau(2)} < \ldots < \tilde{c}_{\tau(q)}$. $f_{\infty k}$ must be in some interval $(\tilde{c}_{\tau(j)}, \tilde{c}_{\tau(j+1)}]$, and the optimal solution is $x_i = u_i$ if $i \leqslant \tau(j)$, and $x_i \leqslant u_i$ if $i > \tau(j)$. So we only need to find the critical edge.

Assume that

$$E_<: = \{e_i \in E \mid \tilde{c}_i < \lambda\},\ E_=: = \{e_i \in E \mid \tilde{c}_i = \lambda\}$$

$$E_>: = \{e_i \in E \mid \tilde{c}_i > \lambda\}$$

$$f_a: = \max_{e_i \in E_<} \tilde{c}_i,\ f_b: = \max_{e_i \in E_= \cup E_>} \tilde{c}_i$$

$$t_2: = \sum_{e_i \in E_=} \frac{W_i^k}{c_i} + \sum_{e_i \in E_>} \frac{W_i^k}{c_i},\ u_<: = \sum_{e_i \in E_<} \tilde{c}_i$$

Since

$$\tilde{c}_{\tau(j)} < f_{\infty k} < \tilde{c}_{\tau(j+1)} \tag{9}$$

the critical cost $\tilde{c}_{\tau(j)}$ is obviously the unique solution of the pair of inequalities

$$f_a t_2 < U' - \sum_{e_i \in E_<} u_i W_i^k \leqslant f_b t_2 \tag{10}$$

Let us detail the algorithm of the bottleneck-type continuous knapsack problem according to Ref. [3].

Algorithm 2

Input the weight vector $\boldsymbol{w}$ of vertices, the length vector $\boldsymbol{l}$, the upper bound vector $\boldsymbol{u}$ and the cost vector $\boldsymbol{c}$ of edges, and two prescribed vertices s, t.

Initialize $E_<: = \varnothing$, $E_=: = E$, $E_>: = \varnothing$, $t_1: = 0$, patition: = "no", $\Delta: = U'$.

While patition: = "no" do

Determine the median λ of the set of values $\{\tilde{c}_i \mid e_i \in E_=\}$.

Let $E_0: = \{e_i \in E_< \mid \tilde{c}_i < \lambda\}$, $E_1: = \{e_i \in E_= \mid \tilde{c}_i = \lambda\}$, $E_2: = \{e_i \in E_> \mid \tilde{c}_i > \lambda\}$, $f_a: = \max_{e_i \in E_0} \tilde{c}_i$, $f_b: = \max_{e_i \in E_1 \cup E_2} \tilde{c}_i$.

Compute $t_2: = \sum_{e_i \in E_1} \frac{W_i^k}{c_i} + \sum_{e_i \in E_2} \frac{W_i^k}{c_i} + t_1$, $\Delta_1: = \Delta - \sum_{e_i \in E_0} u_i$.

If $f_a t_2 < \Delta_1 \leqslant f_b t_2$, then patition: = "yes"

else if $\Delta_1 \leqslant f_a t_2$, then $E_>: = E_> \cup E_1 \cup E_2$,

$$t_1: = \sum_{e_i \in E_1} \frac{W_i^k}{c_i} + \sum_{e_i \in E_2} \frac{W_i^k}{c_i} + t_1,\ E_=: = E_0$$

else if $f_a t_2 < \Delta_1$, then $J: = \{e \in E \mid \tilde{c}_i = f_b\}$ and $E_J: = J \cup E_1 \cup E_0$.

Update $E_<: = E_< \cup E_1$, $E_=: = E_2 \backslash J$, $E_<: = E_< \cup E_1$,

Update $E_<: = E_< \cup E_0$, $E_>: = E_> \cup E_1 \cup E_2$.

Output the optimal value $f_{\infty k}$ and optimal solution x_k^* by Eqs. (7) and (8), where E_B is replaced by $E_<$.

Algorithm 3, which is to solve 2MLIP$_\infty$, is similar to Algorithm 1. The only difference of the two algorithms is the last "else" part. Algorithm 3 is to call on Algorithm 2 to solve the bottleneck continuous knapsack problem $P_{\infty k}$. Algorithm 2 can be done in $O(n)$ operations similar to the algorithm in Ref. [3]. Hence the time complexity of Algorithm 3 is $O(n^2)$.

4 Conclusion

In this paper, we mainly explore the issue of the 2-median location improvement problems on trees under weighted l_1 and l_∞ norms. We show that the two problems can be transformed into a series of sum-type and bottleneck-type continuous knapsack problems, respectively. And hence they can all be solved in $O(n^2)$ operations.

For further research, we can consider p-medians location improvement problems on general networks and special networks including trees, cycles, etc. The cost incurred by the modification of vertex weights and edge weights can be measured under weighted l_1, l_∞ norms and weighted Hamming distances.

References

[1] Burkard R E, Pleschiutschnig C, Zhang J Z. Inverse median problems [J]. *Discrete Optimization*, 2004, **1**(1): 23 - 39.

[2] Guan X C, Zhang B W. Inverse 1-median problem on trees under weighted l_∞ norm [C]//*Lecture Notes in Computer Science*. Springer, 2010, **6124**: 150 - 160.

[3] Guan X C, Zhang B W. Inverse 1-median problem on trees under weighted Hamming distance [J]. *Journal of Global Optimization*, 2012, **54**(1): 75 - 82.

[4] Berman O. Improving the location of minsum facilities through network modification [J]. *Annals of Operations Research*, 1992, **40**(1): 1 - 16.

[5] Burkard R E, Gassner E, Hatzl J. Reverse 2-median problem on trees [J]. *Discrete Applied Mathematics*, 2008, **156**(11): 1963 - 1976.

[6] Bai Y Q, Wang Q, Wu L S. Reverse 1-median problem under Hamming distance [J]. *Computer Engineering and Applications*, 2011, **47**(19): 39 - 41. (in Chinese)

[7] Bai Y Q, Wang Q, Wu L S. Reverse 1-median problem on a cycle under Hamming distance [J]. *Mathematics in Practice and Theory*, 2011, **41**(17): 113 - 118. (in Chinese)

[8] Silvano M, Paolo T. *Knapsack problems: algorithms and computer implementations* [M]. New York: John Wiley and Sons Ltd., 1990: 13 - 18.

[9] Balas E, Zemel E. An algorthim for large zero-one knapsack problems [J]. *Operations Research*, 1980, **28**(5): 1130 - 1154.

在赋权 l_1 模和 l_∞ 模下树上的 2-重心选址改进问题

杨利平　关秀翠

(东南大学数学系,南京 211189)

摘要:研究了在树网络上的2-重心选址改进问题,该问题是指以最少的花费调整各边的权值使得修改后网络中所有顶点到2个预设点的赋权距离的和不超过给定的上界.采用 l_1 模和 l_∞ 模衡量总的修改花费.这2类问题具有较强的实际应用价值与理论研究价值.这2类改进问题可分别等价地转化为一系列的和型及瓶颈型的连续背包问题,基于最优解的特性,提出了时间复杂度为 $O(n^2)$ 的算法来求解这2类问题,其中 n 是树上顶点的个数.

关键词:2-重心;网络改进问题;树;背包问题;l_1 模;l_∞ 模

中图分类号:O224