Available at
http://pvamu.edu/aam
Appl. Appl. Math.
ISSN: 1932-9466

Applications and Applied Mathematics:
An International Journal
(AAM)

Vol. 12, Issue 1 (June 2017), pp. 70 - 77

A Simple Linear Time Algorithm for Computing a 1-Median on Cactus Graphs

Kien Trung Nguyen[1], Pham Van Chien[2], Ly Hong Hai[1], Huynh Duc Quoc[3]

[1]Department of Mathematics
Teacher College, Cantho University
Can Tho, Viet Nam
trungkien@ctu.edu.vn; lyhonghai23@gmail.com;

[2]Department of Mathematics
Cai Sach High School
Soc Trang, Viet Nam
phamvanchien.c3ks@soctrang.edu.vn;

[3]Department of Mathematics
Science College, Cantho University
Can Tho, Viet Nam
hdquoc@ctu.edu.vn;

Received August 30, 2016; Accepted February 20, 2017

Abstract

We address the problem of finding a 1-median on a cactus graph. The problem has already been solved in linear time by the algorithms of Burkard and Krarup (1998), and Lan and Wang (2000). These algorithms are complicated and need efforts. Hence, we develop in this paper a simpler algorithm. First, we construct a condition for a cycle that contains a 1-median or for a vertex that is indeed a 1-median of the cactus. Based on this condition, we localize the search for deriving a 1-median on the underlying cactus. Complexity analysis shows that the approach runs in linear time.

Keywords: Median problem; facilities; cactus graphs; tree; complexity; algorithm; linear

MSC 2010 No.: 90B10, 90B80, 90C27

1. Introduction

Location theory plays an important role in operational research with many applications in real life situation. Here, one finds optimal locations of new facilities based on the given locations of demands. The two most popular objective functions concerning the location problems are

median and center. For the median function, we aim to minimize the total weighted distances from demands to new facilities. On the other hand, a location problem with center function minimizes the maximum weighted distances to new facilities. Regarding the model and solution approach of location problems, readers may refer to Daskin (1995), Drezner and Hamacher (2002), and Eiselt et al. (2011).

Kariv and Hakimi (1979a, 1979b) showed in their papers that the *p*-median and *p*-center problems on general networks are NP-hard. Hence, it is worthwhile to further investigate some special cases such that the problems can be solved in polynomial time. Among them is the problem of locating exactly one new facility with median objective function yields special interest from the community. This problem is the so-called 1-median problem. Hua (1962) and Goldman (1971) independently solved the 1-median problem on tree in linear time based on the optimality criterion.

Recently, center and median location problem on cactus, a simple generalization of tree graphs, have been intensively investigated. For the center problem on cactus graphs, Ben-Moshe et al. (2007) developed an $O(nlogn)$ time algorithm that solves the 1-center problem and an $O(nlog^3n)$ time algorithm that solves the 2-center problem. They also improved the solutions for the *p*-center problem. Furthermore, Zmazek et al. (2004) proposed an $O(cn)$ algorithm that solves the obnoxious 1-center problem on weighted cactus graphs, where c is the number of different vertex weights. Concerning the median problem on cactus graphs, Burkard and Krarup (1998) solved the pos/neg-weighted 1-median on cactus in linear time. The 1-median problem with pos/neg weights is a generalization of the classical problem, as we always assume the weights of vertices are positive. Furthermore, Lan and Wang (2000) investigated the *p*-median problem on 4-cactus graphs, i.e., the cactus graphs with exactly 4 vertices on each cycle, and solved the problem by efficiently polynomial algorithm. The case $p = 1$ is solvable in linear time.

To the best of our knowledge, the most efficient algorithm to solve the 1-median problem on cactus graphs is the linear time algorithm of Burkard and Krarup (1998), and Lan and Wang (2000). However, the mentioned algorithm is rather complicated and needs efforts. Thus, we develop in this paper a simple algorithm to solve the 1-median problem on positive weighted cactus graphs with the same complexity, say linear time.

This paper is organized as follows. In Section 2 we consider preliminary concepts and construct a condition for one or two cycles that contain a 1-median. This condition allows us to localize our search on the desired cycle(s). We then develop a linear time algorithm to solve the 1-median on the cactus in Section 3.

2. Preliminaries and Optimality Criterion

This section aims to identify either a 1-median or a cycle which contains a 1-median of the cactus. To achieve the goal, we first define some preliminaries on the instance of network locations and revisit the definition of cactus graphs and trees.

Given a graph $G = (V, E)$ with vertex set V and edge set E. Each vertex $v \in V$ is associated with a positive weight w_v and each edge $e \in E$ has a positive length l_e. A point on G is either

a vertex or lies on some edge of *G*. The distance *d(a, b)* between two points *a* and *b* on *G* is the length of the shortest path connecting these two points. The 1-median problem on G is to find a point *p* that minimizes the function

$$\sum_{v\in V} w_v d(p,v).$$

The optimal solution of the 1-median problem is called a 1-median of *G*. According to the dominating set of *p*-median given by Kariv and Hakimi (1979b), there exists a 1-median of G that is also a vertex, i.e., it does not lies in the interior of an edge. Therefore, from here on we only focus on vertex of the graph in order to find a 1-median.

We recall the definition of trees and cactus graphs. A tree is an undirected graph in which any two vertices are connected by *exactly one* path. A cactus is a connected graph where any edge lies on at most one simple cycle. A cycle is simple if no edge and no vertex (except the starting and ending vertex) is repeated. If there is no cycle on the cactus, then it is exactly a tree.

Now we consider the 1-median problem on a cactus *G=(V,E)*. Denote by *W* the total weights of vertices on *G*, i.e., $W = \sum_{v\in V} w_v$. Let *C* be either a cycle or an edge of *G*. By deleting all edges of *C* from *G*, we get connected components which are also cactus graphs. For a vertex $v \in C$, we denote by G_C^v the corresponding connected component that contains *v*. Furthermore, for a subgraphs *S* of *G*, *W(S)* stands for the total weights of vertices in S, i.e., $W(S) = \sum_{v\in S} w_v$. We investigate the following lemma.

Lemma 1.

Let *C* be either a cycle or an edge on the cactus *G*, if there exists a connected component G_C^u for a vertex $u \in C$ such that $W(G_C^u) \geq \frac{W}{2}$, then there exists a 1-median of *G* which is contained in G_C^u.

Proof:

Let u' be an arbitrary vertex in *C* different from u. Denote by $G(u,u') := G \setminus (G_C^u \cup G_C^{u'})$, we get

$$\begin{aligned} f(u) - f(u') = &\sum_{v\in G_C^u} w_v (d(u,v) - d(u',v)) + \sum_{v\in G_C^{u'}} w_v (d(u,v) - d(u',v)) \\ &+ \sum_{v\in G(u,u')} w_v (d(u,v) - d(u',v)). \end{aligned} \tag{1}$$

As $d(u,v) - d(u',v) = -d(u,u')$ for $v \in G_C^u$, $d(u,v) - d(u',v) = d(u,u')$ for $v \in G_C^{u'}$ and $d(u,v) - d(u',v) \leq d(u,u')$ for $v \in G(u,u')$ (triangle inequality), we can rewrite (1) as follows

$$f(u) - f(u') \leq \Big(\sum_{v\in G_C^{u'}} w_v + \sum_{v\in G(u,u')} w_v - \sum_{v\in G_C^u} w_v\Big) d(u,u').$$

By the assumption $W(G_C^u) \geq \frac{W}{2}$, we further obtain

$$\sum_{v \in G_C^{u'}} w_v + \sum_{v \in G(u,u')} w_v - \sum_{v \in G_C^u} w_v = \sum_{v \in G_C^{u'}} w_v + \sum_{v \in G(u,u')} w_v + \sum_{v \in G_C^u} w_v - 2\sum_{v \in G_C^u} w_v$$
$$= W - 2\sum_{v \in G_C^u} w_v \leq 0. \tag{2}$$

By (2), we get $f(u') \geq f(u)$ for $u' \in C$ and $u' \neq u$.

Let $u'' \in G_C^{u'}$, we can prove that $f(u'') \geq f(u')$ by the same argument. In other words, any vertex in $G \setminus G_C^u$ is dominated by u. Hence, there exists a 1-median of G which is contained in G_C^u. □

For a cycle C on G, we denote by $B_C(v) := (G \setminus G_C^v) \cup v$. Based on Lemma 1., we can further investigate a condition for a cycle that contains a 1-median.

Lemma 2.

Given a cycle C on G, if the following condition holds

$$W(B_C(v)) \geq \frac{W}{2}, \quad \forall\, v \in C, \tag{3}$$

then C contains a 1-median of G.

Proof:

By Lemma 1, a 1-median of G is contained in $\bigcap_{v \in C} B_C(v) = C$. The proof comes follows.□

Note that if there exist two cycles that satisfy condition (3), then the common vertex of these two cycles is a 1-median of G as a consequence of Lemma 2.

In the next section we develop a simple linear time algorithm to find a 1-median of G based on the condition given in Lemma 2.

3. A Linear Time Approach

The idea for finding a 1-median of the cactus G is two phases. In the first phase we attempt to identify the cycle or an edge that contains a 1-median of the cactus. Then we compute and compare the median value at each vertex of the identified cycle or edge in order to obtain the optimal one.

3.1. Phase 1

We compute the total weight of the cactus and the weight of each cycle in G. It can be done in linear time by summing up the weights of all relevant vertices. Then, we represent the

cactus G as a rooted tree, or the so-called induced tree. Precisely, we consider each cycle of G as a vertex of the induced tree. If the two cycles on G have one vertex in common, then the two corresponding vertices on the induced tree are adjacent.

Now we consider which vertices represented by a 'cycle' vertex in the induced tree. If a cycle has no vertex in common with any other cycles, this 'cycle' vertex represents all vertices in the cycle. If two cycles have one vertex in common, the common vertex is represented in the parent 'cycle'. In the induced tree, the weight of a single vertex is the weight of itself in the cactus. Moreover, the weight of a 'cycle' vertex is the total weights of vertices that it represents. The induced tree and its vertex weights can be obviously computed in linear time.

For example, we consider a cactus and its induced tree in Figure 1. Assume that the tree is rooted at C_2. Then C_1 represents v_1, v_2, v_3; C_2 represents v_4, v_5, v_6, v_7; C_3 represents v_8, v_9, v_{10}. Note that as C_2 is the parent of C_1, the vertex v_4 is presented by C_2.

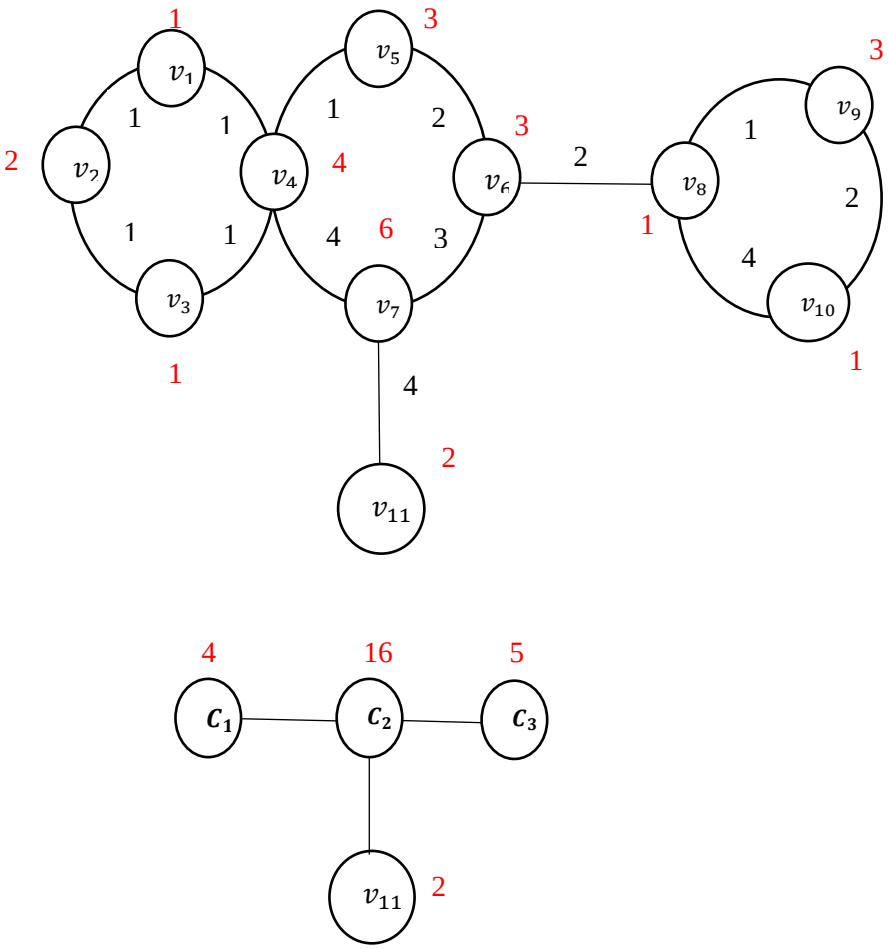

Figure 1. The cactus graph and the corresponding induced tree

Next we find either a 1-median of the cactus G or a cycle that contains a 1-median of G. We can complete the task by the following two steps.

Step 1: Find a 1-median of the induced tree in linear time, see Goldman (1971).

Step 2: If a 1-median of the induced tree is a single vertex, the corresponding vertex in the cactus graph is also a 1-median of cactus. Otherwise, a 1-median of the induced tree is a 'cycle', namely *C*. As *C* may contain common vertices with other cycles, we consider if a 1-median can be contained in other cycle by transferring the common vertices to other cycles and check the optimality criterion again. This can be done in linear time as the required total weights of each part of the induced tree has been already known.

Lemma 3.

There are at most two cycles in the cactus *G* that possibly contains a 1-median.

Proof:

The proof is straight forward as there exists at most two 1-median on an induced tree; see Golman (1971). □

Phase 2

Let *S* be the set of vertices in 1-median cycle(s) of the induced tree. By deleting all edges in the subgraph induced by *S*, we get connected components of *G*. Denote by *G(v)* the connected component that contains a vertex *v* in *S*. We aim to compute the distance from *v* to any other vertices in *G(v)*. For each cycle *C* in *G(v)*, we compute its length $l(C)$, i.e., $l(C) \coloneqq \sum_{e \in C} l(e)$. Then we apply a breath first search algorithm to compute the desired distances. In details, we start from the root *v* and tranverse to its children. Then we get the distance from *v* to its children which are the current vertices. Next we start from the current vertices and tranverse to their children. For a child *v'* which is not contained in any cycle, the computation is trivial. Otherwise, we can compute the left distance *d* from the parent to *v'* then compare *d* with $l(C) - d$ to get the minimum one. The procedure is iterated until the last vertex is considered.

After computing all required distances, we can compute the median objective functions at each vertex in *S*. Therefore, the complexity of the approach is linear.

In the proposed algorithm we first find the cycle that contains a 1-median of the cactus and then compute the objective value with respect to each vertex of the underlying cycle based on a recursive approach. The algorithm is straight forward and needs less efforts than the algorithm of Burkard and Krarup (1998) that searches for a 1-median of a neg/pos weighted cactus based on a decreasing direction of objective function. Moreover, the algorithm in this paper is also simpler than that of Lan and Wang (2000). Here, they study the *p*-median problem on 4-cactus graphs, i.e., the cactus with exactly four vertices on each cycle. They reduce the problem to the same one on tree graph. However, the case $p = 1$ on general cactus has not been specified so far. In short, we get the following main result.

Theorem 1.

The 1-median location problem on a cactus graph can be simply solved in linear time.

Example 1.

Given a cactus G in Figure 1. We aim to find a 1-median of G by applying the two-phased algorithm.

Phase 1. We can construct the corresponding tree T rooted at C_2. A 1-median of T is exactly C_2. After tranferring the vertex v_4 to C_1, the 'cycle' C_2 is still a 1-median of T. Therefore, a 1-median of the cactus is contained in C_2.

Phase 2. We can apply a breath first search algorithm to calculate the distance from v_6 to other vertices in $G(v_6)$ as follows.

1. The length of C_3 is $l(C_3) = 7$.

2. As v_8 is a child of v_6, we get $d(v_6, v_8) = 2$. Consider the two children of v_8, say v_9 and v_{10}. The left distances

$$d^L(v_8, v_{10}) = 4 > l(C_3) - d^L(v_8, v_{10}) = 3$$

and

$$d^L(v_8, v_9) = 6 > l(C_3) - d^L(v_8, v_9) = 1.$$

Therefore, $d(v_8, v_{10}) = 3$ and $d(v_8, v_9) = 1$. We imply $d(v_6, 10) = 5$ and $d(v_6, v_9) = 3$. By the same argument, we get the following results.

$$d(v_4, v_1) = d(v_4, v_3) = 1, d(v_4, v_2) = 2, d(v_7, v_{11}) = 4.$$

3. By elementary computation, we get the median value at each vertex on the cycle C_2.

Finally, we conclude that v_7 is a 1-median of G.

4. Conclusions

We considered the 1-median problem on cactus graphs. To solve this problem, we first established a condition for a vertex that is a 1-median or a cycle that contains a 1-median of the cactus. Then we developed a simple 2-phased algorithm that finds a 1-median of the cactus in linear time. For future research, we consider the 1-median problem on other types of graphs, e.g., interval graphs, block graphs, in order to explore if some special properties as in the cacti holds. Then we develop simple algorithms that solve the corresponding problems.

Acknowledgement:

The authors would like to thank the anonymous referees for their valuable comments which helped to improve the paper.

REFERENCES

Burkard, R.E. and Krarup, J. (1998). A linear algorithm for the pos/neg-weighted 1-median problem on a cactus. Computing, Vol. 60, pp. 193-215.

Ben-Moshe, B., Bhattacharya, B., Shi, Q. and Tamir, A. (2007). Efficient algorithms for center problems in cactus networks., Theoretical Computer Science, Vol. 378, pp. 237-252.

Daskin, M.S. (1995). Networks and Discrete Location, Models, Algorithms, and Applications. Wiley, New York.

Drezner, Z. and Hamacher, H. (2002). Facility Location: Applications and Theory. Springer, Heidelberg.

Eiselt, H.A. and Marianov, V. (2011). Foundations of location analysis, International Series in Operations Research and Management Science; Springer.

Goldman, A.J. (1971). Optimal center location in simple networks. Transport. Sci., Vol. 5, pp. 212–221.

Hua, L.K. (1962). Application of mathematical models to wheat harvesting, Chinese Mathematics, Vol. 2, pp. 539-560.

Kariv, O. and S.L. Hakimi, S.L. (1979a). An algorithmic approach to network location problems, I. The p-centers, SIAM Journal on Applied Mathematics, Vol. 37, pp. 513-538.

Kariv, O. and Hakimi, S.L. (1979b). An algorithmic approach to network location problems, II. The *p*-medians, SIAM Journal on Applied Mathematics, Vol. 37, pp. 536-560.

Lan, Y.F. and Wang, Y.L. (2000). An optimal algorithm for solving the 1-median problem on weighted 4-cactus graphs. Eur. J. Oper. Res., Vol. 122, pp. 602–610.

Zmazek, B. and Zerovnik, J. (2004). The obnoxious center problem on weighted cactus graphs, Discrete Applied Mathematics, Vol. 136, pp. 377-386.