Yugoslav Journal of Operations Research
8 (1998), Number 1, 129-135
Jovan Petrić memorial volume

A NOTE ON SPANNING TREES FOR NETWORK LOCATION PROBLEMS*

Dionisio PÉREZ-BRITO,
D.E.I.O.C. University of La Laguna, Tenerife, Spain,

Nenad MLADENOVIĆ,
G.E.R.A.D. University of Montreal, Montreal, Canada

José A. MORENO-PÉREZ,
D.E.I.O.C. University of La Laguna, Tenerife, Spain

Dedicated to the memory of Professor Jovan Petrić

Abstract: A p-facility location problem on a network N consists of locating p new facilities on N such that some function of the distances from them to the vertices of N is minimized. We consider a class of such problems where the objective function is nondecreasing in distance. Median, center and centdian problems belong to this class. We prove that the optimal solutions on the network and on the corresponding spanning trees are equal. Since location problems on a tree network are easier to solve than on a general one, we propose a descent local search heuristic that optimally solves the problem on a spanning tree at each iteration.

Keywords: Location, network, spanning trees, heuristic.

1. INTRODUCTION

Facility location analysis deals with the problem of locating one or several new facilities with respect to existing facilities (clients, users or demand points) in order to optimize some economic criteria (for an introduction to location analysis see, for example, [9]). Examples of facilities are plants, warehouses, schools, hospitals, administrative buildings, departure stores, waste material dumps, ambulance or fire

* Acknowledgement: The research has been partially supported by the DGCYT project PB95-1237-C03-0.

engine depots. The economic criteria is usually obtained by an objective function that decreases when the distances from the facilities to the clients decrease.

Network location problems occur when new facilities are to be located on a network (see [5]). The network of interest may be a road network, an air transport network, a river network, a telecommunication network or a network of computers. For a given network location problem, the network is represented by a graph and the new and existing facilities are often idealized as points. The demand points are generally taken to be at the vertices of the graph.

A special case arises when the underlying network has a tree structure (a connected graph without a cycle). Most of the problems are easier to solve on trees than on general graphs. The efficiency of the algorithms is due primarily to the general convexity properties of tree networks (see [4]). The algorithms for tree networks may be applied to a general network if the problem can be decomposed into subproblems on trees or if the graph has few cycles and the algorithms can be modified for nearly acyclic networks (many rural networks in a bounded region are nearly acyclic).

In this note we suggest a new heuristic way for solving location problems on a general network that uses the optimal solutions obtained on its spanning tree networks.

2. SPANNING TREES FOR NETWORK LOCATION

We denote by $N(V,E,l)$ a network with given set of vertices $V=\{v_1,\dots,v_n\}$ and set of edges $E=\{e_1,\dots,e_m\}$, where $l(e_i)$ is the length of edge $e_i \in E$, $i=1,\dots,m$. We say that a point $x \in N$ if it belongs to V or it lies on some edge. The point at an edge $[v_i,v_j]$ is given by the length of the subedge between it and one of the extremes; i.e. the vertices v_i and v_j. Let $X=\{x_1,\dots,x_p\}$ be a set of p new location points on the network N. Then the p-facility location-allocation problem we are considering is as follows:

$$\min\{\ f_N(X)=g(d_N(X)) : X \subseteq N, |X|=p\ \} \tag{1}$$

where $d_N(X)=(d_N(X,v_1),\dots,d_N(X,v_n))$ and $d_N(X,v_j)=\min_{x\in X} d_N(x,v_j)$, for $j=1,\dots n$.

The distance function $d_N(x,v_j)$ for location problems on networks is usually defined as the length of the shortest path between two points. A path between two points is a connected set of edges or subedges containing them. Thus, $d_N(x,v_j)$ is the sum of lengths l of the edges on the shortest path between x and v_j. Let us now assume that the globalizing function g used to define the objective function f is nondecreasing in distance, i.e.,

$$z' \le z'' \Rightarrow g(z') \le g(z'')$$

for all $z' = (z_1', \ldots, z_n')$ and $z'' = (z_1'', \ldots, z_n'')$. This assumption is natural because, in most real location problems, costs increase with distance. The special cases of problem (1) are *p*-median, *p*-center and *p*-centdian:

- For the *p*-median: $f_N(X) = g(d_N(X)) = \sum_{j=1}^{n} w_j \cdot d_N(X, v_j)$
- For the *p*-center: $f_N(X) = g(d_N(X)) = \max_{j=1..n} u_j \cdot d_N(X, v_j)$
- For the *p*-centdian: $f_N(X) = g(d_N(X)) = \sum_{j=1}^{n} w_j \cdot d_N(X, v_j) + \max_{j=1..n} u_j \cdot d_N(X, v_j)$

The weights w_j and u_j $(w_j, u_j \ge 0)$, of the *p*-median and *p* -center problems respectively are associated with each vertex (user) and are given. It is obvious that all three objective functions increase if the distances $d_N(X, v_j)$ increase, i.e., they are monotonically nondecreasing and so every user vertex v_j is allocated to its nearest facility point x_i. If there is a tie between two or more facility points, the user vertex v_j is assigned to the first in the list; i.e. if v_j is assigned to x_i and if $d_N(x_i, v_j) = d_N(x_k, v_j)$ then $i < k$. Thus, the set of paths from every facility point to the user vertices assigned to it is a tree. These trees are disjoint and the set of shortest paths from a set of *p* facility points *X* to the user vertices is a spanning *p*-forest; i.e. a set of *p* disjoint trees. Let $T(X)$ be the set of trees containing this forest. They can be obtained by joining these *p* trees with any $p-1$ edges that do not provide a cycle.

Let $ST(N)$ be the set of all spanning trees of *N*. For every $T \in ST(N)$, let $f_T(.)$ denote the objective function of (1) when the distances are obtained from the shortest paths with edges only in *T*; i.e., $f_T(X) = g(d_T(X))$ denotes with X_T^* the optimal solution of problem (1) in *T*, and with f_T^* the corresponding minimum value; i.e.;

$$f_T^* = f_T(X_T^*) = \min\{ f_T(X) : X \subseteq T, |X| = p \}.$$

Theorem. *Let* $X_N^* = \{ x_1^*, \ldots, x_p^* \}$ *be an optimal solution of location problem* (1) *on network N, where the globalizing function g is nondecreasing in distance. Then there exists a spanning tree* T^* *of N such that* X_N^* *is an optimal solution of* (1) *on* T^*, $(f_{T^*} = f_T(X_N^*))$.

Proof: For every $X \subseteq N$ and every $T \in ST(N)$, the following holds:

$$d_N(X, v_j) \le d_T(X, v_j), \ \forall j = 1, \ldots, n \Rightarrow f_N(X) \le f_T(X).$$

Then $f_N(X_N^*) \le f_T(X_N^*)$. Also $\forall X \subseteq N, T \in T(X)$ holds:

$$d_N(X,v_j) = d_T(X,v_j), \forall j = 1,\dots,n \Rightarrow f_N(X) = f_T(X).$$

Taking $X = X_N^*$ in the last equation, we have

$$f_N^* = f_N(X_N^*) = f_T(X_N^*), \forall T \in T(X_N^*).$$

Therefore, $\forall T^* \in T(X_N^*)$ holds:

$$f_{T^*}(X_N^*) \le f_T(X), \forall X \subseteq N, T \in ST(N).$$

Let any $T^* \in T(X_N^*)$. Since X_N^* is an optimal solution of the p-facility location problem (1) on N then

$$f_{T^*}(X_N^*) = f_N(X_N^*) = f_N^* \le f_{T^*}^*.$$

Hence $f_{T^*}^* = f_N^*$. Finally X_N^* is an optimal solution of the p-facility location problem on T^* as well, since $f_{T^*}(X_N^*) \le f_{T^*}^*$.

Note that a similar result was derived in [3], but only for a single center problem. From the last result of the theorem we conclude that problem (1) can be solved by enumerating all trees from $ST(N)$, but this is computationally a very expensive option due to the high number of spanning trees. Namely, the following fact is well known (see [5]): let I be the incidence matrix of a graph G with one row removed, (i.e. with $n-1$ independent rows), and I^t be the transpose of I, then the determinant $| I \cdot I^t |$ gives the number of distinct spanning trees of G. If G is a complete graph of n vertices, the number of distinct spanning tree is n^{n-2} (see [1]). In general, if a network N has m edges, the number of spanning trees is of order $O\binom{m}{n-1}$.

3. HEURISTIC

However, our result can be used to get some *alternating* local search heuristic method. We call it *TreeAlt* as it alternates the location obtained in the tree, in each iteration.

In step 1 of *TreeAlt* algorithm (see Fig. 1), the initial spanning tree T_0 can be obtained by Prim's method [11]. Then F_0 can be obtained by deleting $p-1$ edges with the largest lengths from T_0. Another possibility is to obtain a shortest path tree of any set of p facility points; i.e. $T_0 \in T(X_0)$ where $|X_0| = p$.

In the second step we can apply any efficient algorithm for the corresponding p-facility location problem in tree networks. For example, we can apply one of the following algorithms:

Algorithm TreeAlt

Step 1 (*Initialization*):
Let T_0 be an initial spanning tree of N, and let F_0 be a p-forest with $F_0 \subseteq T_0$. Set $f_0 = \infty$ and $i \leftarrow 1$.

Step 2 (*Location on the tree*):
Let $X_i = X^*_{T_{i-1}}$ be the optimal p-facility location in tree network T_{i-1}.

Step 3 (*Allocation*):
Allocate each user vertex to its closest facility with respect to N, to get spanning forest F_i. Compute $f_i = f_{T_i}(X_i)$.

Step 4 (*Termination*):
If $f_i = f_{i-1}$, stop. Otherwise, let $T_{i+1} \in T(X_i)$ be spanning tree obtained by adding to F_{i+1} the $p-1$ edges with the smallest lengths that do not make a cycle. Set $i \leftarrow i+1$ and go to Step 2.

Figure 1: Algorithm TreeAlt

- An $O(p\ n^2)$ algorithm for the p-median on a tree [13].
- An $O(n\ \log^2 n)$ algorithm for the p-center on a tree [12].
- An $O(p\ n^6)$ algorithm for the p-centdian on a tree [14] [10].

In the allocation step (Step 3), we use Dijkstra method [2] in order to find the shortest path between each user and its closest facility. If a dummy vertex v_0 joined to every point of X with an edge of length 0 then the Dijkstra algorithm applied to v_0 provides, after eliminating the new edges, the p-forest F_i in $O(m\ \log n)$ time.

Finally, note that we repeat Steps 2 and 3 until there is no improvement in the objective function value. Thus, our method converges to a local minimum.

To illustrate the algorithm, consider the 2-center problem in the network shown in Figure 2 where all the weights are equal to 1. Take as initial tree T_0, the minimal spanning tree consisting of the edges: $[v_1,v_2]$, $[v_2,v_5]$, $[v_5,v_7]$, $[v_6,v_3]$, $[v_1,v_4]$ and $[v_1,v_3]$. Thus $F_0 = T_0 \setminus [v_1,v_3]$. Then X_0 is a 2-center in T_0; e.g. v_2 and the point on $[v_3,v_6]$ at distance 1.5 from v_3. Now F_1 consists of the trees $\{ [v_5,v_2], [v_5,v_7], [v_5,v_4] \}$ and $\{ [v_3,v_1], [v_3,v_6] \}$. Thus $T_1 = F_1 \cup [v_1,v_2]$. Now X_1 is a 2-center in T_1; e.g. the point on $[v_3,v_1]$ at distance 1 from v_3 and the point on $[v_5,v_4]$ at distance 1 from v_5. Finally, F_2 is again F_1 and the algorithm stops.

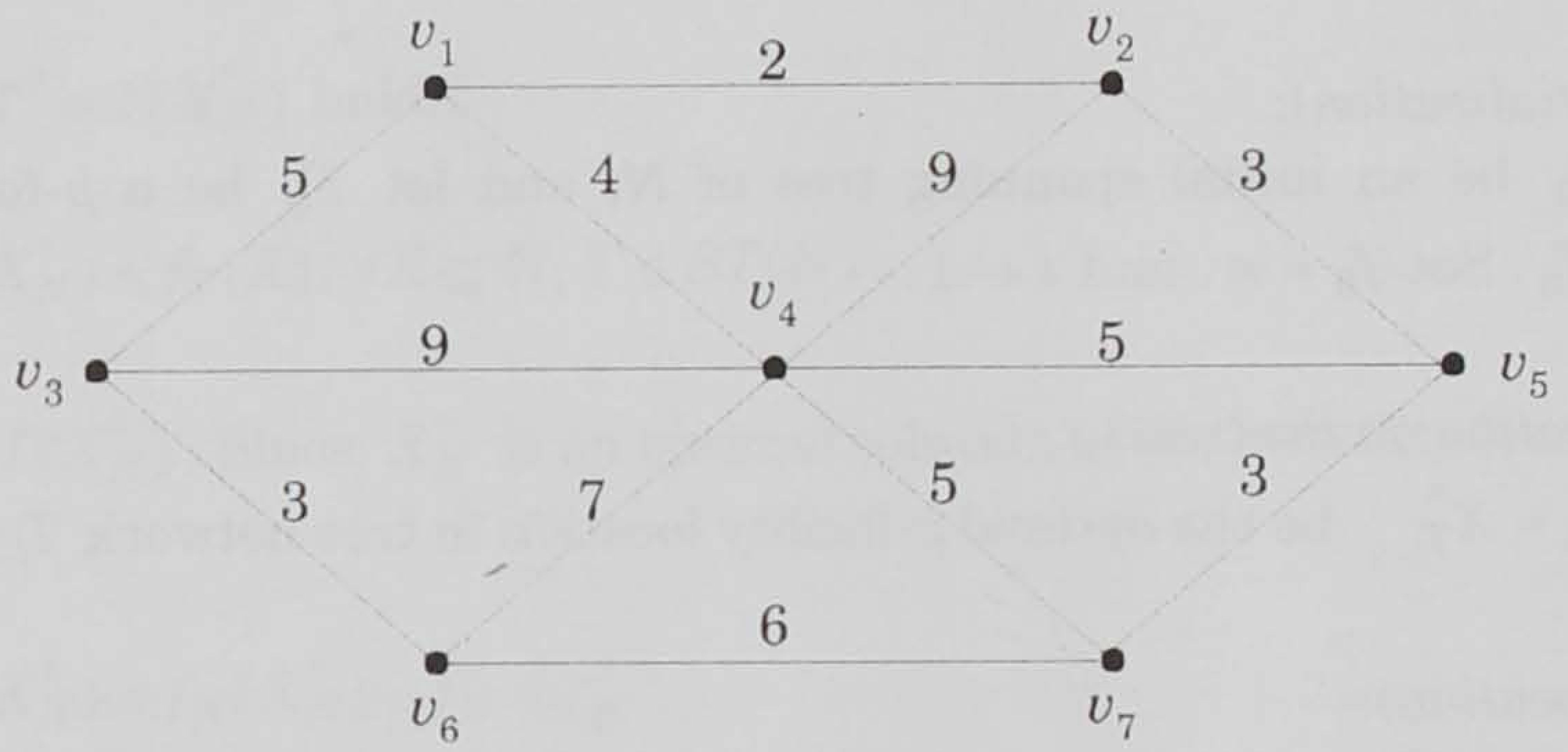

Figure 2: Example

Heuristic *TreeAlt* alternates location and allocation solutions. Another method of this type is suggested in [6], where *p* single facility location problems are solved exactly in the *location step*. We instead solve the *p*-facility problem exactly on the tree.

Since *TreeAlt* is a descent local search heuristic, it can be used as part of some general heuristic method such as Multi Start search [8] or Variable neighborhood search [7]. In Step 1 of the *TreeAlt* algorithm, the initial spanning tree can be obtained at random by a modified version of Prim's method. With this modification, the edge that does not produce a cycle chosen at random is not always added to the current tree. Another possibility is to choose tree $T_0 \in T(X_0)$ for a random set X_0 of *p* points.

REFERENCES

[1] Cayley, A., *Collected Papers*, Quart. Jl. of Mathematics, 13, Cambridge, 1897.

[2] Dijkstra, E. W., "A note on two problems in connection with graphs", *Numer. Math.*, 1 (1959) 269-271.

[3] Dearing, P.M., and Francis, R.L., "A minimax location problem on a network", *Transportation Science*, 8 (1974) 333-343.

[4] Dearing, P.M., Francis, R.L., and Lowe, T.J., "Convex location problems on tree networks", *Operation Research*, 24 (1976) 628-642.

[5] Christofides, N., *Graph Theory: an Algorithms Approach*, Academic Press, New York, 1975.

[6] Maranzana, F.E., "On the location of supply points to minimize transportation cost", *Operation Research Quarterly*, 12 (1964) 138-139.

[7] Mladenović, N., and Hansen, P., "Variable Neighborhood Search", *Computers and Operations Research*, 24-11 (1997) 1097-1100.

[8] Moreno, J.A., Mladenović, N., and Moreno-Vega, J.M., "An statistical analysis of strategies for multistart heuristic searches for p-facility location-allocation problems", *Eight Meeting of the EWG on Locational Analysis*, Lambrecht, Germany, 1995.

[9] Love, R.F., Morris, J.G., and Wesolowsky, G.O., *Facilities Location*, North-Holland, 1988.

[10] Pérez-Brito, D., Moreno-Pérez, J.A., Rodrìguez-Martìn, I., "Finite dominating set for the p-facility cent-dian network location problem", *Studies in Location Ánalysis,* 1997, (forthcoming).

[11] Prim, R. C., "Shortest connection networks and some generalizations", *Bell System Tech. J.*, 36 (1957) 1389-1401.

[12] Tamir, A., and Zemel, E., "Locating centers on tree with discontinues supply and demand regions", *Mathematics of Operation Research*, 7 (1982) 183-197.

[13] Tamir, A., "An $O(pn^2)$ algorithm for the p-median and related problem on tree graphs", *Operations Research Letters*, 19 (1996) 59-64.

[14] Tamir, A., Pérez-Brito, D., and Moreno-Pérez, J.A., "A polynomial algorithm for the p-centdian problem on a tree", *Networks,* 1997, (forthcoming).