A Polynomial Algorithm for the p-Centdian Problem on a Tree

Arie Tamir,[1] Dionisio Pérez-Brito,[2] José A. Moreno-Pérez[2]

[1] Department of Statistics and Operations Research, School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv 69978, Israel

[2] Dpto. Estadística, Investigación Operativa y Computación, University of La Laguna, 38271 La Laguna, Tenerife, Spain

Received 11 February 1997; accepted 13 March 1998

Abstract: The most common problems studied in network location theory are the p-median and the p-center models. The p-median problem on a network is concerned with the location of p points (medians) on the network, such that the total (weighted) distance of all the nodes to their respective nearest points is minimized. The p-center problem is concerned with the location of p-points (centers) on the network, such that the maximum (weighted) distance of all the nodes to their respective nearest points is minimized. To capture more real-world problems and obtain a good way to trade-off minisum (efficiency) and minimax (equity) approaches, Halpern introduced the centdian model, where the objective is to minimize a convex combination of the objective functions of the center and the median problems. In this paper, we studied the p-centdian problem on tree networks and present the first polynomial time algorithm for this problem. © 1998 John Wiley & Sons, Inc. Networks 32: 255–262, 1998

1. INTRODUCTION

In a typical location problem on a network there is a set of customers located at some points on the network. The goal is to locate new facilities (servers) on the network in order to minimize the cost of serving the customers. In most location models, this cost is assumed to be a monotone nondecreasing function of the distances between the customers and the servers. The most fundamental and common problems studied in network location theory are the p-median and the p-center models. The underlying assumption in both models is that the servers are identical and uncapacitated, and as a result, each customer will be served by the nearest server. The p-median problem is concerned with the location of p servers (medians), such that the total (weighted) distance of all the customers to their respective nearest servers is minimized. The p-center problem is to locate p servers (centers) such that the maximum (weighted) distance of all customers to their respective servers is minimized. To capture more real-world problems and obtain a good way to trade-off the minimum (efficiency) and minimax (equity) approaches of the p-median and the p-center problems, Halpern [8–10] suggested to consider a convex combination of the center and median objective functions, which he labeled the centdian function. More recently, Carrizosa et al. [3] presented an axiomatic approach justifying the use of the centdian criterion.

The p-median and the p-center models have been stud-

Correspondence to: A. Tamir; e-mail: atamir@math.tau.ac.il

Contract grant sponsor: DGICYT; contract grant number: PB95-1237-C03-02.

CCC 0028-3045/98/040255-08

ied extensively from both the theoretical and algorithmic points of view. (We refer to Mirchandani and Francis [16] and the references cited there for the major results on these models.) Nevertheless, there are only very few algorithmic results on the p-centdian problem. Several complexity results on the p-centdian problem, like NP-hardness on general networks, follow directly from the fact that the problem is a generalization of the median and the center problems.

In this paper, we focus on the p-centdian problem on tree networks. Our main contributions are the identification of a set of points of a polynomial size, which is guaranteed to contain an optimal solution, and a polynomial time algorithm to find an optimal solution. To the best of our knowledge, this is the first polynomial time algorithm for the p-centdian problem on tree networks.

Efficient algorithms for solving certain location problems on tree networks can also be useful in deriving approximate solutions to the same problems on general networks. For example, the recent work of Bartal [1, 2] implies that an algorithm to solve the p-median problem on a tree can be used directly to derive a randomized algorithm with a guaranteed performance ratio for the p-median problem on a general network. His approach can be applied to other problems, including the p-centdian problem.

We start with a formal definition of the model on a tree network. Let $T = (V, E)$ be an undirected tree with a node set $V = \{v_1, \ldots, v_n\}$ and an edge set E. Each edge has a positive length and is assumed to be rectifiable. We refer to interior points on an edge by their distances along the edge from the two nodes of the edge. We let $A(T)$ denote the continuum set of points on the edges of T. The edge lengths induce a distance function on $A(T)$: For any pair of points x, y in $A(T)$, we let $d(x, y)$ denote the length of $P(x, y)$, the unique simple path connecting x and y. Also, for any subset $Y \subseteq A(T)$ and x in $A(T)$, we define $d(x, Y) = d(Y, x) = \text{Infimum } \{d(x, y) | y \in Y\}$. $A(T)$ is a metric space with respect to the above distance function.

Suppose that each node $v_i \in V$ is associated with a pair of nonnegative weights, (u_i, w_i).

Restricting ourselves to tree networks, and using the above notation, we now define the p-center, the p-median, and the p-centdian problems. (Note that the node set V is identified as the set of customers in these problems.)

The (u-weighted) *p-center problem* is to select a subset of p points $X \subseteq A(T)$, a *p-center set,* to minimize the objective $C(X)$, where

$$C(X) = \max\{u_i d(X, v_i) | i = 1, \ldots, n\}.$$

The (w-weighted) *p-median problem* is to select a subset of p points $X \subseteq A(T)$, a *p-median set,* to minimize the objective $M(X)$, where

$$M(X) = \sum \{w_i d(X, v_i) | i = 1, \ldots, n\}.$$

With our notation the *p-centdian problem* on the tree T is to find a set of p points $X \subseteq A(T)$, a *p-centdian set,* minimizing the objective

$$C(X) + M(X).$$

Efficient algorithms to solve both the p-center and the p-median problems on trees are well known. The most efficient algorithm known for the weighted p-center problem is the $O(n \log^2 n \log\log n)$ algorithm of Megiddo and Tamir [15]. This algorithm can be further improved to $O(n \log^2 n)$ by using the modification in Cole [4]. [For the unweighted case, i.e., when $u_i = 1$, $i = 1, \ldots, n$, Frederickson [6] developed an optimal $O(n)$ time algorithm.]

The most efficient algorithm known for the weighted p-median problem is the $O(pn^2)$ algorithm in Tamir [19]. No polynomial time algorithms for the p-centdian problem on a tree, for a general p, have been reported in the literature. Efficient algorithms for the case $p = 1$ were reported in Halpern [8] and Handler [11].

We present the first polynomial algorithm for the p-centdian problem on trees. The complexity of our algorithm is $O(pn^6)$, for $p \geq 6$, and $O(n^p)$, for $p < 6$. We also consider the discrete version of the model where the p selected points must be nodes and show that the complexity bound for this case reduces to $O(pn^4)$, for $p \geq 4$, and $O(n^p)$, for $p < 4$.

2. SOLVING THE p-CENTDIAN PROBLEM

In this section, we outline the general solution approach which leads to a polynomial time algorithm. For each nonnegative real r, define the *r-restricted p-median* problem on the tree T to be the problem of minimizing the sum of w-weighted distances of the nodes to a p-median set, given that the u-weighted distance of each node to the p-median set is at most r. (Note that unlike the unrestricted p-median problem, where there is always an optimal p-median set which consists of nodes only, the p-median set for the restricted problem is not necessarily a subset of nodes.) Let $m(r)$ denote the optimal value of the above r-restricted problem. Then,

$$m(r) = \min_{\{X | X \subseteq A(T), |X| = p\}} \{M(X) | u_i d(X, v_i) \leq r, i = 1, \ldots, n\}.$$

For each nonnegative real r, let $g(r) = r + m(r)$. The p-centdian problem can now be reformulated as

$$\min_r \{g(r)\}.$$

Let r^* be a minimizer of the function $g(r)$. The general strategy that we apply is as follows: We will first identify explicitly a set R of $O(n^3)$ cardinality, called a finite dominating set, which includes r^*. To find r^*, it will then suffice to compute $g(r)$ for all values of $r \in R$ and evaluate the minimum of the function $g(r)$ over R. [In the last section, we will show that the function $g(r)$ is not necessarily convex or unimodal, and, therefore, a binary search on the set R to locate r^* is not directly applicable.] Thus, to obtain a polynomial time scheme with the above approach, we will need a polynomial algorithm to compute the function $g(r)$ [equivalently $m(r)$] for any specified value of r. Indeed, we will show that for any value of r, $m(r)$ can be computed directly by the algorithm in Tamir [19] in $O(pn^4)$ time. We will then show how to modify and adapt the algorithm in Tamir [19] to compute $m(r)$ in $O(pn^3)$ time. This will result in an $O(pn^6)$ algorithm for solving the p-centdian problem on tree networks.

2.1. A Finite Dominating Set

In this section, we identify a set R of $O(n^3)$ cardinality containing r^*, the optimal solution of the p-centdian problem.

Theorem 2.1. *Let r^* be a minimizer of the function $g(r) = r + m(r)$. Then, r^* is an element in the set*

$$R = R_1 \cup R_2 \cup R_3,$$

where

$$R_1 = \{u_i d(v_i, v_j) | v_i, v_j \in V\},$$

$$R_2 = \{d(v_i, v_j)/(1/u_i + 1/u_j) | v_i, v_j \in V\},$$

$$R_3 = \{(d(v_j, v_k) - d(v_i, v_k))/(1/u_j - 1/u_i) | v_i, v_j, v_k \in V, v_k \in P(v_i, v_j)\}.$$

Proof. Let X' be an optimal solution to the p-centdian problem. Then, there is a partition of V into p subtrees, $T_1, \ldots, T_p$, where all the nodes in each subtree T_j are served by the same point, say x_j', in X', that is, $d(v_i, X') = d(v_i, x_j')$, for each node v_i in T_j. In particular, at optimality, the subproblem corresponding to T_j, $j = 1, \ldots, p$, is an r^*-restricted 1-median problem. For a real nonnegative r, let $m_j(r)$ be the objective value of the r-restricted 1-median subproblem, defined on T_j.

We show that $m_j(r)$ is a convex, decreasing, piecewise linear function of r, with breakpoints in the set R defined above. Let x_j^* be the (unique) solution to the u-weighted 1-center problem on T_j, and let y_j^*, a node in T_j, be the solution to the (unrestricted) w-weighted 1-median problem on T_j, which is closest to x_j^*. We view the path $P(x_j^*, y_j^*)$ connecting x_j^* and y_j^* as a line segment of length $d(x_j^*, y_j^*)$ and consider the u-weighted 1-center function, $C_j(x_j)$, and the w-weighted (unrestricted) 1-median function, $M_j(x_j)$, defined on this path (segment). For convenience, we assume without loss of generality that x_j is a real variable restricted to the interval (segment) $[0, d(x_j^*, y_j^*)]$, where $x_j = 0$ corresponds to the 1-centerof T_j, x_j^*, and $x_j = d(x_j^*, y_j^*)$ corresponds to the 1- median of T_j, y_j^*.

For each node v_i of T_j, the distance function $d(v_i, x_j)$ is a convex piecewise linear function with at most one breakpoint. Therefore, from its definition, the median function, $M_j(x_j)$, is a convex, decreasing, piecewise linear function of its (one-dimensional) argument x_j. Its breakpoints correspond to the nodes of T_j on the path $P(x_j^*, y_j^*)$. (See also [5].) Similarly, the center function, $C_j(x_j)$, is a convex, increasing, piecewise linear function of x_j. Each breakpoint of this function is a point x on $P(x_j^*, y_j^*)$ whose u-weighted distances from two nodes of T_j, say v_i and v_t, are equal, that is, $u_i d(v_i, x) = u_t d(v_t, x)$.

Let r_j' and r_j'' denote, respectively, the optimal values of $C_j(x_j)$ at x_j^* and y_j^*. For each value of r, in the interval $[r_j', r_j'']$, let $x_j(r)$ denote the unique solution to the equation $C_j(x_j) = r$. In particular, $x_j(r)$ is a concave, increasing, piecewise linear function of r. [The concavity follows directly from the convexity of $C_j(x_j)$.] From the above, if r is a breakpoint of $x_j(r)$, then there exist a pair of nodes of T_j, say v_i and v_t, such that $u_i d(v_i, x_j(r)) = u_t d(v_t, x_j(r)) = r$.

For each value of r, in $[r_j', r_j'']$, the solution to the r-restricted 1-median problem is attained at the closest point to y_j^* in the set $\{z | z \in A(T_j) | u_i d(v_i, z) \le r, \forall v_i \in T_j\}$. Since the latter set contains x_j^*, the optimal solution to the r-restricted 1-median problem on T_j is attained at the point x_j on $P(x_j^*, y_j^*)$ for which $C_j(x_j) = r$, i.e., at $x_j(r)$. Therefore, $m_j(r) = M_j(x_j(r))$. ($m_j(r) = \infty$ for $r < r_j'$,and $m_j(r) = m_j(r_j'')$ for $r \ge r_j''$).

From the above discussion, it follows that $m_j(r)$ is a convex, decreasing and piecewise linear function of r. [The convexity of $m_j(r)$ follows directly from the convexity of $M_j(x_j)$, the concavity of $x_j(r)$, and the monotonicity of $M_j(x_j)$.] Moreover, since the functions $M_j(x_j)$ and $x_j(r)$ are piecewise linear, each breakpoint of $m_j(r)$ must correspond to a breakpoint of $M_j(x_j)$ or $x_j(r)$. Specifically, if r is a breakpoint of $m_j(r)$, then $x_j(r)$ is either a breakpoint of the median function $M_j(x_j)$ [i.e., $x_j(r)$ is a node of T_j on $P(x_j^*, y_j^*)$] or $x_j(r)$ is a breakpoint of the center function $C_j(x_j)$. If $x_j(r)$ is a node of T_j, say v_k, then from the definition of $C_j(x_j)$, there exists some node, say v_i of

T_j, such that $r = u_i d(v_i, v_k)$. Thus, in this case, $r \in R_1$. If $x_j(r)$ is a breakpoint of $C_j(x_j)$, then there exist a pair of nodes in T_j, say v_i and v_t, such that $u_i d(v_i, x_j(r)) = u_t d(v_t, x_j(r)) = r$. If $x_j(r)$ is on $P(v_i, v_t)$, the path connecting v_i with v_t, then it is easy to confirm that $r = d(v_i, v_t)/(1/u_i + 1/u_t)$ and, therefore, $r \in R_2$. Otherwise, let v_k be the closest point to $x_j(r)$ on $P(v_i, v_t)$. In this case, the equations $u_i d(v_i, x_j(r)) = u_t d(v_t, x_j(r)) = r$ imply that $r = (d(v_t, v_k) - d(v_i, v_k))/(1/u_t - 1/u_i)$ and, therefore, $r \in R_3$.

We have shown that for each $j = 1, \ldots, p$ the breakpoints of the function $m_j(r)$ belong to the set R defined in the statement of the theorem. To conclude the proof, observe that r^*, the optimal solution to the p-centdian problem, is a minimizer of the function $g'(r) = r + \Sigma_{\{j=1,\ldots,p\}} m_j(r)$.

$g'(r)$ is a convex piecewise linear function, and its breakpoints coincide with the breakpoints of the functions $m_j(r)$, $j = 1, \ldots, p$. Therefore, the minimum point of $g'(r)$ is attained at one of these breakpoints. This concludes the proof of the theorem. ∎

We note that the specialization of the above theorem for the unweighted case, that is, when $u_i = w_i = 1$, for all $i = 1, \ldots, n$, can be obtained from the results in Pérez-Brito et al. [17, 18].

3. SOLVING THE r-RESTRICTED p-MEDIAN PROBLEM

To solve the p-centdian problem on a tree network T, it will suffice to compute the solution to the r-restricted p-median problem for all values of r in the set R, specified in Theorem 2.1.

We will show how to solve an r-restricted problem in polynomial time, by adapting and modifying the algorithm for the (generalized) p-median model in Tamir [19]. The generalized model is defined as follows:

Each node v_i of the tree T is associated with a real nondecreasing function f_i. (f_i is viewed as a transportation cost function.) The problem is to select a subset S of at most p nodes in V minimizing the objective

$$\sum_{v_i \in V} f_i(d(v_i, S)).$$

(Note that in the above model the p points to be selected are restricted to the node set of the tree.) The algorithm in Tamir [19] solves the above generalized p-median problem in $O(pn^2)$ time. We now show how to formulate the r-restricted p-median problem as an instance of the above-generalized p-median problem.

Let r be a positive real and consider the r-restricted p-median problem. From the nature of the objective function, it is clear that the optimal p-median set can be restricted without loss of generality to the following set $Y(r)$:

$$Y(r) = V \cup \{y \mid y \in A(T),\ u_i d(v_i, y) = r, \text{ for some } v_i \in V\}.$$

The set $Y(r)$ is of $O(n^2)$ cardinality. In Kim et al. [13], it is shown how to compute the set $Y(r)$ and augment its points to the node set of T in $O(n^2)$ time. Let $T(r)$ denote the augmented tree with the node set $Y(r)$. Finally, to formulate the r-restricted p-median problem as the generalized p-median model in Tamir [19], we define the transportation cost functions for all points in $Y(r)$. The transportation cost function of each point in $Y(r)$ which is not a node of the original tree T is the zero function. For each $v_i \in V$, define the transportation cost function f_i by

$$f_i(t) = \begin{cases} w_i t & t \le r \\ \infty & t > r. \end{cases}$$

Solving the r-restricted p-median problem reduces to the solution of the above-generalized p-median problem on the augmented tree $T(r)$. Since $T(r)$ has $O(n^2)$ nodes, the algorithm in Tamir [19] will solve the latter problem in $O(pn^4)$ time.

Since our approach for solving the p-centdian problem on T requires the solution of the r-restricted p-median problem for $O(n^3)$ values of r, the complexity bound is $O(pn^7)$. In the next section, we show how to reduce this bound to $O(pn^6)$, by improving the complexity of the r-restricted p-median problem to $O(pn^3)$.

3.1. An Improved $O(pn^3)$ Algorithm for the r-Restricted p-Median Problem

Suppose now that the given tree $T = (V, E)$ is rooted at some distinguished node, say, v_1. For each pair of nodes v_i, v_j, we say that v_i is a *descendant* of v_j if v_j is on the unique path connecting v_i to the root v_1. If v_i is a descendant of v_j and v_i is connected to v_j with an edge, then v_i is a *child* of v_j and v_j is the (unique) *father* of v_i. If a node has no children it is called a *leaf* of the tree.

As shown in Tamir [19], we can now assume without loss of generality that the original tree is a binary tree, where each nonleaf node v_j has exactly two children, $v_{j(1)}$ and $v_{j(2)}$. The former is called the *left child*, and the latter is the *right child*. For each node v_j, V_j will denote the set of its descendants, and T_j will be the subtree induced by V_j. (v_j is also viewed as the root of T_j.)

In the first preprocessing step, we augment to T the set of points $Y = Y(r)$, defined in the previous section.

(Recall that there is an optimal solution to the r-restricted problem, where all p selected points are in Y. Moreover, since $V \subseteq Y$, we only need to add the points in $Y - V$.) Each point in Y is called a *seminode.* In particular, a node in V is also a seminode. For each node v_j, Y_j will denote the subset of seminodes which have v_j on the path connecting them to the root v_1. As mentioned above, the cardinality of Y is $m = O(n^2)$, and the augmentation of Y to T can be performed in $O(n^2)$ time, as described in Kim et al. [13].

In the second preprocessing step, for each node v_j, we compute and sort the distances from v_j to all seminodes in Y. Let this sequence be denoted by $L_j = \{r_j^1, \ldots, r_j^m\}$, where $r_j^i \le r_j^{i+1}$, $i = 1, \ldots, m - 1$, and $r_j^1 = 0$. For convenience, to handle a degenerate case, where the elements in L_j are not distinct, we assume that there is a one-to-one correspondence between the elements in L_j and the seminodes in Y, such that

1. If $y_k \in Y$ corresponds to r_j^i, then $r_j^i = d(v_j, y_k)$.
2. If y_k and y_t are two distinct seminodes in Y_j, and y_k is on the unique path connecting y_t with the v_j, then the element of L_j representing y_k will precede the one representing y_t. In particular, v_j corresponds to r_j^1.
3. If v_j is not a leaf, and y_k and y_t are both in $V_{j(1)}$ ($V_{j(2)}$), where the element representing y_k in $L_{j(1)}$ ($L_{j(2)}$) precedes the one representing y_t, then the element of L_j representing y_k will precede the one representing y_t.
4. If y_k is in Y_j, y_t is in $Y - Y_j$, and $d(v_j, y_k) = d(v_j, y_t)$, then the element of L_j representing y_k will precede the one representing y_t.

For $i = 1, \ldots, m$, the seminode corresponding to r_j^i is denoted by y_j^i.

We note that the total effort of the second preprocessing step is $O(n^3)$. It can be achieved by using the centroid decomposition approach as in Kim et al. [13] or the procedure described in Tamir [19].

We are now ready to present the "leaves to root" dynamic programming algorithm to solve the r-restricted p-median problem. The algorithm is a modified version of the algorithm in Tamir [19] to solve the generalized p-median problem.

For each node v_j, an integer $q = 1, \ldots, p$, and $r_j^i \in L_j$, let $G(v_j, q, r_j^i)$ be the optimal value of the subproblem defined on the subtree T_j, given that a total of at least 1 and at most q seminodes (service centers) can be selected in T_j and that at least one of them has to be in $\{y_j^1, y_j^2, \ldots, y_j^i\} \cap Y_j$. (In the above subproblem, we implicitly assume no interaction between the seminodes in T_j and the rest of the seminodes in T.) The function $G(v_j, q, r)$ is computed only for $q \le |V_j|$. Also, for each node v_j, we define

$$G(v_j, 0, r) = \sum_{v_i \in V_j} f_i(\infty).$$

Similarly, for each node v_j and an integer $q = 0, 1, \ldots, p$, we define $F(v_j, q, r)$ to be the optimal value of the subproblem defined on the subtree T_j, under the following two constraints:

1. A total of at most q seminodes can be selected in Y_j.
2. There are already some selected seminodes (service centers) in $Y - Y_j$, and the closest among them to v_j is at a distance of exactly r from v_j.

[$F(v_j, q, r)$ is computed only for $q \le |V_j|$, and $r = r_j^i$, where r_j^i corresponds to a seminode y_j^i in $Y - Y_j$.]

To motivate the above definitions, we note that if all the elements in L_j are distinct, then $G(v_j, q, r_j^i)$ is the optimal value of the subproblem defined on T_j, given that at least 1 and at most q seminodes are selected in T_j, and the closest among them to v_j is at a distance of at most r_j^i from v_j. The conditions on L_j, required above, ensure that the same interpretation of $G(v_j, q, r_j^i)$ can be made for the "distinct" elements of L_j, that is, the elements r_j^i, satisfying $r_j^i < r_j^{i+1}$.

The algorithm defines the functions G and F at all leaves of T and then recursively, proceeding from the leaves to the root, computes these functions at all nodes of T. The optimal value of the problem will be given by $\min\{G(v_1, p, r_1^m), G(v_1, 0, r_1^m)\}$, where v_1 is the root of the tree.

Let v_j be a leaf of T. Then,

$$G(v_j, 1, r_j^i) = 0, \quad i = 1, \ldots, m.$$

For each $i = 1, \ldots, m$, such that $y_j^i \in Y - Y_j$,

$$F(v_j, 0, r_j^i) = f_j(d(v_j, y_j^i))$$

and

$$F(v_j, 1, r_j^i) = \min\{F(v_j, 0, r_j^i), G(v_j, 1, r_j^i)\}.$$

Let v_j be a nonleaf node in V, and let $v_{j(1)}$ and $v_{j(2)}$ be its left and right children, respectively. The element r_j^1 corresponds to $y_j^1 = v_j$, which, in turn, corresponds to a pair of elements, say $r_{j(1)}^k$ and $r_{j(2)}^l$ in $L_{j(1)}$ and $L_{j(2)}$, respectively. Therefore,

$$G(v_j, q, r_j^1) = \min_{\substack{q_1+q_2=q-1 \\ q_1 \le |V_{j(1)}| \\ q_2 \le |V_{j(2)}|}} \{F(v_{j(1)}, q_1, r_{j(1)}^k) + F(v_{j(2)}, q_2, r_{j(2)}^l)\}.$$

Generally, for $i = 2, \ldots, m$, consider r_j^i. If r_j^i corresponds to a seminode $y_j^i \in Y - Y_j$, then

$$G(v_j, q, r_j^i) = G(v_j, q, r_j^{i-1}).$$

If $y_j^i \in Y_j$, then y_j^i corresponds to some element, say $r_{j(1)}^k$ in $L_{j(1)}$, and to some element, say $r_{j(2)}^l$ in $L_{j(2)}$. If $y_j^i \in Y_{j(1)}$, then

$$G(v_j, q, r_j^i) = \min\{G(v_j, q, r_j^{i-1}), f_j(r_j^i) + \min_{\substack{q_1+q_2=q \\ 1\le q_1\le |V_{j(1)}| \\ q_2 \le |V_{j(2)}|}} \{G(v_{j(1)}, q_1, r_{j(1)}^k) + F(v_{j(2)}, q_2, r_{j(2)}^l)\}\}.$$

If y_j^i is on the edge $(v_j, v_{j(1)})$ and $y_j^i \neq v_{j(1)}$, then

$$G(v_j, q, r_j^i) = \min\{G(v_j, q, r_j^{i-1}), f_j(r_j^i) + \min_{\substack{q_1+q_2=q-1 \\ q_1\le |V_{j(1)}| \\ q_2 \le |V_{j(2)}|}} \{F(v_{j(1)}, q_1, r_{j(1)}^k) + F(v_{j(2)}, q_2, r_{j(2)}^l)\}\}.$$

If $y_j^i \in Y_{j(2)}$, then

$$G(v_j, q, r_j^i) = \min\{G(v_j, q, r_j^{i-1}), f_j(r_j^i) + \min_{\substack{q_1+q_2=q \\ q_1\le |V_{j(1)}| \\ 1\le q_2 \le |V_{j(2)}|}} \{F(v_{j(1)}, q_1, r_{j(1)}^k) + G(v_{j(2)}, q_2, r_{j(2)}^l)\}\}.$$

If y_j^i is on the edge $(v_j, v_{j(2)})$ and $y_j^i \neq v_{j(2)}$, then

$$G(v_j, q, r_j^i) = \min\{G(v_j, q, r_j^{i-1}), f_j(r_j^i) + \min_{\substack{q_1+q_2=q-1 \\ q_1\le |V_{j(1)}| \\ q_2 \le |V_{j(2)}|}} \{F(v_{j(1)}, q_1, r_{j(1)}^k) + F(v_{j(2)}, q_2, r_{j(2)}^l)\}\}.$$

Having defined the function G at v_j, we can compute the function F at v_j for all relevant arguments. Let y_j^i be a seminode in $Y - Y_j$. Then, y_j^i corresponds to some elements, say $r_{j(1)}^k$ and $r_{j(2)}^l$ in $L_{j(1)}$ and $L_{j(2)}$, respectively. Therefore,

$$F(v_j, q, r_j^i) = \min\{G(v_j, q, r_j^i), f_j(r_j^i) + \min_{\substack{q_1+q_2=q \\ q_1\le |V_{j(1)}| \\ q_2 \le |V_{j(2)}|}} \{F(v_{j(1)}, q_1, r_{j(1)}^k) + F(v_{j(2)}, q_2, r_{j(2)}^l)\}\}.$$

3.2. Complexity of the Algorithm

It follows directly from the recursive equations that the total effort to compute the functions G and F at a given node v_j for all relevant values of q and r is $O(n^2 \min\{(|V_{j(1)}|, p)\}\min\{(|V_{j(2)}|, p)\})$. Therefore, the total effort of the algorithm is clearly $O(p^2n^3)$. However, it is easy to verify that the finer and detailed analysis in Tamir [19] can also be applied to the above model to improve the bound to $O(pn^3)$.

4. AN ALTERNATIVE ALGORITHM FOR THE p-CENTDIAN PROBLEM

We have presented above an $O(pn^6)$ algorithm for solving the p-centdian problem on a tree graph. When p is small, there is an alternative scheme, based on the proof approach of Theorem 2.1. We will show that this approach yields an $O(n^p)$ complexity bound for each fixed value of p.

Consider all the $O(n^{p-1})$ partitions of the tree T into p subtrees, obtained by deleting a subset of $p - 1$ edges of T. To achieve the $O(n^p)$ complexity bound, we now show a linear time algorithm to solve the p-centdian problem on a forest consisting of a partition of T into p subtrees, $T_1, \ldots, T_p$.

We adapt the notation used in the proof of Theorem 2.1. For each subtree T_j, let x_j^* be the (unique) solution to the u-weighted 1-center problem on T_j, and let y_j^*, a node in T_j, be the solution to the (unrestricted) w-weighted 1-median problem on T_j, which is closest to x_j^*. We view the path $P(x_j^*, y_j^*)$ connecting x_j^* and y_j^* as a line segment of length $d(x_j^*, y_j^*)$ and consider the u-weighted 1-center function, $C_j(x_j)$, and the w-weighted (unrestricted) 1-median function $M_j(x_j)$, defined on this path (segment). The median function, $M_j(x_j)$, is a convex, decreasing, piecewise linear function of its (one-dimensional) argument x_j. Its breakpoints correspond to the nodes of T_j on the path $P(x_j^*, y_j^*)$. Similarly, the center function, $C_j(x_j)$, is a convex, increasing, piecewise linear function of x_j. Each breakpoint of this function is a point x on $P(x_j^*, y_j^*)$ whose u-weighted distances from two nodes of T_j, say v_i and v_t, are equal, that is, $u_i d(v_i, x) = u_t d(v_t, x)$.

We now formulate the p-centdian problem on the forest consisting of the p subtrees $T_1, \ldots, T_p$ as a linear program expressed in terms of the variables $x_1, \ldots, x_p$, and $n + 1$ auxiliary variables. For convenience, we assume without loss of generality that each real variable x_j, $j = 1, \ldots, p$, is restricted to the interval (segment) $[0, d(x_j^*, y_j^*)]$, where $x_j = 0$ corresponds to the 1-center of T_j, x_j^*, and $x_j = d(x_j^*, y_j^*)$ corresponds to the 1-median of T_j, y_j^*. If the 1-center, x_j^*, is not an original node of T_j, we augment it to the node set of T_j, and let its u and w weights be equal to 0. V^j will denote the node set of T_j. We note that the u-weighted 1-center, x_j^*, can be found in $O(|V^j|)$ time by the algorithm in Megiddo [14].

Similarly, the w-weighted 1-median, y_j^*, can be found in $O(|V^j|)$ time by the algorithm in Goldman [7]. Thus, the total time to find $\{x_1^*, \ldots, x_p^*\}$, and $\{y_1^*, \ldots, y_p^*\}$ is $O(n)$.

For each node $v_i \in V^j$, on the path $P(x_j^*, y_j^*)$, we let a_i be the respective value of the real variable x_j, that is, $a_i = d(v_i, x_j^*)$. Next, for each node $v_k \in V^j$, let $v_{i(k)}$ be the closest node to v_k on $P(x_j^*, y_j^*)$ and let $b_k = d(v_k, v_{i(k)})$.

For each subtree $T_j, j = 1, \ldots, p$, the 1-center function on the path $P(x_j^*, y_j^*)$ is defined by

$$C_j(x_j) = \max_{v_k \in V^j} \{u_k(|x_j - a_{i(k)}| + b_k)\}.$$

Similarly, the 1-median function is defined by

$$M_j(x_j) = \sum_{v_k \in V^j} \{w_k(|x_j - a_{i(k)}| + b_k)\}.$$

The p-centdian problem on the above forest is to find reals, $x_1, \ldots, x_p$, minimizing the objective

$$\max_{\{j=1,\ldots,p\}} C_j(x_j) + \sum_{\{j=1,\ldots,p\}} M_j(x_j).$$

A reformulation of the problem can be obtained by associating a real variable z_k with each node v_k of T. The problem is to find real variables, $x_1, \ldots, x_p, z_1, \ldots, z_n$ and r, minimizing $\{r + \Sigma_{\{i=1,\ldots,n\}} z_i\}$, subject to the constraints

$$r \geq u_k(|x_j - a_{i(k)}| + b_k),$$

$$z_k \geq w_k(|x_j - a_{i(k)}| + b_k),$$

for all $v_k \in V^j, j = 1, \ldots, p$.

The above problem can now be expressed as the following linear program:

$$\min\{r + \sum_{\{i=1,\ldots,n\}} z_i\},$$

subject to the constraints

$$r \geq u_k((x_j - a_{i(k)}) + b_k),$$

$$r \geq u_k(-(x_j - a_{i(k)}) + b_k),$$

$$z_k \geq w_k((x_j - a_{i(k)}) + b_k),$$

$$z_k \geq w_k(-(x_j - a_{i(k)}) + b_k),$$

for all $v_k \in V^j, j = 1, \ldots, p$.

The above linear program is an instance of the model presented in Zemel [20]. Specifically, it is a special case of the dual of the p-dimensional Multiple Choice Linear Programming Problem discussed in Section 3 of [20]. Thus, for any fixed value of p, the above linear program can be solved in $O(n)$ time.

To conclude, we have shown that for each fixed value of p the p-centdian problem on T can be solved in $O(n^p)$ time. This latter bound dominates the bound $O(pn^6)$ for $p < 6$.

5. FINAL REMARKS

First, we note that the discrete model, where the p-centdian set is restricted to the node set V, is solvable in $O(pn^4)$ time. This follows directly from the facts that in this case the set R_1 in Theorem 2.1 must include the optimal value r^* and the discrete r-restricted p-median problem is solvable in $O(pn^2)$ time by the algorithm in Tamir [19]. We can also adapt the alternative algorithm of Section 4 to solve the discrete model in $O(n^p)$ time for $p < 4$. In the discrete case, the algorithm in [19] permits us to introduce setup costs for the p servers into the objective function without affecting the complexity bound.

The (continuous) model with $u_i = 1, i = 1, \ldots, n$, is solvable in $O(pn^5)$ time. This follows from the fact that the set $R_1 \cup R_2$ in Theorem 2.1 includes the optimal value r^*, that is, $R = R_1 \cup R_2$ in this case.

In the case of a path network, the complexity of the (continuous and discrete) p-centdian problems will be reduced to $O(pn^3)$. To obtain this bound, note that in the case of a path the proof of Theorem 2.1 implies that the node v_k, in the definition of the set R_3 in Theorem 2.1, must coincide with either v_i or v_j. Therefore, $|R_3| = O(n^2)$ in this case. Moreover, the r-restricted p-median problem can be solved in $O(pn)$ time as follows:

Each node contributes at most 2 points on the path whose distance from that node is equal to r. Thus, it is sufficient to consider a discrete problem with at most $3n$ points. This discrete problem is solvable in $O(pn)$ time by the results in Hassin and Tamir [12].

In view of the relatively high polynomial bound of $O(pn^6)$, a relevant question is whether the function $g(r)$, defined in Section 2, possesses some properties that will enable us to avoid its explicit computation for all values of r in the set R, defined in Theorem 2.1. This would have been possible if we could show that the function $g(r)$ is convex or unimodal. From the discussion in Sections 2 and 4, it follows that $g(r)$ is piecewise linear and all its local minimum breakpoints are in R. Nevertheless, $g(r)$ is neither convex nor unimodal, as illustrated by the following example:

Let $p = 2$. Consider a path with 4 nodes v_1, v_2, v_3, v_4, where the distance between adjacent nodes $(v_i, v_{(i+1)})$ is equal to 1. Suppose that $u_i = 1, i = 1, \ldots, 4$. Set $w_1 = w_2 = 4$ and $w_3 = w_4 = 1$. We have $g(r) = 5 + r$ for $\frac{1}{2} \leq r \leq 1$, $g(r) = 7 - r$ for $1 \leq r \leq 2$, and $g(r) = 3 + r$ for $r \geq 2$.

The research of two of the authors (D.P.-B. and J.A.M.-P.) was partially supported by the DGICYT project PB95-1237-C03-02.

REFERENCES

[1] Y. Bartal, Probabilistic approximation of metric spaces and its algorithmic applications. Technical Report, International Computer Science Institute, Berkeley, CA (1997).

[2] Y. Bartal, Private communication (Sept. 1997).

[3] E. J. Carrizosa, E. Conde, F. R. Fernández, and J. Puerto, An axiomatic approach to the cent-dian criterion. *Loc. Sci.* **3** (1994) 1–7.

[4] R. Cole, Slowing down sorting networks to obtain faster sorting algorithms. *J. ACM* **34** (1987) 200–208.

[5] P. M. Dearing, R. L. Francis, and T. J. Lowe, Convex location problems on tree networks. *Oper. Res.* **24** (1976) 628–642.

[6] G. N. Frederickson, Optimal algorithms for partitioning trees and locating p-centers in trees. Technical Report, Department of Computer Science, Purdue University (1990).

[7] A. J. Goldman, Optimal center location in simple networks. *Trans. Sci.* **5** (1971) 212–221.

[8] J. Halpern, The location of a cent-dian convex combination on an undirected tree. *J. Reg. Sci.* **16** (1976) 237–245.

[9] J. Halpern, Finding minimal center-median convex combination (cent-dian) of a graph. *Mgmt. Sci.* **24** (1978) 535–544.

[10] J. Halpern, Duality in the cent-dian of a graph. *Oper. Res.* **28** (1980) 722–735.

[11] G. Y. Handler, Medi-centers of a tree. *Trans. Sci.* **19** (1985) 246–260.

[12] R. Hassin and A. Tamir, Improved complexity bounds for location problems on the real line. *Oper. Res. Lett.* **10** (1991) 395–402.

[13] T. U. Kim, T. J. Lowe, A. Tamir, and J. E. Ward, On the location of a tree-shaped facility. *Networks* **28** (1996) 167–175.

[14] N. Megiddo, Linear-time algorithms for linear programing in R^3 and related problems. *SIAM J. Comput.* **12** (1983) 751–758.

[15] N. Megiddo and A. Tamir, New results on the complexity of p-center problems. *SIAM J. Comput.* **12** (1983) 751–758.

[16] P. B. Mirchandani and R. L. Francis, Eds., *Discrete Location Theory*. Wiley, New York (1990).

[17] D. Pérez-Brito, J. A. Moreno-Pérez, and I. Rodríguez-Martín, The 2-facility centdian network problem. *Loc. Sci.*, to appear.

[18] D. Pérez-Brito, J. A. Moreno-Pérez, and I. Rodríguez-Martín, The finite dominating set for the p-facility centdian network location problem. *Stud. Loc. Anal.*, **11** (1997) 27–40.

[19] A. Tamir, An $O(pn^2)$ algorithm for the p-median and related problems on tree graphs. *Oper. Res. Lett.* **19** (1996) 59–64.

[20] E. Zemel, An $O(n)$ algorithm for the linear multiple choice knapsack problem and related problems. *Info. Process. Lett.* **18** (1984) 123–128.