Math. Program., Ser. A (2004)

Digital Object Identifier (DOI) 10.1007/s10107-004-0547-2

J. Puerto · A. Tamir

Locating tree-shaped facilities using the ordered median objective

Received: November 19, 2002 / Accepted: June 15, 2004
Published online: 1 October 2004 –

Abstract. In this paper we consider the location of a tree-shaped facility S on a tree network, using the ordered median function of the weighted distances to represent the total transportation cost objective. This function unifies and generalizes the most common criteria used in location modeling, e.g., median and center. If there are n demand points at the nodes of the tree $T = (V, E)$, this function is characterized by a sequence of reals, $\Lambda = (\lambda_1, \dots, \lambda_n)$, satisfying $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n \geq 0$. For a given subtree S let $X(S) = \{x_1, \dots, x_n\}$ be the set of weighted distances of the n demand points from S. The value of the ordered median objective at S is obtained as follows: Sort the n elements in $X(S)$ in nonincreasing order, and then compute the scalar product of the sorted list with the sequence Λ. Two models are discussed. In the tactical model, there is an explicit bound L on the length of the subtree, and the goal is to select a subtree of size L, which minimizes the above transportation cost function. In the strategic model the length of the subtree is variable, and the objective is to minimize the sum of the transportation cost and the length of the subtree. We consider both discrete and continuous versions of the tactical and the strategic models. We note that the discrete tactical problem is NP-hard, and we solve the continuous tactical problem in polynomial time using a Linear Programming (LP) approach. We also prove submodularity properties for the strategic problem. These properties allow us to solve the discrete strategic version in strongly polynomial time. Moreover the continuous version is also solved via LP. For the special case of the k-centrum objective we obtain improved algorithmic results using a Dynamic Programming (DP) algorithm and discretization and nestedness results.

1. Introduction

In a typical location problem there is a set of demand points embedded in some metric space and the objective is to locate a specified number of servers optimizing some criterion, which usually depends on the distances between the demand points and their respective servers. Traditionally most papers focus on location problems where a server (facility) is representable by a point in the metric space. However, in the last years there has been a growing interest in studying the location of connected structures, which can not be represented by isolated points in the space. These problems were motivated by concrete decision problems related to routing and network design. For instance, in order to improve the mobility of the population and reduce traffic congestion, many existing rapid transit networks are being updated by extending or adding lines. These lines can be viewed as new facilites. Studies on location of connected structures (which are called *extensive* facilities), appeared already in the early eighties [22], [13], [30], [21], [20], [2]). [12] focused on the complexity of solving many versions of location problems of extensive facilities. Almost all the location problems of extensive facilities that have

J. Puerto: Facultad de Matemáticas, Universidad de Sevilla and Fundacion Centra, Spain

A. Tamir: School of Mathematical Sciences, Tel Aviv University, Israel

been discussed in the literature and shown to be polynomially solvable, are defined on tree networks. (See e.g., [1], [3], [4], [25], [26], [29], [30], [33], [36], [38], [41], [42], and the references therein.) In the case of a metric space induced by a tree network, the extensive facilities are identified as (closed) subtrees. If the leaves of a subtree are restricted to be nodes of the tree, the subtree facility is called *discrete*, otherwise it is *continuous*. Hence, we distinguish between continuous and discrete location models. Another parameter is the length of the selected subtree. In some of the problems there is a constraint on the total length of the extensive facility, and the objective is then to minimize some monotone function of the distances between the demand points and the facility. In other models, the length of the facility is not a constraint but a decision variable, which enters also into the objective. Following a common classification in logistic models, e.g., [5], we call the former class of problems *tactical* and the latter class *strategic*. Most papers mentioned above deal only with tactical, discrete and continuous location problems. Strategic models on tree networks have been analyzed by [17].

Finally, turning to the objective function, the dependence of the distances of the demand points to the selected facility has commonly been expressed either as the minsum (*median*), or the minmax (*center*) criteria. (An exception is [38], where the authors consider the *centdian* objective which is a convex combination of the median and the center criteria.)

The goal of this paper is to study tactical and strategic location of tree-shaped facilities, using the ordered median objective. This objective function unifies and generalizes the most common criteria mentioned above, e.g., median and center. If there are n demand points, this function is characterized by a sequence of reals, $\Lambda = (\lambda_1, \ldots, \lambda_n)$, satisfying $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$. For a given subtree S, let $X(S) = \{x_1, \ldots, x_n\}$ be the set of weighted distances of the n demand points from S. The value of the ordered median objective at S is obtained as follows: Sort the n elements in $X(S)$ in nonincreasing order, and then compute the scalar product of the sorted list with the sequence Λ. It is easy to see that when $\lambda_i = 1, i = 1, \ldots, n$, we get the median objective and when $\lambda_1 = 1$ and $\lambda_i = 0, i = 2, \ldots, n$, we obtain the center objective. Another important case is the k-centrum objective, characterized by $\lambda_i = 1, i = 1, \ldots, k$, and $\lambda_i = 0$, $i = k + 1, \ldots, n$. The ordered median objective function has been studied recently in the context of locating facilities that can be represented by points in the respective space. (See e.g., [9], [10], [15], [16], [23], [28].)

Our goal is to investigate some algorithmic aspects of locating tree-shaped facilities with the ordered median objective.

It is worth noting the difference between the analysis of problems with the ordered median objective and with other classical functions in location analysis (median, center, centdian, or even centrum). In spite of its inherent similarity the ordered median objective does not possess any clear separability properties. As a result standard DP approaches, commonly used to solve location problems with median, center and even centrum objectives [33], [34], [37], [38], are not applicable here. As a result we need to look for new approaches that rely on alternative tools. We have proven submodularity properties and found linear programming formulations that turn out to be instrumental in solving discrete and continuous versions of the problems, respectively.

Table 1. New results in the paper

Complexity of Tree-shaped facilities [(1)]				
	Tactical		Strategic	
	Discrete	Continuous	Discrete	Continuous
Convex ordered median	NP-hard	$O(n^7 + n^6 I)$	$O(n^6 \log n)$	$O(n^7 + n^6 I)$
k-centrum	NP-hard	$O(n^3 + n^{2.5} I)^{(2)}$	$O(kn^3)$ [(2)]	$O(kn^7)$ [(2)]

[1] I denotes the input size of the problem.
[2] A nestedness property with respect to the point solution holds.

In this paper we consider the strategic and tactical versions of the problem of locating a tree-shaped facility. For the strategic model we prove submodularity of the discrete and continuous models, when the subtree contains a specified node, and the objective is the convex ordered median function. This property leads to strongly polynomial algorithms for the discrete model, based on minimizing a submodular function over a lattice. We could not use this approach to solve the continuous problem since it is not known how to discretize this continuous case. Instead, we develop a compact LP formulation for the continuous version, when the size of the subtree is its length. Specifically, when the tree network has n nodes the LP formulation uses $O(n^2)$ variables and $O(n^2)$ constraints.

For the tactical model the discrete version is NP-hard even for the special case of the median objective. Therefore, we deal only with the continuous version. We formulate the continuous problem, when the subtree contains a specified node, in terms of a compact linear problem with $O(n^2)$ variables and $O(n^2)$ constraints. The validity of this formulation implies that the objective value of the parametric version of the model, where the length of the subtree (facility) is the parameter, is a decreasing piecewise linear convex function of the length of the subtree.

For the particular case of the k-centrum objective we derive stronger results. We present a dynamic programming algorithm that can process both, the discrete and continuous strategic models of the k-centrum version. For this objective we also prove nestedness properties of the subtree solution with respect to the point solution, for the continuous tactical and the continuous and discrete strategic models. The former result enables us to solve the continuous tactical model on the entire tree by solving only one LP problem. In Table 1 we present the complexity of the algorithms developed in the paper.

The paper is organized as follows. In Section 2 we define the models that we deal with, and state the notation used throughout the paper. Section 3 and 4 analyze the tactical and strategic models of the convex ordered median subtree problem, respectively. In Section 5 we restrict ourselves to the special case of the subtree k-centrum problem. There we prove nestedness properties and develop a DP algorithm that solves both the strategic continuous and discrete subtree k-centrum problems. The paper ends with some remarks on extensions to other problems.

2. Notation and model definitions

Let $T = (V, E)$ be an undirected tree network with node set $V = \{v_1, \ldots, v_n\}$ and edge set $E = \{e_2, \ldots, e_n\}$. Each edge e_j, $j = 2, 3, \ldots, n$, has a positive length l_j, and is assumed to be rectifiable. In particular, an edge e_j is identified as an interval of length l_j so that we can refer to its interior points. We assume that T is embedded in the Euclidean plane. Let $A(T)$ denote the continuum set of points on the edges of T. We view $A(T)$ as a connected and closed set which is the union of $n-1$ intervals. Let $P[v_i, v_j]$ denote the unique simple path in $A(T)$ connecting v_i and v_j. Suppose that the tree T is rooted at some distinguished node, say v_1. For each node v_j, $j = 2, 3, \ldots, n$, let $p(v_j)$, the *parent* of v_j, be the node $v \in V$, closest to v_j, $v \neq v_j$ on $P[v_1, v_j]$. v_j is a *child* of $p(v_j)$. e_j is the edge connecting v_j with its parent $p(v_j)$. S_j will denote the set of all children of v_j. A node v_i is a *descendant* of v_j if v_j is on $P[v_i, v_1]$. V_j will denote the set of all descendants of v_j. We refer to interior points on an edge by their distances along the edge from the two nodes of the edge. 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 in $A(T)$ connecting x and y. $A(T)$ is a metric space with respect to the above distance function. The path $P[x, y]$ is also viewed as a collection of edges and at most two subedges (partial edges). $P(x, y)$ will denote the open path obtained from $P[x, y]$, by deleting the points x, y, and $P(x, y]$ will denote the half open path obtained from $P[x, y]$, by deleting the point x. Also, for any pair of subsets $X, Y \subset A(T)$ we define $d(X, Y) = d(Y, X) =$ Infimum $\{d(x, y)|x \in X, y \in Y\}$. If X is a singleton, i.e., $X = \{x\}$ for some $x \in A(T)$, we will use the notation $d(x, Y) = d(Y, x) = d(\{x\}, Y)$. A subset $Y \subset A(T)$ is called a *subtree* if it is closed and connected. Y is also viewed as a finite (connected) collection of partial edges (closed subintervals), such that the intersection of any pair of distinct partial edges is empty or is a point in V. We call a subtree *discrete* if all its (relative) boundary points are nodes of T. For each $i = 1, \ldots, n$, we denote by T_i the subtree induced by V_i. T_i^+ is the subtree induced by $V_i \cup \{p(v_i)\}$.

If Y is a subtree we define the *length* or size of Y, $L(Y)$, to be the sum of the lengths of its partial edges. We also denote by $D(Y)$ the *diameter* of Y. In our location model the nodes of the tree are viewed as demand points (customers), and each node $v_i \in V$ is associated with a nonnegative weight w_i. The set of potential servers consists of subtrees. There is a transportation cost function of the customers (assumed to be at the nodes of the tree) to the serving facility. We use the ordered median objective.

This function is characterized by a sequence of reals, $\Lambda = (\lambda_1, \ldots, \lambda_n)$. For a given subtree S, let $X(S) = \{w_1 d(v_1, S), \ldots, w_n d(v_n, S)\}$ be the set of weighted distances of the n demand points from S. Let $X^-(S)$ be the sequence obtained by sorting the elements in $X(S)$ in nonincreasing order. The value of the ordered median objective at S is then the scalar product of $X^-(S)$ with the sequence Λ. The ordered median objective is called *convex* if $\lambda_1 \geq \lambda_2 \ldots \geq \lambda_n \geq 0$.

Given a real L, the *tactical* subtree problem with an ordered median objective is the problem of finding a subtree (facility) Y of length smaller than or equal to L, minimizing the ordered median objective. When $L = 0$, we will refer to the problem as the *point problem* instead of the subtree problem, and call its solution a *point solution*.

Given a positive real α, the *strategic* subtree model is the problem of finding a subtree (facility) Y minimizing the sum of the ordered median objective and the setup cost of the facility. (The setup cost will be represented by $\alpha L(Y)$.)

We say that a tactical (strategic) model is *discrete* when the endnodes (leaves) of Y must be nodes in V. If the endnodes of Y may be anywhere in $A(T)$ we call the model *continuous*.

3. Tactical subtree with convex ordered median objective

In this section we consider the problem of selecting a subtree of a given length which minimizes the convex ordered median objective. We recall that the discrete model is NP-hard even for the median function, since the Knapsack Problem is a special case. Thus, we consider the continuous version.

First we note that if a continuous subtree does not contain a node it is properly contained in an edge. Hence, to solve the continuous model it is sufficient to solve $O(n)$ continuous subproblems where the subtree is restricted to contain a distinguished node, and in addition, $n-1$ continuous subproblems where the subtree is restricted to be a subedge of a distinguished edge. We show later how to find an optimal subedge of a given edge in $O(n\log^2 n)$ time, by converting the problem into a minimization of a single variable, convex and piecewise linear function. To find an optimal subtree containing a distinguished point (node) we require a different machinery.

3.1. Finding an optimal tactical continuous subtree containing a given node

We first assume that the selected subtree must contain v_1, the root of the tree T. For each edge e_j of the rooted tree, connecting v_j with its parent, $p(v_j)$, assign a variable x_j, $0 \le x_j \le l_j$. The interpretation of x_j is as follows: Suppose that $x_j > 0$, and let $x_j(e_j)$ be the point on edge e_j whose distance from $p(v_j)$, the parent of v_j, is x_j. The only part of e_j, included in the selected subtree rooted at v_1 is the subedge $P[p(v_j), x_j(e_j)]$. In order for the representation of a subtree by the x_j variables to be valid, we also need the following condition to be satisfied:

$$x_j(l_i - x_i) = 0, \text{ if } v_i = p(v_j), v_i \ne v_1, \text{ and } j = 2, \dots, n. \tag{1}$$

Equation (1) ensures the connectivity of the subtree represented by the variables x_j. Unfortunately, these constraints are non-linear. For each node v_t, the weighted distance of v_t from a subtree (represented by the variables x_j), is

$$y_t = w_t \sum_{v_k \in P[v_t, v_1)} (l_k - x_k).$$

We call a set $X = \{x_2, \dots, x_n\}$ *admissible* if $x_2, \dots, x_n$, satisfy $0 \le x_j \le l_j$, $j = 2, \dots, n$, and

$$\sum_{j=2}^{n} x_j \le L. \tag{2}$$

Note that an admissible solution induces a closed subset of $A(T)$ of length less than or equal to L, which contains v_1.

Proposition 3.1. *Let $X = \{x_2, \dots, x_n\}$ be admissible. For each $v_t \in V$, define $y_t = w_t \sum_{v_k \in P[v_t, v_1)} (l_k - x_k)$. Then, there exists an admissible set $X' = \{x'_2, \dots, x'_n\}$, satisfying (1), such that for each $v_t \in V$*

$$y'_t = w_t \sum_{v_k \in P[v_t, v_1)} (l_k - x'_k) \le y_t.$$

Proof. The positive components of X induce a collection of connected components (subtrees) on the rooted tree T. Each such subtree is rooted at a node of T. One of these components contains v_1. (If no positive variable is associated with an edge incident to v_1, then $\{v_1\}$ will be considered as a connected component.) If there is only one component in the collection X itself satisfies (1). Suppose that there are at least two connected components. A connected component T' is called minimal if there is no other component T'', such that the path from T'' to v_1 passes through T'. Let T' be a minimal component. Let v_i be the root of T', and let $x_j, x_j > 0$, induce a leaf edge of T'. In other words, v_i is the closest node to v_1 in T', and there is a node v_j, possibly in T', such that $x_t = 0$ for each descendant v_t of v_j. Let T_1 be the connected component containing v_1. Consider the point of T_1, which is closest to v_i. It is uniquely identified by a node v_k on $P[v_i, v_1]$, and its associated variable x_k. (It is the point on the edge $(v_k, p(v_k))$ whose distance from $p(v_k)$ is exactly x_k.) Let T^*, be the closest component to v_k, amongst all the components which intersect $P[v_i, v_k]$ and do not contain v_1. Let v_q be the root of T^*. Set

$$\epsilon = \min[x_j, d(v_q, v_k) + l_k - x_k].$$

If $\epsilon = d(v_q, v_k) + l_k - x_k$, define an admissible solution $Z = \{z_2, \dots, z_n\}$ by $z_p = l_p$ for each node v_p on $P[v_q, v_k]$, $z_j = x_j - \epsilon$, and $z_s = x_s$ for any other node v_s, $v_s \neq v_j$, v_s not on $P[v_q, v_k]$. It is easy to check that $w_t \sum_{v_k \in P[v_t, v_1)} (l_k - z_k) \leq y_t$, for each $v_t \in V$. Note that the number of connected components induced by Z is smaller than the number of connected components induced by X. We can replace X by Z and proceed. Suppose that $\epsilon = x_j < d(v_q, v_k) + l_k - x_k$. We similarly define an admissible solution $Z = \{z_2, \dots, z_n\}$. Specifically, let v_m be a node on $P[v_i, v_k]$ such that $d(v_m, v_k) + l_k - x_k > x_j \geq d(v_m, v_k) - l_m + l_k - x_k$. Define $z_p = l_p$ for each node v_p, $v_p \neq v_m$, on $P[v_m, v_k]$, $z_m = x_j + x_k + l_m - d(v_m, v_k) - l_k$, $z_j = 0$, and $z_s = x_s$ for any other node v_s, $v_s \neq v_j$, v_s not on $P[v_m, v_k]$. It is easy to check that $w_t \sum_{v_k \in P[v_t, v_1)} (l_k - z_k) \leq y_t$, for each $v_t \in V$. Note that the number of leaf edges of minimal connected components in the solution induced by Z is smaller than the respective number in the solution induced by X. We can replace X by Z and proceed. To conclude in $O(n)$ steps we will identify X' as required. □

The representation of subtrees by the variables x_j is not linear. In spite of that fact, Proposition 3.1 proves that looking for optimal subtrees with respect to an isotone function of the weighted distances, one obtains connectivity without imposing it explicitly. (A function is isotone if it is monotone nondecreasing in each one of its variables.) This argument is formalized in the following corollary.

Corollary 3.1. *Consider the problem of selecting a subtree of total length L, containing v_1, and minimizing an isotone function, $f(y_2, \dots, y_n)$, of the weighted distances $\{y_t\}$, of the nodes $v_t \in V$ from the selected subtree. Then the following is a valid formulation of the problem:*

$$\begin{aligned}
&\min\ f(y_2, \dots, y_n),\\
&s.t.\ \ y_t = w_t \sum_{v_k \in P[v_t, v_1)} (l_k - x_k), \textit{ for each } v_t \in V,\\
&\qquad 0 \leq x_j \leq l_j, \qquad\qquad j = 2, \dots, n,\\
&\qquad \textstyle\sum_{j=2}^{n} x_j \leq L.
\end{aligned}$$

We can now insert the LP formulations from [24] for the k-centrum and the convex ordered median objectives. This will provide a compact LP formulation for finding a continuous subtree rooted at a distinguished point, whose length is at most L, minimizing a convex ordered median objective. For convenience we define $\lambda_{n+1} = 0$.

$$\begin{array}{lll} \min & \sum_{k=1}^{n} (\lambda_k - \lambda_{k+1})(kt_k + \sum_{i=1}^{n} d_{i,k}^{+}) & \\ \text{s.t.} & d_{i,k}^{+} \geq y_i - t_k,\ d_{i,k}^{+} \geq 0, & i = 1, \ldots, n,\ k = 1, \ldots, n, \\ & y_t = w_t \sum_{v_k \in P[v_t, v_1]} (l_k - x_k), & \text{for each } v_t \in V, \\ & 0 \leq x_j \leq l_j, & j = 2, \ldots, n, \\ & \sum_{j=2}^{n} x_j \leq L. & \end{array}$$

(This LP formulation is later used to formulate the continuous strategic model, when the objective is to minimize the sum of the total length of the selected subtree and the ordered median function of the weighted distances to the subtree. Specifically, we add the linear function $\sum_{j=2}^{n} x_j$ to the objective, and remove the length constraint $\sum_{j=2}^{n} x_j \leq L$.) The above formulation uses $p = O(n^2)$ variables and $q = O(n^2)$ constraints. Assuming integer data, let I denote the total number of bits needed to represent the input. Then, by [40], the above LP can be solved by using only $O(n^6 + n^5 I)$ arithmetic operations. We also note that in the unweighted model, where the distances of all demand points are equally weighted, all the entries of the constraint matrix can be assumed to be 0, 1 or -1. Therefore, by [39], the number of arithmetic operations needed to solve the unweighted version is strongly polynomial, i.e., it is bounded by a polynomial in n, and is independent of the input size I.

3.2. Finding an optimal tactical subedge

We show here how to find an optimal subedge for the tactical model. As mentioned above, these are subproblems that we need to consider in the process of finding optimal continuous subtrees. Consider an arbitrary edge of the tree $e_k = (v_s, v_t)$. We show that an optimal subedge of e_k with respect to the tactical model can be found in $O(n \log^2 n)$ time. A point on e_k is identified by its distance, x, from the node v_s. A subedge of length L, incident to x is identified by its two endpoints: x and $x+L$. Let V^s be the set of nodes in the component of T, obtained by removing e_k, which contains v_s. For each node $v_i \in V^s$, its weighted distance from the subedge is given by $h_i(x, L) = w_i(d(v_i, v_s) + x)$. For each node $v_i \in V \setminus V^s$, the respective distance is $h_i(x, L) = w_i(d(v_i, v_t) + d(v_s, v_t) - (x + L))$. In the tactical model L is fixed, and the problem of finding an optimal subedge of e_k of length L reduces to minimizing the single variable, x, ordered median function of a collection of the linear functions $\{h_i(x, L)\}$. (The domain is defined by the constraints $x \geq 0$, and $x + L \leq d(v_s, v_t) = l_k$.) The latter task can be performed in $O(n \log^2 n)$ time, [16].

Remark 3.1. We note in passing that instead of minimizing over each edge separately, we can use the convexity of the objective and save by using global minimization. Globally, the problem reduces to finding a continuous subpath of the tree of length smaller than or equal to L, which minimizes the ordered median objective. (In this tactical continuous subpath model, the objective can be expressed in a way that makes it convex over each path of the tree. Specifically, restricting ourselves to a path of T, and using x as a single variable point along the path, each function $h_i(x, L)$ becomes a single variable, x, piecewise linear convex function with three pieces. The respective slopes are $\{-w_i, 0, w_i\}$.) We will discuss this model in a sequel paper focusing on path-shaped facilities.

To evaluate the overall time complexity of our algorithm for the continuous tactical problem, we note that we solve n linear programs, and find $n-1$ optimal subedges. Therefore, the total complexity is $O(n^7 + n^6 I)$.

4. Strategic subtree with convex ordered median objective

Unlike the tactical model, we will show in this section that the strategic discrete subtree problem with convex ordered median objective is solvable in polynomial time. Specifically, we will formulate this discrete model as a minimization problem of a submodular function over a lattice. But first we consider the continuous version, and show how to solve it, using LP techniques. We assume that the objective function is the sum of the convex ordered median function and the total length of the selected subtree. The argument used above for the tactical continuous subtree problem, proves that it is sufficient to consider only two types of (strategic) subproblems. The first type corresponds to a subproblem where the optimal subtree must contain a distinguished node. In the second type the optimal subtree is a subedge of a given edge.

4.1. Finding an optimal strategic continuous subtree containing a given node

We use the notation in the above section dealing with the tactical continuous subtree problem. In particular, we assume that the optimal subtree must contain v_1, the root of the tree T.

From the discussion above we conclude that this restricted version of the strategic continuous problem can be formulated as the following compact linear program:

$$
\begin{aligned}
\min\ & \sum_{k=1}^{n} (\lambda_k - \lambda_{k+1})(k t_k + \sum_{i=1}^{n} d^+_{i,k}) + \alpha \sum_{j=2}^{n} x_j \\
s.t.\ & d^+_{i,k} \geq y_i - t_k,\ d^+_{i,k} \geq 0, && i = 1, \dots, n,\ k = 1, \dots, n, \\
& y_t = w_t \sum_{v_k \in P[v_t, v_1)} (l_k - x_k), && \text{for each } v_t \in V, \\
& 0 \leq x_j \leq l_j, && j = 2, \dots, n.
\end{aligned}
$$

4.2. Finding an optimal strategic subedge

We use the same notation as in the subsection on finding the tactical subedge. In the case of the strategic model L is variable, and the problem of finding an optimal subedge of

e_k reduces to minimizing the 2-variable, $\{x, L\}$, ordered median function of a collection of the linear functions $\{h_i(x, L)\}$. The feasible set for this problem is defined by the constraints $0 \leq x$, $0 \leq L$, and $x + L \leq d(v_s, v_t) = l_k$. This 2-dimensional minimization can be performed in $O(n \log^4 n)$ time, [16]. By the analysis used above for the tactical model, it is easy to see that the overall time complexity of our algorithm for the continuous strategic problem is also $O(n^7 + n^6 I)$.

4.3. Submodularity of convex ordered median functions

To solve the strategic discrete subtree problem we first prove a submodularity property of the convex ordered median objective.

Let $a = (a_i)$, $b = (b_i)$, be vectors in $\mathbb{R}^n$. Define the *meet* of a, b to be the vector $a \bigwedge b = (\min[a_i, b_i])$, and the *join* of a, b by $a \bigvee b = (\max[a_i, b_i])$. The meet and join operations define a lattice on $\mathbb{R}^n$. Also, for any vector $a \in \mathbb{R}^n$ and $k = 1, \ldots, n$, define $a_{(k)}$ to be the k-th largest component of a, and $S_k(a)$ to be the sum of the k largest components of a, i.e., $S_k(a) = \sum_{t=1}^{k} a_{(t)}$.

Theorem 4.1 (Submodularity Theorem)**.** *Given is a vector* $\Lambda = (\lambda_1, \ldots, \lambda_n)$*, satisfying* $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$. *For any* $a \in \mathbb{R}^n$ *define the function* $f(a) = \sum_{i=1}^{n} \lambda_i a_{(i)}$. *Then,* $f(a)$ *is submodular over the lattice defined by the above meet and join operations, i.e., for any pair of vectors* a, b *in* $\mathbb{R}^n$,

$$f(a \bigvee b) + f(a \bigwedge b) \leq f(a) + f(b).$$

Proof. To prove the above theorem we first observe that

$$f(a) = \sum_{k=1}^{n} (\lambda_k - \lambda_{k+1}) S_k(a).$$

(For convenience we set $\lambda_{n+1} = 0$.) Hence, it is sufficient to prove that for any $k = 1, \ldots, n$,

$$S_k(a \bigvee b) + S_k(a \bigwedge b) \leq S_k(a) + S_k(b). \tag{3}$$

Given a pair of vectors a and b in $\mathbb{R}^n$, let $c = a \bigwedge b$ and $d = a \bigvee b$. To prove the submodularity inequality for $S_k(a)$, it will suffice to prove that for any subset C' of k components of c, and any subset D' of k components of d, there exist a subset A' of k components of a and a subset B' of k components of b, such that the sum of the $2k$ elements in $C' \cup D'$ is smaller than or equal to the sum of the $2k$ elements in $A' \cup B'$. Formally, we prove the following claim:

Claim: Let I and J be two subsets of $\{1, 2, ..., n\}$, with $|I| = |J| = k$. There exist two subsets I' and J' of $\{1, 2, \ldots, n\}$, with $|I'| = |J'| = k$, such that

$$\sum_{i \in I} c_i + \sum_{j \in J} d_j \leq \sum_{s \in I'} a_s + \sum_{t \in J'} b_t.$$

Proof of Claim. Without loss of generality suppose that $a_i \neq b_i$ for all $i = 1, \ldots, n$. Let $I_a = \{i \in I : c_i = a_i\}$, $I_b = \{i \in I : c_i = b_i\}$, $J_a = \{j \in J : d_j = a_j\}$, and $J_b = \{j \in J : d_j = b_j\}$. Since $a_i \neq b_i$ for $i = 1, \ldots, n$, we obtain $|I_a| + |I_b| = |J_a| + |J_b| = k$, I_a and J_a are mutually disjoint, and I_b and J_b are mutually disjoint. Therefore, if $|I_a| + |J_a| = k$, (which in turn implies that $|I_b| + |J_b| = k$), the claim holds with equality for $I' = I_a \cup J_a$, and $J' = I_b \cup J_b$. Hence, suppose without loss of generality that $|I_a| + |J_a| > k$, and $|I_b| + |J_b| < k$. Define $I'' = I_a \cup J_a$, and $J'' = I_b \cup J_b$. Let $K = I'' \cap J''$. $|I''| > k$, and $|J''| < k$. We have

$$\sum_{i \in I} c_i + \sum_{j \in J} d_j = \sum_{i \in K} (a_i + b_i) + \sum_{s \in I'' \setminus K} a_s + \sum_{t \in J'' \setminus K} b_t. \tag{4}$$

On each side of the last equation we sum exactly k components from c, ("minimum elements"), and k components from d, ("maximum elements"). Moreover, the set of components $\{a_s : s \in I'' \setminus K\} \cup \{b_t : t \in J'' \setminus K\}$ contains exactly $k - |K|$ minimum elements and exactly $k - |K|$ maximum elements. In particular, the set $\{a_s : s \in I'' \setminus K\}$ contains at most $k - |K|$ maximum elements. Therefore the set $\{a_s : s \in I'' \setminus K\}$ contains at least $q = |I''| - |K| - (k - |K|) = |I''| - k$ minimum elements. Let $I^* \subset I'' \setminus K$, $|I^*| = q$, denote the index set of such a subset of minimum elements. We therefore have,

$$a_i \leq b_i, i \in I^*. \tag{5}$$

Note that from the construction I^* and J'' are mutually disjoint. Finally define $I' = I'' \setminus I^*$ and $J' = J'' \cup I^*$, and use (4) and (5) to observe that the claim is satisfied for this choice of sets. □

We next use the submodularity property to claim that the strategic discrete subtree ordered median problem is solvable in strongly polynomial time, i.e., the total number of arithmetic operations is polynomial in n and independent of the input size I. First note that the set of all subtrees containing a distinguished point of the tree forms a lattice. For each subtree S of T, consider the vector $a(S) = (a_i(S))$ in $\mathbb{R}^n$, where $a_i(S) = w_i d(v_i, S)$, $i = 1, \ldots, n$. Define

$$f(S) = \sum_{i=1}^{n} \lambda_i a_{(i)}(S).$$

($a_{(i)}(S)$ denotes the i-th largest element in the set $\{a_1(S), \ldots, a_n(S)\}$.) We now demonstrate that $f(S)$ is submodular over the above lattice. Let S_1 and S_2 be a pair of subtrees with nonempty intersection. Then, from [31], we note that

$$w_i d(v_i, S_1 \cup S_2) = w_i \min[d(v_i, S_1), d(v_i, S_2)],$$
$$w_i d(v_i, S_1 \cap S_2) = w_i \max[d(v_i, S_1), d(v_i, S_2)].$$

Then, $\min[a_i(S_1), a_i(S_2)] = w_i d(v_i, S_1 \cup S_2)$ and $\max[a_i(S_1), a_i(S_2)] = w_i d(v_i, S_1 \cap S_2)$. Applying the above Submodularity Theorem we conclude that $f(S)$, the convex ordered median objective, defined on this lattice of subtrees, is submodular, i.e.,

$$f(S_1 \cup S_2) + f(S_1 \cap S_2) \leq f(S_1) + f(S_2).$$

With the above notation the objective function of the strategic problem is $f(S)+\alpha L(S)$. From [31] we note that the length function $L(S)$ is modular and the diameter function $D(S)$ is submodular. Thus, any linear combination with nonnegative coefficients, of the functions $f(S)$, $L(S)$, and $D(S)$ is submodular. Therefore, we can also add the diameter function to the objective and preserve the submodularity. Specifically, we can extend the objective function of the discrete strategic model to

$$f(S) + \alpha L(S) + \beta D(S),$$

where β is a nonnegative real. In addition to the ellipsoidal algorithms, [11], there are now combinatorial strongly polynomial algorithms to minimize a submodular function over a lattice, [18]. The best complexity for a strongly polynomial algorithm, according to [18], is $O(n^4 OE)$ in [11], (quoted from [27]), where OE is the time to compute the submodular function.

It is easy to check that for the strategic discrete subtree problem with a convex ordered median function $OE = O(n \log n)$. Therefore, minimizing over each lattice family (subtrees containing a distinguished node) takes $O(n^5 \log n)$. Solving the discrete strategic subtree ordered median problem will then take $O(n^6 \log n)$!!! (For comparison purposes, the special case of the discrete k-centrum objective is solved in the next section in $O(kn^3) = O(n^4)$ time.)

To solve the continuous strategic model by the above approach where we use the submodularity, we would need to discretize the problem. (Otherwise, even if we assume rational data, the cardinality of the set of subtrees will be exponential in the input size, and not just in the number of nodes.) We do not know of any polynomial discretization for the general convex ordered median objective. As proved in Section 5.4, for the case of the k-centrum objective, when the size of the subtree is its length, we can discretize the problem by augmenting $O(n^3)$ points to the node set.

4.4. Corollaries and related extensions

The problem above can be generalized by replacing the sum of weighted distances by the sum of transportation cost functions of the nodes (demand points) to the subtree (facility). Indeed, let f_i, $i = 1, \dots, n$, be real isotone functions. We consider S_1, S_2 two subtrees of the given lattice. (Subtrees containing a distinguished node.) Then, it is clear that

$$\begin{aligned} g_i(S_1 \cup S_2) &:= f_i(d(v_i, S_1 \cup S_2)) = f_i(d(v_i, S_1) \wedge d(v_i, S_2)) \\ &= \min[f_i(d(v_i, S_1)), f_i(d(v_i, S_2))], \\ g_i(S_1 \cap S_2) &:= f_i(d(v_i, S_1 \cap S_2)) = f_i(d(v_i, S_1) \vee d(v_i, S_2)) \\ &= \max[f_i(d(v_i, S_1)), f_i(d(v_i, S_2))]. \end{aligned}$$

These inequalities allow us to apply Theorem 4.1 to the function:

$$g(S) = \sum_{i=1}^{n} \lambda_i g_{(i)}(S), \quad \text{where } g_i(S) = f_i(d(v_i, S));\ i = 1, \dots, n.$$

($g_{(i)}(S)$ is the i-th largest element in the set $\{g_1(S), \ldots, g_n(S)\}$.) Thus, we conclude that $g(S)$ is submodular on the considered lattice of subtrees.

A special case of the above extension is the *conditional* version of our model, defined as follows. It is assumed that there are already facilities established at some closed subset Y' of $A(T)$. If we select a subtree S as the new facility to be setup, then each demand point v_i will be served by the closest of S and Y'. Hence, the respective transportation cost function is given by $g_i(S) = w_i \min[d(v_i, S), d(v_i, Y')]$.

We can extend further the result on submodularity to the case where customers are represented by subtrees instead of nodes. Let us assume that we are given a family of subtrees $T^i = (V^i, E^i),\ i = 1, \ldots, t$, such that $\bigcup_{i=1}^{t} V^i = V$. For each subtree S of T consider the vector $b(S) = (b_i(S)) \in \mathbb{R}^t$, where $b_i(S) = w_i d(T^i, S)$, $i = 1, \ldots, t$. (In particular, when each $T^i = (\{v_i\}, \emptyset)$, $i = 1, \ldots, n$, we get the customer-node model.) Define

$$g(S) = \sum_{i=1}^{t} \lambda_i b_{(i)}(S).$$

($b_{(i)}(S)$ is the i-th largest element in the set $\{b_1(S), \ldots, b_t(S)\}$.) If S_1, S_2 is a pair of subtrees with non-empty intersection, then we note that

$$w_i d(T^i, S_1 \cup S_2) = w_i \min[d(T^i, S_1), d(T^i, S_2)] = \min[b_i(S_1), b_i(S_2)],$$
$$w_i d(T^i, S_1 \cap S_2) = w_i \max[d(T^i, S_1), d(T^i, S_2)] = \max[b_i(S_1), b_i(S_2)].$$

Once more, we can apply our submodularity Theorem 4.1 to conclude that $g(S)$ is submodular over the considered lattice of subtrees.

5. The special case of the subtree k-centrum problem

In this section we focus on the special case of the k-centrum objective. Recall that the k-centrum objective value for a subtree S is given by the sum of the k largest elements in $X(S) = \{w_1 d(v_1, S), \ldots, w_n d(v_n, S)\}$, the set of weighted distances from S. We consider the strategic discrete and continuous subtree k-centrum problems, and show how to solve them polynomially using dynamic programming techniques. We prove that the discrete version is solvable in cubic time for any fixed k, and we also extend the same recursive approach to the continuous version of the problem. As noted above the tactical discrete model is NP-hard. The tactical continuous model is polynomially solvable, as a special case of the convex ordered median problem discussed above. However, we can do better for the k-centrum objective case. We will show in the next subsections that it is sufficient to solve only one subproblem where the selected subtree must contain a point k-centrum of the tree. (Recall that a point k-centrum is a solution to the tactical model when the length of the subtree is zero.)

5.1. *Nestedness property for the strategic discrete k-centrum problem*

We are interested in studying whether there exists a distinguished point, e.g., an optimal solution to the point k-centrum problem, that must be included in some optimal solution to the subtree k-centrum problem. We call this property *nestedness*. Before we study the nestedness property for the strategic discrete model we first observe that the discrete tactical k-centrum problem does not have the nestedness property with respect to the point solution. Indeed, it can be seen that even for the regular median objective, n-centrum, the property fails to be true, as shown by the following example on the line: $v_1 = 0$, $v_2 = 2$, $v_3 = v_2 + 1/4$, and $v_4 = v_3 + 1$. $w_1 = 2$ and $w_i = 1$ for $i = 2, 3, 4$.

The unique solution for the tactical discrete problem with $L = 0$ is v_2, and the unique solution for the tactical discrete problem with $L = 1$ is the edge (v_3, v_4).

This negative result leads us to investigate the strategic discrete k-centrum problem. We will show that a nestedness property holds for this model.

Theorem 5.1. *Let v' be an optimal solution for the continuous point k-centrum problem. If v' is a node, there is an optimal solution to the strategic discrete subtree k-centrum problem which contains v'. If v' is not a node there is an optimal solution which contains one of the two nodes of the edge containing v'.*

Proof. Let $v^1, \dots, v^t$ be the set of nodes which are neighbors of v'. (If v' is not a node then it has only two neighbors, say v^1, v^2.) Suppose the tree is rooted at v'. Let $T^1, \dots, T^t$ be the components rooted respectively at $v^1, \dots, v^t$. Let $d(v')$ be the k-th weighted largest distance from v'.

Let T' be an optimal discrete subtree which does not satisfy the property stated in the theorem. Without loss of generality, suppose that T' is in T^1, and let x be the node in T' closest to v'. (x is the root of T'.)

For each subtree S, let $F(S)$ denote the objective value of the strategic k-centrum problem at S, and let $U(S)$ denote a set of k nodes corresponding to the k largest weighted distances of nodes from S. (α will denote the coefficient of the length of the subtree in the objective.) Let $z \neq v'$ be a point of the tree, which is either on the path $P[v', v^1]$ or $z \in T^1$. Let $T(z)$ denote the subtree rooted at z, and induced by all the descendants of z. We also define $U^1_z(S) = U(S) \cap T(z)$. (To simplify the notation, if a subtree S consists of a single point, say $S = \{y\}$, we define $U(y) = U(\{y\})$ and $U^1_z(y) = U^1_z(\{y\})$.)

From the optimality of v' we conclude that the derivative of the k-centrum objective is nonnegative in the direction of v^1, the root of T^1. To evaluate this derivative, we assume without loss of generality, that if v'' is a point on the path (subedge) $P[v', v^1]$, sufficiently close to v', the ordering of the k largest weighted distances from any point z on $P[v', v'']$ is fixed and independent of the point. In particular, the k-centrum objective is linear on $P[v', v'']$. We can therefore assume that $U(z)$ is identical for all z in $P[v', v'']$, i.e., $U(z) = U(v')$.

Suppose that k_1 of the nodes in $U(v')$ are in T^1, and k_2 are outside T^1 with $k_1 + k_2 = k$. Let W_1 denote the total weight of the k_1 nodes in T^1, and let W_2 denote the total weight of the other k_2 nodes.

From optimality we have

$$W_2 \geq W_1.$$

Case I. $\alpha \leq W_2$.
Consider an arbitrary point $y \neq v'$ on $P[v', x]$. Let $T'(y)$ be the subtree obtained from T' by augmenting $P[y, x]$, and let $f(y) = F(T'(y))$ denote the objective value at $T'(y)$. Also, let $f'(y)$ be the derivative of $f(y)$ in the direction of v'. We will show

that $f'(y) \leq \alpha - W_2$. (Since $\alpha - W_2 \leq 0$, this will prove that $f(y)$, the objective value at $T'(y)$, is a monotone function of y, along $P[v', x]$, attaining its minimum at v'. Therefore, the statement of the theorem will hold in this case.)

Consider $U(v')$, the set of nodes corresponding to the k largest weighted distances of nodes from v'. Decompose $U(v')$ into three subsets:

$U_y^1(v')$, the subset of $U(v')$ corresponding to nodes in $T(y)$;
$U^2(v')$, the subset of $U(v')$ corresponding to nodes in $T \setminus T^1$; and
$U^3(v') = U(v') \setminus U_y^1(v') \setminus U^2(v')$.

We have $|U^2(v')| = k_2$.

Next consider $U(T'(y))$, a set of k nodes corresponding to the k largest weighted distances from the subtree $T'(y)$. (If $U(T'(y))$ is not uniquely defined we select a solution where $|U(T'(y)) \cap U^2(v')|$ is maximized.) Without loss of generality we assume that y is an interior point. We also assume that for all points u which are outside $T(y)$, and sufficiently close to y, the ordering of the weighted distances of nodes outside $T(y)$ from u, is fixed and independent of u. If $U^2(v') \subset U(T'(y))$, then we clearly have

$$f'(y) \leq \alpha - W_2.$$

Hence, suppose that $U_-^2(y)$, defined by $U_-^2(y) = U^2(v') \setminus U(T'(y))$, is nonempty, and let $q = |U_-^2(y)|$. Consider a node $v_i \in U_-^2(y)$. It satisfies $w_i d(v_i, T') \geq d(v')$, which in turn implies, in view of the maximality property of $|U(T'(y)) \cap U^2(v')|$, that $U(T'(y))$ contains at most $|U_y^1(v')|$ nodes from $T(y)$. (For each node $v_j \in T(y) \setminus U_y^1(v')$, we have $w_j d(v_j, T') \leq w_j d(v_j, v') \leq d(v')$.) Hence, $U(T'(y))$ should include at least $|U^2(v')| + |U^3(v')|$ nodes from $T \setminus T(y)$. Therefore, $U(T'(y))$ contains at least $q = |U_-^2(y)|$ nodes in $T \setminus T(y)$, which are not in $U^2(v') \cup U^3(v')$. Let $U_+^2(y)$ denote such a set of q nodes.

To complete the proof of Case I, it is sufficient to show that for each node $v_j \in U_+^2(y)$, we have

$$w_j \geq w' = \max_{v_i \in U_-^2(y)} w_i,$$

since the latter would imply

$$\begin{aligned} f'(y) &\leq \alpha - \sum_{v_i \in U(T'(y)) \cap U^2(v')} w_i - \sum_{v_j \in U_+^2(y)} w_j \\ &\leq \alpha - \sum_{v_i \in U(T'(y)) \cap U^2(v')} w_i - \sum_{v_i \in U_-^2(y)} w_i \\ &\leq \alpha - \sum_{v_i \in U^2(v')} w_i = \alpha - W_2. \end{aligned}$$

Indeed, let $v_j \in U_+^2(y)$, and $v_i \in U_-^2(y)$. From the definition $v_i \in U^2(v') \setminus U(T'(y))$ and $v_j \in U(T'(y)) \setminus U(v')$. We have

$$w_j d(v_j, v') \leq d(v') \leq w_i d(v_i, v'),$$

and

$$w_j d(v_j, y) \geq w_i d(v_i, y).$$

We note that $v_i \in T \setminus T^1$. Also, $v_j \in T \setminus T^1$ or $v_j \in T^1 \setminus T(y)$. Therefore, the above inequalities imply

$$w_j \geq w_i.$$

This completes the proof of Case I.

Case II. $\alpha > W_2$. We will show that in this case, the objective value does not increase if we shrink T' to its root x.

Assume that the length of T' is L, i.e., $L(T') = L$.

We introduce the following notation. Denote by $D^j(T')$, (respectively $D^j(x)$), $j = 1, \ldots, t$, the sum of the weighted distances to T', (respectively x), from all the nodes in T^j that contribute to the k-centrum objective with respect to T', (respectively x). We decompose $D^1(T')$, (respectively $D^1(x)$), into two terms: $OD^1(T')$, (respectively $OD^1(x)$), the sum of weighted distances that correspond to nodes in $T^1 \setminus T(x)$, and $ID^1(T')$, (respectively $ID^1(x)$), the sum of weighted distances that correspond to nodes in $T(x)$.

Using the above notation, $U_x^1(T')$, (respectively $U_x^1(x)$), is the set of the nodes associated with the weighted distances in $ID^1(T')$ (respectively in $ID^1(x)$). (In case of ties, if the sets $U_x^1(T')$ and $U_x^1(x)$ are not uniquely defined, we select them arbitrarily.) We have,

$$F(T') - F(\{x\}) = OD^1(T') + ID^1(T') + \sum_{j=2}^{t} D^j(T') + \alpha L$$
$$-(OD^1(x) + ID^1(x) + \sum_{j=2}^{t} D^j(x)).$$

To complete the proof of Case II, we will prove that $F(T') - F(\{x\}) \geq 0$. We first make several useful observations. (Note that F1-F4 follow directly from the definitions.)

FACTS:

F1: $D^j(T') \geq D^j(x)$, $j = 2, \ldots, t$.
F2: $OD^1(T') \geq OD^1(x)$.
F3: $p = |U_x^1(T')| \leq |U_x^1(x)| = q' \leq k_1$.
F4: For any subset $A \subset U_x^1(x)$, $|A| = p$, $\sum_{v_j \in A} w_j d(v_j, T') \leq ID^1(T')$.
F5: The sum of the weights of all the nodes that contribute to $ID^1(x)$ is smaller than or equal to W_1, i.e., $\sum_{v_j \in U_x^1(x)} w_j \leq W_1$.

To prove F5 consider $F(\{v'\})$, the k-centrum objective at the (degenerate) subtree consisting only of the point v'. $U_{v^1}^1(v')$ is the set of k_1 nodes in T^1 which contribute to $F(\{v'\})$. In particular, $\sum_{v_j \in U_{v^1}^1(v')} w_j = W_1$. Since $|U_x^1(x)| = q' \leq k_1$, it is sufficient to prove the following claim.

Claim: Suppose that $U^1_{v^1}(v') \setminus U^1_x(x)$ is nonempty and let $v_i \in U^1_{v^1}(v') \setminus U^1_x(x)$ satisfy $w_i = \min_{v_j \in U^1_{v^1}(v') \setminus U^1_x(x)} w_j$. Then $w_j \le w_i$, for each $v_j \in U^1_x(x) \setminus U^1_{v^1}(v')$.

To prove the claim, let u_i be the closest point to v_i on $P[x, v']$. Then,

$$w_j(d(v_j, x) + d(u_i, v')) \le w_j d(v_j, v') \le w_i(d(v_i, u_i) + d(u_i, v')),$$

and

$$w_j d(v_j, x) \ge w_i(d(v_i, u_i) + d(u_i, x)) \ge w_i d(v_i, u_i).$$

The above inequalities imply $w_j d(u_i, v') \le w_i d(u_i, v')$. Since v' is not in T^1, we have $d(u_i, v') > 0$, and therefore $w_j \le w_i$, as claimed.

F6: If $p = q'$, then from F4-F5 we have

$$\begin{aligned} ID^1(x) &= \sum_{v_j \in U^1_x(x)} w_j d(v_j, x) \le \sum_{v_j \in U^1_x(x)} w_j d(v_j, T') + L \sum_{v_j \in U^1_x(x)} w_j \\ &\le ID^1(T') + W_1 L. \end{aligned}$$

F7: If $p = q'$, then from F1, F2 and F6 we have

$$F(T') - F(\{x\}) \ge ID^1(T') + \alpha L - ID^1(x) \ge (\alpha - W_1)L \ge 0.$$

Suppose that $r = q' - p > 0$. Then $|U^1_x(x)| = |U^1_x(T')| + r$. Consider a set of r distinct nodes, $\{v_{i_1}, \ldots, v_{i_r}\}$ in $U^1_x(x) \setminus U^1_x(T')$. Thus, there exist r distinct nodes in $T \setminus T(x)$, $\{v_{j_1}, \ldots, v_{j_r}\}$, such that these nodes contribute to the objective value $F(T')$ but not to $F(\{x\})$. (For $s = 1, \ldots, r$, $w_{j_s} d(v_{j_s}, T') = w_{j_s} d(v_{j_s}, x)$ is one of the k-largest weighted distances from T', but it is not one of the selected k-largest weighted distances from x.) In particular, we have

$$w_{i_m} d(v_{i_m}, T') \le \min_{s=1,\ldots,r} w_{j_s} d(v_{j_s}, T'),$$

for each $m = 1, \ldots, r$, and

$$\sum_{j=2}^{t} D^j(T') + OD^1(T') = \sum_{j=2}^{t} D^j(x) + OD^1(x) + \sum_{s=1}^{r} w_{j_s} d(v_{j_s}, T'). \qquad (6)$$

Moreover, using also the fact that $d(v_j, x) \leq d(v_j, T') + L$, for any $v_j \in T(x)$, and F4-F5, we obtain

$$\begin{aligned}
ID^1(x) = &\sum_{v_j \in U_x^1(x) \setminus \{v_{i_1}, \ldots, v_{i_r}\}} w_j d(v_j, x) + \sum_{s=1}^{r} w_{i_s} d(v_{i_s}, x) \\
&\leq L \sum_{v_j \in U_x^1(x)} w_j + \sum_{v_j \in U_x^1(x) \setminus \{v_{i_1}, \ldots, v_{i_r}\}} w_j d(v_j, T') + \sum_{s=1}^{r} w_{i_s} d(v_{i_s}, T') \\
&\leq L \sum_{v_j \in U_x^1(x)} w_j + \sum_{v_j \in U_x^1(x) \setminus \{v_{i_1}, \ldots, v_{i_r}\}} w_j d(v_j, T') + \sum_{s=1}^{r} w_{j_s} d(v_{j_s}, T') \\
&\leq W_1 L + ID^1(T') + \sum_{s=1}^{r} w_{j_s} d(v_{j_s}, T') \\
&\leq \alpha L + ID^1(T') + \sum_{s=1}^{r} w_{j_s} d(v_{j_s}, T').
\end{aligned}$$

Finally, combining the last upper bound on $ID^1(x)$ with equation (6), we obtain

$$\begin{aligned}
F(T') - F(\{x\}) = OD^1(T') + ID^1(T') + \sum_{j=2}^{t} D^j(T') + \alpha L \\
-(OD^1(x) + ID^1(x) + \sum_{j=2}^{t} D^j(x)) \geq 0.
\end{aligned}$$

This completes the proof for Case II. □

It is easy to verify that the above proof also validates the following nestedness property of the continuous model.

Theorem 5.2. *Let v' be an optimal solution for the continuous point k-centrum problem. There is an optimal solution to the strategic continuous subtree k-centrum problem which contains v'.*

Following the above theorems we note that an optimal solution to the continuous point k-centrum problem can be found in $O(n \log n)$ time by the algorithm in [16].

5.2. Nestedness property for the tactical continuous k-centrum problem

We illustrated in the previous subsection that the strategic discrete subtree k-centrum problem possesses nestedness property which does not extend to the tactical version of the model. We will now use the proof of that result to show that the tactical continuous subtree k-centrum problem has this property. We will use the notation of the previous subsection.

Theorem 5.3. *Let v' be an optimal solution to the continuous point k-centrum problem. Then there exists an optimal solution to the tactical continuous subtree k-centrum problem which contains v'.*

Proof. We use the notation from the proof of Theorem 5.1 in the previous subsection. Let T' be an optimal continuous subtree which does not satisfy the property stated in the theorem. Without loss of generality, suppose that T' is in $T^1 \cup P(v', v^1]$, and let x be the closest point to v' in T'. (x is the root of T'.) Also suppose that amongst all subtrees with these qualifications, T' has the root which is closest to v'. Note that $L(T') = L$.

To prove the theorem we will perturb the tree T' at its root x and at one of its leaves x'', chosen appropriately, by a small and positive δ to obtain a perturbed tree T'' of length L. Specifically, T'' is obtained from T' by decreasing the length of the unique subedge of T' which is incident to x'', by δ, and by augmenting to T' the path $P[x, y]$, where y is the point on $P[x, v']$ satisfying $d(x, y) = \delta$. Denoting by $G(T')$ and $G(T'')$ the sums of the k largest weighted distances of nodes from T' and T'', respectively, we will show that $G(T'') - G(T') \leq 0$.

We now show how to choose the appropriate leaf x''. Recall that $U_x^1(v')$ is the set of nodes in $T(x)$ that contribute to the k-centrum objective at v', and $U_x^1(T')$ is the set of nodes in $T(x)$ that contribute to the k-centrum at T'. Therefore,

$$|U_x^1(T')| \leq |U_x^1(v')|.$$

The set of nodes $U_x^1(T')$ may not be uniquely defined. Nevertheless, without loss of generality, we make the following nondegeneracy assumption on $U_x^1(T')$. Let Y denote the set of leaves of T'. For each $y \in Y$ let e_y be the unique subedge of T' incident to y. Suppose that there are $k_y \geq 0$ nodes in $T(y)$ that contribute to the k-centrum objective at T'. If they are not uniquely defined we select them as follows: Due to the fact that all weighted distance functions are piecewise linear, there is a point $y' \neq y$, sufficiently close to y, on e_y, such that no pair of functions in the collection of linear functions $\{h_j(z) = w_j d(v_j, z) : v_j \in T(y)\}$ has an interior intersection point in the path $P[y, y']$. In particular, the ordering of these linear functions is independent of z. We assume that the k_y nodes in $T(y)$ contributing to the k-centrum objective at T', correspond to the k_y largest functions over $P[y, y']$ in the above collection. In addition, suppose that there are $k_x \geq 0$ nodes in $T \setminus T(x)$ that contribute to the k-centrum objective at T'. By the above argument, there is a point $x^* \neq x$, sufficiently close to x on $P[x, v']$, such that no pair of linear functions in the collection $\{g_j(z) = w_j d(v_j, z) : v_j \in T \setminus T(x)\}$ has an interior intersection point in $P[x, x^*]$. We assume that the k_x nodes in $T \setminus T(y)$ contributing to the k-centrum objective at T', correspond to the k_x largest functions over $P[x, x^*]$ in the above collection.

We distinguish two cases.

1. ($\mathbf{U_x^1(T') = U_x^1(v')}$) In this case we choose x'' to be any arbitrary leaf of T'. Let δ be sufficiently small, and consider the change in the weighted k-centrum objective when we perturb T' at both, the root x and the leaf x''. By the above nondegeneracy assumption, this change is linear in δ. Moreover, this change is the sum of the variations at x and x''. From the discussion in Case I of the proof of Theorem 5.1, we conclude that the variation at x in the direction of v' is less than or equal to $-\delta W_2$.

Using again the nondegeneracy assumption and the fact that $U_x^1(T') = U_x^1(v')$, the variation at x'' in the direction of v' is less than or equal to $\delta \sum_{v_j \in U_x^1(v')} w_j \leq \delta W_1$. Hence, $G(T'') - G(T') \leq \delta(W_1 - W_2) \leq 0$, contradicting the property that T' has the closest root to v' amongst all optimal subtrees of length L.

2. ($\mathbf{U_x^1(T') \neq U_x^1(v')}$) In this case we arbitrarily select a node $v_i \in U_x^1(v') \setminus U_x^1(T')$, satisfying

$$w_i = \min\{w_j : v_j \in U_x^1(v') \setminus U_x^1(T')\}.$$

Let x' be the closest point to v_i in T'. x' is on $P[v_i, v']$, the unique path connecting v_i and v'. (Note that x' is not necessarily a leaf of T', even when $x' \neq v_i$.) Define

$$V(T(x')) = \{v_j \in T(x') : v_j \in U_x^1(T') \setminus U_x^1(v')\}.$$

We make the following claim.

Claim: For each $v_j \in V(T(x'))$, $w_j \leq w_i$.
To prove the claim note that for $v_j \in V(T(x'))$, we have

$$w_i(d(v_i, x') + d(x', v')) \geq w_j(d(v_j, x') + d(x', v')),$$

and

$$w_i d(v_i, x') \leq w_j d(v_j, T') \leq w_j d(v_j, x').$$

The above inequalities clearly imply $w_i d(x', v') \geq w_j d(x', v')$, which proves the claim.
Suppose first that $V(T(x'))$ is empty. In this case all the nodes in $T(x')$ that contribute to $F(T')$ are in $U_x^1(v')$. Select an arbitrary leaf x'' of T' in $T(x')$. Decrease the length of the subedge incident to x'' by a small δ, and perturb T' at x by δ. From the above arguments, the net effect on the objective value is bounded above by $-\delta W_2 + \delta \sum_{v_j \in U_x^1(v')} w_j \leq \delta(W_1 - W_2)$, and therefore

$$G(T'') - G(T') \leq \delta(W_1 - W_2) \leq 0.$$

Next suppose that $V(T(x'))$ is nonempty. Select an arbitrary leaf x'' of T' in $T(x')$. Again, decrease the length of the subedge incident to x'' by a small δ, and perturb T' at x by δ. The net effect on the objective value is bounded above by $\delta(B - W_2)$, where

$$B = \sum_{v_j \in U_x^1(v') \cap U_x^1(T'), v_j \in T(x')} w_j + \sum_{v_j \in U_x^1(T') \setminus U_x^1(v'), v_j \in T(x')} w_j. \tag{7}$$

B_1, the first term in expression (7) is bounded above by

$$\sum_{v_j \in U_x^1(v') \cap U_x^1(T')} w_j.$$

Using the above claim, and the fact that $|U_x^1(T')| \leq |U_x^1(v')|$, we note that B_2, the second term in expression (7), is bounded above by

$$|U_x^1(T') \setminus U_x^1(v')|w_i \leq |U_x^1(v') \setminus U_x^1(T')|w_i \leq \sum_{v_j \in U_x^1(v') \setminus U_x^1(T')} w_j.$$

Thus, $B \leq \sum_{v_j \in U_x^1(v')} w_j \leq W_1$. Therefore, we conclude that

$$G(T'') - G(T') \leq \delta(W_1 - W_2) \leq 0.$$

The proof is now complete.

□

Remark 5.1. A nestedness property for the continuous tactical model implies the result for the continuous case of the strategic version. Thus, the last theorem provides an alternative proof of Theorem 5.2.

From the above nestedness result we can assume that an optimal subtree to the tactical continuous k-centrum problem is rooted at some known point of the tree. Without loss of generality suppose that this point is the node v_1. Therefore, using the formulation in Section 3.1, the problem reduces to a single linear program. Moreover, since $\lambda_i = 1$ for $i = 1, ...k$, and $\lambda_i = 0$ for $i = k+1, \ldots, n$, the formulation has only $O(n)$ variables and $O(n)$ constraints:

$$\begin{aligned}
\min\ & kt_k + \sum_{i=1}^{n} d_{i,k}^{+} \\
\text{s.t. } & d_{i,k}^{+} \geq y_i - t_k,\ d_{i,k}^{+} \geq 0, && i = 1, \ldots, n, \\
& y_t = w_t \sum_{v_q \in P[v_t, v_1]} (l_q - x_q), && \text{for each } v_t \in V, \\
& 0 \leq x_j \leq l_j, && j = 2, \ldots, n, \\
& \textstyle\sum_{j=2}^{n} x_j \leq L.
\end{aligned}$$

We conclude that the overall time complexity to solve the continuous tactical k-centrum problem is $O(n^3 + n^{2.5} I)$.

5.3. *A dynamic programming algorithm for the strategic discrete subtree k-centrum problem*

In this section we present a dynamic programming bottom-up algorithm to solve the strategic discrete subtree k-centrum problem when the selected subtree is restricted to contain a distinguished node, say v_1. From the nestedness results proved above, we know that solving at most two such restricted subproblems will suffice for the solution of the unrestricted problem.

We follow the approach in [32], and assume without loss of generality that the tree T is binary and rooted at v_1. (If a node v_i is not a leaf its two children are denoted by $v_{i(1)}$ and $v_{i(2)}$). Also, following the arguments in [34], we assume without loss of

generality that all weighted distances between pairs of nodes are distinct. For the sake of readability, we recall that V_i is the set of descendants of v_i, T_i is the subtree induced by V_i, and T_i^+ is the subtree induced by $V_i \cup \{p(v_i)\}$. (See Section 2.)
Define $G_i(q,r)$ to be the optimal value of the objective of the subproblem with a subtree rooted at v_i restricted to T_i, (we take the sum of the q largest weighted distances), when the q-th largest weighted distance is exactly r. (If there is no feasible solution set $G_i(q,r) = +\infty$.)
Define $G_i^+(q,r)$ to be the optimal value of the objective of the subproblem with a subtree rooted at v_i restricted to T_i, when the q-th largest weighted distance is greater than r, and the $(q+1)$-st largest weighted distance is smaller than r.(If there is no feasible solution set $G_i^+(q,r) = +\infty$.)
Define $A_i(q,r)$ to be the sum of the q largest weighted distances of nodes in $T \setminus T_i$ to v_i, when the q-th largest is exactly r. (In case there is no such set, let $A_i(q,r) = +\infty$.)
Define $A_i^+(q,r)$ as above, with the condition that the q-th largest is greater than r, and the $(q+1)$-st is smaller than r. (If there is no feasible solution set $A_i^+(q,r) = +\infty$.)
Define $B_i(1,q,r)$ to be the sum of the q largest weighted distances of nodes in $V_{i(1)} \cup \{v_i\}$ to v_i, when the q-th largest is exactly r. (In case there is no such set, let $B_i(1,q,r) = +\infty$.)
Define $B_i(2,q,r)$ to be the sum of the q largest weighted distances of nodes in $V_{i(2)} \cup \{v_i\}$ to v_i, when the q-th largest is exactly r. (In case there is no such set, let $B_i(2,q,r) = +\infty$.)
Define $B_i^+(1,q,r)$ to be the sum of the q largest weighted distances of nodes in $V_{i(1)} \cup \{v_i\}$ to v_i, with the condition that the q-th largest is greater than r, and the $(q+1)$-st is smaller than r. (In case there is no such set, let $B_i^+(1,q,r) = +\infty$.)
Define $B_i^+(2,q,r)$ to be the sum of the q largest weighted distances of nodes in $V_{i(2)} \cup \{v_i\}$ to v_i, with the condition that the q-th largest is greater than r, and the $(q+1)$-st is smaller than r. (In case there is no such set, let $B_i^+(2,q,r) = +\infty$.)

Remark. For convenience the 0-th largest weighted distance is $+\infty$, and for each i, in the definition of G_i, G_i^+, B_i and B_i^+, for any $q > |V_i|$, the $(q+1)$-th largest distance is $-\infty$. A similar convention is used for A_i and A_i^+.

In the above definitions the parameter r is restricted to the set $R^* = \{w_i d(v_i, v_j)\}$, $v_i, v_j \in V$. (Note that $r = 0$ is an element of R^*.) With the above notation, the optimal objective value for the strategic discrete subtree problem, when the subtree is rooted at v_1, is

$$OV_1 = \min_{r \in R^*} G_1(k,r).$$

It is convenient to consider the case $r = 0$ separately. $G_1(k,0)$ is the optimal value when the k-th largest weighted distance is equal to 0. Hence, for any $p \geq k$, the p-th largest weighted distance is equal to 0. Therefore, the problem reduces to finding a subtree, rooted at v_1, which contains at least $n+1-k$ nodes, and minimizes the sum of weighted distances of all nodes to the subtree and the length of the subtree. This problem can be phrased as a special case of the model considered in [14]. (See also [6–8].) In particular, using the algorithm in [14], $G_1(k,0)$ can be computed in $O(kn)$ time.

Recursive Equations for $G_i(q,r)$ and $G_i^+(q,r)$ when $r > 0$

$G_i(q,r)$: Without loss of generality suppose that $r > 0$ is the weighted distance of some node $v_j \in V_{i(1)}$ from the selected subtree.

If the subtree does not include the edges $(v_i, v_{i(1)})$ and $(v_i, v_{i(2)})$, then the best value is

$$C_i = \min_{1 \le q_1 \le q} \{B_i(1, q_1, r) + B_i^+(2, q - q_1, r)\}.$$

If the subtree includes the edge $(v_i, v_{i(1)})$ but not $(v_i, v_{i(2)})$, then the best value is

$$D_i = \alpha d(v_i, v_{i(1)}) + \min_{1 \le q_1 \le q} \{G_{i(1)}(q_1, r) + B_i^+(2, q - q_1, r)\}.$$

If the subtree includes the edge $(v_i, v_{i(2)})$ but not $(v_i, v_{i(1)})$, then the best value is

$$E_i = \alpha d(v_i, v_{i(2)}) + \min_{1 \le q_1 \le q} \{B_i(1, q_1, r) + G_{i(2)}^+(q - q_1, r)\}.$$

If the subtree includes the edges $(v_i, v_{i(1)})$ and $(v_i, v_{i(2)})$, then the best value is

$$F_i = \alpha(d(v_i, v_{i(1)}) + d(v_i, v_{i(2)})) + \min_{1 \le q_1 \le q} \{G_{i(1)}(q_1, r) + G_{i(2)}^+(q - q_1, r)\}.$$

We then have,

$$G_i(q,r) = \min\{C_i, D_i, E_i, F_i\}.$$

$G_i^+(q,r)$: First of all, we define $G_i^+(0,r)$ to be the length of the minimum subtree, T_i', rooted at v_i, ensuring that the weighted distance to each node in V_i is smaller than r. (For each node v_j in V_i, define $v_{k(j)}$ to be the node on $P[v_i, v_j]$, closest to v_i, satisfying $w_j d(v_j, v_{k(j)}) < r$. T_i' is the subtree induced by v_i, and $v_{k(j)}, v_j \in V_i$.)

If the subtree does not include the edges $(v_i, v_{i(1)})$ and $(v_i, v_{i(2)})$, then the best value is

$$C_i^+ = \min_{0 \le q_1 \le q} \{B_i^+(1, q_1, r) + B_i^+(2, q - q_1, r)\}.$$

If the subtree includes the edge $(v_i, v_{i(1)})$ but not $(v_i, v_{i(2)})$, then the best value is

$$D_i^+ = \alpha d(v_i, v_{i(1)}) + \min_{0 \le q_1 \le q} \{G_{i(1)}^+(q_1, r) + B_i^+(2, q - q_1, r)\}.$$

If the subtree includes the edge $(v_i, v_{i(2)})$ but not $(v_i, v_{i(1)})$, then the best value is

$$E_i^+ = \alpha d(v_i, v_{i(2)}) + \min_{0 \le q_1 \le q} \{B_i^+(1, q_1, r) + G_{i(2)}^+(q - q_1, r)\}.$$

If the subtree includes the edges $(v_i, v_{i(1)})$ and $(v_i, v_{i(2)})$, then the best value is

$$F_i^+ = \alpha(d(v_i, v_{i(1)}) + d(v_i, v_{i(2)})) + \min_{0 \le q_1 \le q} \{G_{i(1)}^+(q_1, r) + G_{i(2)}^+(q - q_1, r)\}.$$

We then have,

$$G_i^+(q,r) = min\{C_i^+, D_i^+, E_i^+, F_i^+\}.$$

Preprocessing: To compute all the functions A_i, A_i^+, B_i and B_i^+, it is actually sufficient to find and sort the largest k weighted distances in V_i and $V \setminus V_i$ for all v_i. The latter task will consume only $O(n^2 + nk \log k)$ time. (In fact this task can be performed in $O(n^2)$ time by the methods presented recently in [35].)

Initialization: If v_i is a leaf, then by the general definitions and conventions above, for $r > 0$, $G_i^+(1, r) = +\infty$ and $G_i^+(0, r) = 0$. As noted above, the best objective value attained by a subtree containing v_1 is

$$OV_1 = \min_{r \in R^*} G_1(k, r).$$

To evaluate the total complexity of the algorithm we note that for each node v_i, the functions $G_i(q, r)$ and $G_i^+(q, r)$ are computed for k values of q and $O(n^2)$ values of r. Hence the total complexity, including the preprocessing, is $O(k^2 n^3)$. (We note in passing that since the parameter q is bounded by $\min\{k, |V_i|\}$, the analysis in [32] is applicable also to the above algorithm. In particular, the actual complexity is only $O(kn^3)$.)

5.4. *Solving the strategic continuous subtree k-centrum problem*

The nestedness result shown above, implies that it is sufficient to solve a restricted model where the continuous selected subtree contains a point k-centrum of the tree. (As noted above the latter can be found in $O(n \log n)$ by the algorithm in [16].) Without loss of generality suppose that the subtree must contain the root v_1. We now discretize the continuous problem as follows. A finite set $X \subset A(T)$ is a Finite Dominating Set (FDS) for the strategic continuous subtree k-centrum problem, if there exists an optimal subtree S^* such that all the leaves of S^* are in X. Let

$$\begin{aligned}\mathcal{R} = \{r \in \mathbb{R}^+ : \exists x \in A(T), v_i, v_j \in V, w_i \neq w_j, w_i d(v_i, x)\\ = w_j d(v_j, x) = r, \text{ or } \exists v_i, v_j \in V, r = w_i d(v_i, v_j)\}.\end{aligned}$$

For each node v_i and $r \in \mathcal{R}$, let $x_i(r)$ be the point on $P[v_i, v_1]$ satisfying $w_i d(v_i, x_i(r)) = r$. (If there is no such point set $x_i(r) = v_i$.) Next define

$$\mathcal{PEQ} = \{x_i(r) \in A(T) : \text{ for any } r \in \mathcal{R} \text{ and } i = 1, 2, \ldots, n\}.$$

Theorem 5.4. *The set $\mathcal{PEQ}$ is a Finite Dominating Set (FDS) for the strategic continuous subtree k-centrum problem.*

Proof. Let Y be a given subtree containing v_1, and suppose that no leaf of Y is a node. (The leaves are interior points of the edges.) The objective function is the sum of the function $\alpha L(Y)$ and the sum of the k-largest weighted distances to Y. Consider the change in the objective, due to small perturbations of the leaves of Y. Since the change in $L(\cdot)$ is linear in these perturbations, an FDS for the continuous k-centrum subtree location problem depends only on the change in the sum of the k-largest weighted distances from Y. It is known that the k-centrum objective is reduced to the convex ordered median problem with only two different λ values (namely 0, 1). Moreover, if Y has p leaves then the subtree problem can now be reduced to a p-facility problem. Indeed, since the tree can be considered rooted at v_1, the distance of each node v_j from Y is the distance between v_j and the closest point to v_j in $Y \cap P[v_1, v_j]$. (If v_j is in Y, then $d(v_j, Y) = 0$.) Thus, the p leaves of Y can be identified with the p facilities (points) to be located in the p-facility problem. Therefore, an FDS for the p-facility convex ordered median problem with only two different λ values is also an FDS for the strategic

k-centrum subtree problem. Finally, in [15] it is proved that the set $\mathcal{PEQ}$ is an FDS for this special p-facility location problem. Hence, it is also an FDS for our strategic continuous k-centrum subtree problem. □

To summarize, we note that the strategic continuous k-centrum subtree problem can be discretized by introducing $O(n^3)$ additional nodes, with zero weight, and solving the respective discrete model as above. The $O(n^3)$ augmented nodes do not contribute to the k-centrum objective, and each one of them has only one child of the rooted augmented tree. Therefore, it is easy to check that the implementation of the above algorithm to the problem defined on the augmented tree consumes only $O(kn^7)$ and not $O(kn^9)$ time.

Remark 5.2. Notice that the same FDS is also valid for the modified continuous strategic subtree problem in which the sum of the k-largest weighted distances from a subtree T' is replaced by:

$$a \sum_{l=1}^{k} d_{(l)}(T') + b \sum_{l=k+1}^{n} d_{(l)}(T'); \quad a > b > 0, \tag{8}$$

($d_{(l)}(T')$ is the l-th largest element in the set $\{w_1 d(v_1, T'), \ldots, w_n d(v_n, T')\}$.) We note in passing that the DP algorithm of Section 5.3 can easily be adapted to deal with the more general objective function described in (8). Thus, it also solves the corresponding strategic discrete and continuous subtree problems.

Remark 5.3. It is also worth noting that the set $\mathcal{R}$ considered in the definition of the FDS, $\mathcal{PEQ}$, coincides with the set $\mathcal{R}$ defined in Theorem 1 in [38]. The radii r such that there exist x, v_i, v_j, satisfying $r = w_i d(v_i, x) = w_j d(v_j, x)$ correspond to elements of R_2 and those for which $r = w_i d(v_i, v_j)$ for some $v_i \neq v_j$, correspond to elements in R_1. Unfortunately, it is not clear at this point what kind of FDS is valid for the general version of the p-facility ordered median problem. Thus, the same question applies to the strategic version of our convex ordered subtree problem.

6. Concluding remarks

We have presented above a variety of polynomial algorithms to solve problems of locating tree-shaped facilities using the convex ordered median objective. For the special case of the k-centrum objective we proved nestedness results, which led to improved algorithms. At the moment we do not know whether these nestedness results extend to the ordered median objective. In a sequel paper we plan to investigate the case of path-shaped facilities, using the convex ordered median objective. The goal is to extend the recent subquadratic algorithms for locating path-shaped facilities with the median and center objectives reported in [1, 37, 41, 42]. Note that unlike the case of tree-shaped facilities, there are only $O(n^2)$ topologically different path-shaped facilities on a tree. Therefore, the models with path-shaped facilities are clearly polynomial. The goal is to determine whether low order algorithms exist also for the ordered median objective.

Acknowledgements. We would like to thank Noga Alon for the current version of the proof of Theorem 4.1, which simplifies our original proof significantly. J. Puerto also thanks the Spanish *Ministerio de Ciencia y Tecnología* through grant number BFM2001-2378 for partially supporting his research.

References

1. Alstrup, S., Lauridsen, P.W., Sommerlund, P., Thorup, M.: Finding cores of limited length. In: Algorithms and Data Structures, Lecture Notes in Computer Science n. 1271, Dehne, F., Rau-Chaplin, A. Sack, J-R., Tamassia R. (eds.), Springer, 1997, pp. 45–54
2. Becker, R.I., Perl, Y.: Finding the two-core of a tree. Discrete Appl. Math. **11**, 103–113 (1985)
3. Becker, R.I., Lari, I., Scozzari, A.: Efficient algorithms for finding the (k, l)-core of tree networks. Networks **40**, 208–215 (2003)
4. Boffey, B., Mesa, J.A.: A review of extensive facility location in networks. European J. Operational Res. **95**, 592–600 (1996)
5. Bramel, J., Simchi-Levi, D.: The Logic of Logistics: Theory, Algorithms and Applications for Logistics Management. Springer 1997, Berlin
6. Cho, G., Shaw, D.X.: A depth-first dynamic programming algorithm for the tree knapsack problem. INFORMS J. Comput. **9**, 431–438 (1997)
7. Faigle, U., Kern, W.: Computational complexity of some maximum average weight problems with precedence constraints. Operations Res. **42**, 688–693 (1994)
8. Fischetti, M., Hamacher, H.W., Jornsten, K., Maffioli, F.: Weighted k-cardinality trees: complexity and polyhedral structure. Networks **24**, 11–21 (1994)
9. Francis, R.L., Lowe, T.J., Tamir, A.: Aggregation error bounds for a class of location models. Operations Res. **48**, 294–307 (2000)
10. Francis, R.L., Lowe, T.J., Tamir, A.: Worst-case incremental analysis for a class of p-facility location problems. Networks **39**, 139–143 (2002)
11. Groetschel, M., Lovasz, L., Schrijver, A.: Geometric Algorithms and Combinatorial Optimization. Springer-Verlag 1993, Berlin
12. Hakimi, S.L., Schmeichel, E.F., Labbe, M.: On locating path-or tree shaped facilities on networks. Networks **23**, 543–555 (1993)
13. Hedetniemi, S.M., Cockaine, E.J., Hedetniemi, S.T.: Linear algorithms for finding the Jordan center and path center of a tree. Transportation Sci. **15**, 98–114 (1981)
14. Johnson, D.S., Niemi, K.A.: On knapsack, partitions and a new dynamic technique for trees. Math. Operations Res. **8**, 1–14 (1983)
15. Kalcsics, J., Nickel, S., Puerto, J.: Multi-facility ordered median problems: A further analysis. Networks **41**, 1–12 (2003)
16. Kalcsics, J., Nickel, S., Puerto, J., Tamir, A.: Algorithmic results for ordered median problems defined on networks and the plane. Operations Res. Lett. **30**, 149–158 (2002)
17. Kim, T.U., Lowe, T.J., Tamir, A., Ward, J.E.: On the location of a tree-shaped facility. Networks **28**, 167–175 (1996)
18. McCormick, S.T.: Submodular function minimization. To appear as a chapter in Handbook on Discrete Optimization, edited by K. Aardal, G. Nemhauser and R. Weismantel
19. Minieka, E.: Conditional centers and medians on a graph. Networks **10**, 265–272 (1980)
20. Minieka, E.: The optimal location of a path or tree in a tree network. Networks **15**, 309–321 (1985)
21. Minieka, E., Patel, N.H.: On finding the core of a tree with a specified length. J. Algorithms **4**, 345–352 (1983)
22. Morgan, C.A., Slater, J.P.: A linear algorithm for a core of a tree. J. Algorithms **1**, 247–258 (1980)
23. Nickel, S., Puerto, J.: A unified approach to network location problems. Networks **34**, 283–290 (1999)
24. Ogryczak, W., Tamir, A.: Minimizing the sum of the k largest functions in linear time. Inf. Processing Lett. **85**, 117–122 (2003)
25. Peng, S., Lo, W.: Efficient algorithms for finding a core of a tree with specified length. J. Algorithms **20**, 445–458 (1996)
26. Peng, S., Stephens, A.B., Yesha, Y.: Algorithms for a core and k-tree core of a tree. J. Algorithms **15**, 143–159 (1993)
27. Queyranne, N.M.: Minimizing symmetric submodular functions. Math. Programming **82**, 3–12 (1998)
28. Rodríguez-Chía, A.M., Nickel, S., Puerto, J., Fernández, F.R.: A flexible approach to location problems. Math. Meth. Oper. Res. **51**, 69–89 (2000)
29. Shioura, A., Shigeno, M.: The tree center problems and the relationship with the bottleneck knapsack problems. Networks **29**, 107–110 (1997)
30. Slater, P.J.: Locating central paths in a graph. Transportation Sci **16**, 1–18 (1982)
31. Tamir, A.: A unifying location model on tree graphs based on submodularity properties. Discrete Appl. Math. **47**, 275–283 (1993)
32. Tamir, A.: An $O(pn^2)$ algorithm for the p-median and related problems on tree graphs. Oper. Res. Lett. **19**, 59–64 (1996)

33. Tamir, A.: Fully polynomial approximation schemes for locating a tree-shaped facility: a generalization of the knapsack problem. Discrete Appl. Math. **87**, 229–243 (1998)
34. Tamir, A.: The k-centrum multi-facility location problem. Discrete Appl. Math. **109**, 292–307 (2000)
35. Tamir, A.: Sorting weighted distances with applications to objective function evaluations in single facility location problems. Oper. Res. Lett. **32**, 249–257 (2004)
36. Tamir, A., Lowe, T.J.: The generalized p-forest problem on a tree network. Networks **22**, 217–230 (1992)
37. Tamir, A., Puerto, J., Mesa, J.A., Rodriguez-Chia, A.M.: Conditional location of path and tree shaped facilities on trees. Technical Report, School of Mathematical Sciences, Tel-Aviv University, August 2001
38. Tamir, A., Puerto, J., Perez-Brito, D.: The centdian subtree on tree networks. Discrete Appl. Math. **118**, 263–278 (2002)
39. Tardos, E.: A strongly polynomial algorithm to solve combinatorial linear programs. Oper. Res. **34**, 250–256 (1986)
40. Vaidya, P.M.: An algorithm for linear programming which requires $O((m+n)n^2 + (m+n)^{1.5}nL)$ arithmetic operations. Math. Programming **47**, 175–201 (1990)
41. Wang, B-F: Efficient parallel algorithms for optimally locating a path and a tree of a specified length in a weighted tree network. J. Algorithms **34**, 90–108 (2000)
42. Wang, B-F: Finding a two-core of a tree in linear time. SIAM J. Discrete Math. **15**, 193–210 (2002)