Available online at www.sciencedirect.com
SCIENCE DIRECT®

Discrete Applied Mathematics 130 (2003) 437–448

DISCRETE APPLIED MATHEMATICS

www.elsevier.com/locate/dam

Improved algorithms for several network location problems with equality measures

Juan A. Mesa[a,1], Justo Puerto[b,2], Arie Tamir[c,*,3]

[a] *Dpto. de Matemática Aplicada II, Universidad de Sevilla, Spain*
[b] *Dpto. de Estadística e Investigación Operativa, Universidad de Sevilla, Spain*
[c] *School of Mathematical Sciences, Tel Aviv University, Tel-Aviv, Israel*

Received 11 March 2002; accepted 29 July 2002

Abstract

We consider single facility location problems with equity measures, defined on networks. The models discussed are, the variance, the sum of weighted absolute deviations, the maximum weighted absolute deviation, the sum of absolute weighted differences, the range, and the Lorenz measure. We review the known algorithmic results and present improved algorithms for some of these models.

Keywords: Network location; Equality measures; Algorithms

1. Introduction

During the last two decades there has been a major effort to develop location models capturing more features of real problems. In particular, in the public sector the issue of equity becomes relevant when locating facilities. However, while for efficiency and effectiveness there is almost a consensus that median and center, respectively, are the most representative objective functions, for purposes of equity there does not seem to be an agreement on the proper criteria. One can find in the literature a plethora of

* Corresponding author.
E-mail addresses: jmesa@us.es (J.A. Mesa), puerto@us.es (J. Puerto), atamir@post.tau.ac.il (A. Tamir).
[1] Partially supported by project BFM2000-1052-C02-01 of the Spanish Ministry of Science and Technology.
[2] Acknowledges the Spanish Ministry of Science and Technology through grant number BFM2001-2378.
[3] This paper started during a visit of the author to the Universidad de Sevilla supported by "Ayuda a la Investigación of the Junta de Andalucía".

doi:10.1016/S0166-218X(02)00599-1

functions measuring the inequality of the distribution of distances from demand points to the facility. Two main different lines of research can be observed regarding location models focusing on equity issues.

The first one deals with general aspects such as: how to measure equality, how to define equality measures, what properties equality functions have and what they should have, what the relative positions of the solutions provided by the corresponding optimization problems are and the comparisons among the standardized functions. These topics have been studied in several papers [27,9,26,7]. A review of the existing literature on equity measurement in Location Theory is given by Marsh and Schilling [22]. An interesting discussion on how to select an appropriate equality measure is contained in the paper by Eiselt and Laporte [6].

The second line of research is oriented towards obtaining efficient algorithms for solving location problems involving equality measures. The utility of this research lies not only in supplying good algorithms but it will also allow one to design more general computational experiments for understanding the behavior of and the relationships among the optima of the different associated problems.

We list below some of the most frequently considered equity location models. The most popular models are those in which the variance measure is applied. Maimon [20] obtained a linear time algorithm for the variance location problem on tree networks. Kincaid and Maimon [13] extended this algorithm to the class of 3-cactus graphs satisfying the triangular inequality. The same procedure of reducing 3-blocks to subtrees is applied in [14], for the discrete case. Hansen and Zheng [10] have proposed an $O(mn \log n)$ time algorithm for finding the minimum variance point in a network with n nodes and m edges. Berman [2] has combined efficiency and equality measures into three models: minimizing the variance subject to an upper bound on the average distance, minimizing the average subject to an upper bound on the variance, and minimizing a linear utility function of the average and the variance. Other recent approaches to the variance measure can be found in [15–17].

A second measure is the mean or the sum of weighted absolute deviations from the average distance. This objective was used by Berman and Kaplan [3], where they obtain an $O(mn^2)$ time algorithm to find an optimal solution on a general network. Tamir [28] presented a modified algorithm of $O(mn \log n)$ complexity.

The maximum weighted absolute deviation is considered by López-de-los-Mozos and Mesa in [18]. Based on Hershberger's algorithm for constructing the upper envelope of n segments, an $O(mn^2 \log n)$ time algorithm for the corresponding problem is obtained in [18]. In the above three models: variance, sum of absolute deviations and the maximum deviation, the objective is a monotone function of the deviations from the average distance. We now list and review results for equality measures that do not explicitly depend on these deviations. Minimizing the range, which is a measure conceptually related to the maximum absolute deviation, amounts to minimizing the difference between the maximum and the minimum weighted distances. An $O(mn \log n)$ algorithm for solving this model on a general network can be obtained by applying the algorithm in [4]. Another equality measure, the sum of weighted absolute differences between all pairs of customers, has also been studied by López-de-los-Mozos and Mesa [19]. For general networks they present an $O(mn^2 \log n)$ time algorithm. Finally we note

Table 1
Bold letters indicate new results in the paper

Equality measures in networks				
	General networks		Trees	
	Complexity	Ref.	Complexity	Ref.
Variance	$O(mn \log n)$	[10]	$O(n)$	[20]
SAD[a]	$O(mn \log n)$	[28]	$\mathbf{O(n^2)}$	
MAD[b]	$\mathbf{O(mn^2)}$		$\mathbf{O(n^2)}$	
SAWD[c]	$O(mn^2 \log n)$	[19]	$\mathbf{O(n^2 \log^2 n)}$	
Range	$O(mn \log n)$	[4]	$O(nk \log^2 n)$[d]	[4]
Lorenz	$\mathbf{O(mn^2 \log n)}$		$\mathbf{O(n^2 \log^2 n)}$	

[a]Mean (sum) absolute deviation with respect to the average.
[b]Maximum absolute deviation with respect to the average.
[c]Sum of absolute weighted differences.
[d]k depends on the structure of the tree. For paths $k = O(1)$, but there are trees with $k = \Theta(n)$.

the use of the intricate Lorenz measure for the purpose of equalizing location on tree networks, by Maimon [21]. He derives an $O(n^3 \log n)$ time algorithm for this model. In this paper we develop modified algorithms for some of the models listed above. We consider the case of a general network, as well as tree networks. Our final results are summarized in Table 1.

2. Notation and definitions

Let $G = (V, E)$ be an undirected connected graph with node set $V = \{v_1, \ldots, v_n\}$ and edge set E. Suppose that $|E| = m$. Each edge $e \in E$, has a positive length l_e, and is assumed to be rectifiable. In particular, an edge $e = (v_r, v_s)$ is identified with an interval of length l_e so that we can refer to its interior points. An interior point is identified by its distance along the edge from the two nodes v_r and v_s. Let $A(G)$ denote the continuum set of points on the edges of G. We also view $A(G)$ as a connected set which is the union of m intervals. The edge lengths induce a distance function on $A(G)$. For any pair of points $x, y \in A(G)$, we let $d(x, y)$ denote the length of a shortest path $P(x, y)$, connecting x and y. $A(G)$ is a metric space with respect to the above distance function. We refer to $A(G)$ as the network induced by G and the edge lengths $\{l_e\}$.

Each node v_i, $i = 1, \ldots, n$, is also viewed as a location site of a demand point or a customer. v_i is associated with a pair of nonnegative weights, s_i and w_i. s_i can be interpreted as the inverse of the (constant) speed of the customer situated at v_i. Thus, if there is a server located at some point $x \in A(G)$ the travel time of the customer to the server is $s_i d(v_i, x)$. w_i may represent the number of times the customer will travel to the server. Alternatively, w_i can be viewed as the number of customers, all having speed s_i, located at v_i.

We now define the equality measures which are the subject of this paper. For each $x \in A(G)$, let $d_i(x) = d(v_i, x)$, $i = 1, \ldots, n$. Let $W = \sum_{i=1}^{n} w_i$. We define the average travel time by

$$z_m(x) = \sum_{i=1}^{n} w_i s_i d_i(x)/W.$$

Recall that a point $x \in A(G)$, minimizing $z_m(x)$ is called a weighted median of the network. It is well known that there is always at least one node which is also a weighted median.

Next we consider several equality measures defined by the following objective functions:

$$f_0(x) = \sum_{i=1}^{n} w_i(s_i d_i(x) - z_m(x))^2/W,$$

$$f_1(x) = \sum_{i=1}^{n} w_i|s_i d_i(x) - z_m(x)|/W,$$

$$f_2(x) = \max_{i=1,\ldots,n} w_i|s_i d_i(x) - z_m(x)|,$$

$$f_3(x) = \sum_{i=1}^{n}\sum_{j=1}^{n} |w_i s_i d_i(x) - w_j s_j d_j(x)|,$$

$$f_4(x) = \max_{i=1,\ldots,n} w_i s_i d_i(x) - \min_{i=1,\ldots,n} w_i s_i d_i(x).$$

Minimizing $f_0(x)$ asks for the location minimizing the variance, and $f_1(x)$ amounts to finding the location minimizing the mean or the sum of absolute deviations in travel times of the W customers. Similarly, with $f_2(x)$ as the equality measure, we wish to minimize the maximum weighted absolute deviation. $f_3(x)$ is the sum of all weighted differences. Finally, $f_4(x)$ is the range of variation of the weighted distances.

In addition to the above measures we will also consider the Lorenz equality measure, $f_5(x)$, which is explicitly defined in Section 7.

Our general solution approach to optimize the equality measures on $A(G)$ is based on decomposing the problem, and solving (independently) a restricted problem on each edge. Thus, the properties of the function $z_m(x)$ and the functions $\{|s_i d_i(x) - z_m(x)|\}$ on an edge are relevant.

Consider an edge $e = (u, v) \in E$. For each vertex $v_i \in V$, the function $b_i(x) = s_i d_i(x)$, restricted to this edge, is concave, piecewise linear with at most two segments with slopes s_i and $-s_i$, respectively. (Its breakpoint is called a bottleneck point.) Therefore, the function $z_m(x)$ is concave and piecewise linear on e, and all its breakpoints are bottleneck points. Let us denote by B_e the set of bottleneck points of the edge $e \in E$. Since there is at most one bottleneck point for each vertex on each edge, $|B_e| \leqslant n$.

The difference $s_i d_i(x) - z_m(x)$ changes sign at most four times on each edge. Let I_e be the set of (at most) $4n$ intersection points of the functions $s_i d_i(x)$, $i = 1, 2 \ldots, n$, with $z_m(x)$ on the edge $e = (u, v)$. The sorting of the $O(n)$ points in $B_e \cup I_e$ determines $O(n)$ intervals or *secondary* regions on the edge, each of them limited by two consecutive points of $B_e \cup I_e \cup \{u, v\}$. Let $[x_j(e), x_{j+1}(e)]$ denote such a secondary interval. Then all functions $s_i d_i(x)$, $i=1,\dots,n$, as well as $z_m(x)$, are linear over this interval. Moreover, each function $s_i d_i(x)$ is either above (or coincides at some points) or below the function $z_m(x)$. Denote by

$$N_j^+(e) = \{i \in \{1,\dots,n\} | s_i d_i(x) \geqslant z_m(x),\ \forall x \in (x_j(e), x_{j+1}(e))\},$$

$$N_j^-(e) = \{i \in \{1,\dots,n\} | s_i d_i(x) < z_m(x),\ \forall x \in (x_j(e), x_{j+1}(e))\}.$$

For each pair of nodes v_i, v_j, the functions $b_i(x)=s_i d_i(x)$ and $b_j(x)=s_j d_j(x)$ intersect at most twice on the edge e. Let F_e denote the set of $O(n^2)$ intersection points of all pairs of functions on e, and let $H_e = F_e \cup B_e \cup \{u,v\}$. Additionally, the functions $c_i(x) = w_i s_i d_i(x)$ and $c_j(x) = w_j s_j d_j(x)$ intersect at most twice on the edge e. Let C_e denote the set of $O(n^2)$ intersection points of all pairs of $c_i(x)$, $c_j(x)$ functions on e, and let $D_e = C_e \cup B_e \cup \{u,v\}$. (For convenience we view the sets D_e and H_e as *multi-sets,* i.e., if several pairs of functions intersect at the same point, then this point is accordingly multiplied in the corresponding set.)

3. The mean absolute deviation problem

In this section we consider the minimization of the function $f_1(x)$. As mentioned in the introduction the special case where all customers have the same speed, i.e., $s_i = s$, for $i=1,\dots,n$, was solved by Berman and Kaplan [3] in $O(mn^2)$ time. Tamir [28] presented a modified solution approach improving the bound to $O(mn\log n)$. It is easy to see that the algorithm in [28] is directly applicable to the general case of arbitrary $\{s_i\}$ without affecting the $O(mn\log n)$ bound. This bound provides an $O(n^2\log n)$ time algorithm when it is applied to the particular case of tree networks. We will next show how to improve this bound and solve the problem on a tree in $O(n^2)$ time. More specifically, we show how to minimize the objective on each edge of the tree in $O(n)$ time.

In order to determine the local minima on each edge e, the restricted problem can be formulated as a linear program:

$$\min \sum_{i=1}^{n} y_i$$

$$y_i \geqslant w_i(s_i d_i(x) - z_m(x)), \quad i=1,2,\dots,n,$$

$$y_i \geqslant -\, w_i(s_i d_i(x) - z_m(x)), \quad i=1,2,\dots,n,$$

$$0 \leqslant x \leqslant l_e,$$

in which $s_i d_i(x) - z_m(x)$ is linear since both $s_i d_i(x)$ and $z_m(x)$ are linear on each edge.

This formulation can be considered as the dual of a Multiple Choice Knapsack Problem for which Zemel [29] obtained a linear time algorithm based on adaptation of Megiddo's algorithm for linear programming in fixed dimension, [23]. Hence, in $O(n)$ time we identify a minimizer of $f_1(x)$ on each edge e. The total time to solve the problem is therefore $O(n^2)$.

4. The maximum weighted absolute deviation problem

The most common equality measures used in location problems, the variance and the mean absolute deviation, do not sufficiently account for the worst performance of the system. In order to overcome this drawback, the maximum weighted absolute deviation measure, $f_2(x)$, has been proposed.

For the problem of minimizing the maximum weighted absolute deviation an $O(mn^2 \log n)$ time algorithm, based on the determination of the non-dominant intersection points of all pairs of weighted absolute deviation functions and using the upper envelope in each primary region, was proposed in [18]. (Actually, [18] considers only the case $s_i = 1,\ i = 1, \ldots, n$.)

However, we will next show that a different approach based on solving a linear program on each *secondary* region improves the time complexity to $O(mn^2)$.

Consider an edge $e = (u, v) \in E$. Consider the set of points, $B_e \cup I_e \cup \{u, v\}$ on e, defined above. The $O(n)$ points in $B_e \cup I_e$ induce $O(n)$ intervals or *secondary* regions on the edge, each of them limited by two consecutive points of $B_e \cup I_e \cup \{u, v\}$. Let $[x_j(e), x_{j+1}(e)]$ be such a secondary interval. Then the problem

$$\min_{x \in [x_j(e), x_{j+1}(e)]} \max_{i=1,\ldots,n} w_i |s_i d_i(x) - z_m(x)|$$

can be formulated as

$$\begin{aligned}
&\min y \\
&y \geqslant w_i(s_i d_i(x) - z_m(x)), \quad i \in N_j^+(e), \\
&y \geqslant -w_i(s_i d_i(x) - z_m(x)), \quad i \in N_j^-(e), \\
&x_j(e) \leqslant x \leqslant x_{j+1}(e).
\end{aligned}$$

The above linear program can be solved in $O(n)$ time by the algorithm in [23].

In each edge the procedure involves the sorting of $O(n)$ points, which can be done in $O(n \log n)$ time, and solving $O(n)$ linear programs. Thus the resulting time is $O(n^2)$ per edge and $O(mn^2)$ for the whole network.

The above bound reduces to $O(n^3)$ for tree networks. However, using the arguments of the previous section we observe that it takes only $O(n)$ time to find the best solution on each edge e. Specifically, the restricted problem on e can be formulated as the following two variable linear program, and therefore can be solved in $O(n)$ time by the algorithm in Megiddo [23].

$$\begin{aligned}
&\min y \\
&y \geqslant w_i(s_i d_i(x) - z_m(x)), \quad i = 1, 2, \ldots, n, \\
&y \geqslant -w_i(s_i d_i(x) - z_m(x)), \quad i = 1, 2, \ldots, n, \\
&0 \leqslant x \leqslant l_e.
\end{aligned}$$

We conclude that for tree networks $f_2(x)$ can be minimized in $O(n^2)$ time.

5. The maximum absolute deviation problem

In this section we consider the unweighted case where there is a single customer at each node v_i, i.e., $w_i = 1$ for $i = 1, \dots, n$. We will show that the $O(mn^2)$ bound for the general case can be improved to $O(mn \log^2 n)$ when

$$f_2(x) = \max_{i=1,\dots,n} |s_i d_i(x) - z_m(x)|.$$

For this problem we use a different approach to solve the restricted problem on each edge.

Consider an edge $e = (u, v) \in E$. Using the notation in the previous section let

$$x_1(e) < x_2(e) < \cdots < x_t(e), \quad t = O(n),$$

be the sorted sequence of distinct points in $B_e \cup I_e \cup \{u, v\}$. For each interval $[x_j(e), x_{j+1}(e)]$ consider both, the set of indices of the functions $s_i d_i(x)$ above and below $z_m(x)$: $N_j^+(e)$ and $N_j^-(e)$.

Next, define the following piecewise convex functions over the interval $[x_j(e), x_{j+1}(e)]$.

$$g_j(x) = \max_{i \in N_j^+(e)} s_i d_i(x),$$

$$h_j(x) = \max_{i \in N_j^-(e)} \{-s_i d_i(x)\},$$

$$F_j(x) = \max\{g_j(x) - z_m(x), h_j(x) + z_m(x)\}.$$

Then, to find the best point on this interval we need to find a minimizer of $F_j(x)$. Equivalently we will solve the following linear problem dynamically, using the machinery developed in Hershberger and Suri [12].

$$\min y,$$

$$y \geqslant g_j(x) - z_m(x),$$

$$y \geqslant h_j(x) + z_m(x),$$

$$x_j(e) \leqslant x \leqslant x_{j+1}(e).$$

With the above machinery we (separately) maintain dynamically the breakpoints (or the slopes) of the functions $g_j(x)$ and $h_j(x)$. In particular, for each value of x we can compute both functions and their directional derivatives in $O(\log n)$ time. Hence, for each value of x, we can determine in $O(\log n)$ time whether x is to the left or to the right of a minimizer of the function $F_j(x)$. Therefore, by applying a binary search over the breakpoints of $g_j(x)$ and $h_j(x)$ we can find a minimizer of $F_j(x)$ over the given interval in $O(\log^2 n)$ time. (We suspect that the latter bound can be improved to $O(\log n)$ but we do not yet know how to achieve that.)

Next we discuss the total effort needed to maintain the sequence of functions $\{g_j(x)\}$ and $\{h_j(x)\}$. Each of these functions is an upper envelope of a collection of at most n linear functions, which is maintained dynamically by the data structure in Hershberger and Suri [12]. It takes $O(\log n)$ time to delete or insert a linear function, if, as is the case here, the sequence of insertions and deletions is known a priori. To explain and identify this sequence, consider two consecutive intervals, $[x_j(e), x_{j+1}(e)]$ and $[x_{j+1}(e), x_{j+2}(e)]$. Suppose first that $x_{j+1}(e)$ is in B_e. Then it is a maximum point of some function $s_i d_i(x)$. (To simplify, assume that there is only one such index i, and $x_{j+1}(e)$ is not in I_e. Otherwise we perform the following step sequentially for all relevant indices.) We now delete the linear function, which is the increasing part of $s_i d_i(x)$, and replace it in the same collection, by the decreasing part of that function. Respectively, we update the (linear) function $z_m(x)$ by subtracting the increasing part of $s_i d_i(x)$ and adding its decreasing part. Next suppose that $x_{j+1}(e)$ is in I_e. Then there is a function $s_i d_i(x)$ which intersects $z_m(x)$. (Again, to simplify the discussion, suppose that there is only one such index i, and $x_{j+1}(e)$ is not in B_e. Also assume that it is the increasing part of $s_i d_i(x)$ which intersects $z_m(x)$, and $z_m(x_{j+1}(e)-) > s_i d_i(x_{j+1}(e)-)$. In this case the function $s_i d_i(x)$ is in the collection $N_j^-(e)$.) We delete i from $N_j^-(e)$ to obtain $N_{j+1}^-(e)$, and add it to $N_j^+(e)$ to obtain $N_{j+1}^+(e)$. (The other cases are treated similarly.) From the discussion in the previous sections it is clear that the total number of deletions and insertions over the underlying edge e is $O(n)$. Moreover, the sequence of deletions and insertions can be determined a priori during the process of computing and sorting the sequence $x_1(e), x_2(e), \ldots, x_t(e)$. Therefore, after an initial effort of $O(n \log n)$, we can conclude that the total effort to perform all the deletions and insertions is also $O(n \log n)$ according to the algorithm proposed by Hershberger and Suri [12]. As noted above the minimization over each interval $[x_j(e), x_{j+1}(e)]$ takes $O(\log^2 n)$ time. Thus, in $O(n \log^2 n)$ time we find a minimum point on an edge, leading to an $O(mn \log^2 n)$ time algorithm for the entire network.

6. The sum of absolute weighted differences problem

In this section we consider the minimization of the objective

$$f_3(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} |w_i s_i d_i(x) - w_j s_j d_j(x)|.$$

An $O(mn^2 \log n)$ algorithm to minimize this measure over a general network is given in López-de-los-Mozos and Mesa [19]. We briefly describe such an algorithm, and then show an improved algorithm for tree networks.

Consider an edge $e = (u, v) \in E$. We will show how to find a minimizer of $f_3(x)$ on e in $O(n^2 \log n)$ time.

The function $f_3(x)$ is clearly linear over each interval of e connecting two adjacent points of D_e. In particular, there is a point in D_e which minimizes $f_3(x)$ over e. Therefore, to optimize $f_3(x)$ over e it is sufficient to evaluate this objective at all points of D_e. To perform the latter task, we first sort the points in D_e, consuming $O(n^2 \log n)$ time.

To evaluate the objective at a point x, we note that if the ordering of the elements in the multi-set $\{c_i(x)\}$ is known, $f_3(x)$ can be computed in $O(n)$ time. (Recall that $c_i(x) = w_i s_i d_i(x)$ was defined in Section 2.) Specifically, assume without loss of generality that

$c_1(x) \leqslant c_2(x) \leqslant \cdots \leqslant c_n(x)$, $i = 1, \ldots, n$. Also, define $C_k(x) = c_1(x) + \cdots + c_k(x)$, for $k = 1, \ldots, n$. Then,

$$f_3(x) = 2[((n-1)c_n(x) - C_{n-1}(x)) + ((n-2)c_{n-1}(x) - C_{n-2}(x)) + \cdots + (c_2(x) - C_1(x))],$$

or

$$f_3(x) = 2\sum_{i=1}^{n} (2i - n - 1)c_i(x).$$

We start the evaluation by computing $f_3(x)$ at the node u, using $O(n \log n)$ time. We then follow L_e, the (sorted) sequence of points in the multiset D_e. It is easy to see that if x_q and x_{q+1} are consecutive points of L_e, then it takes constant time to compute $f_3(x_{q+1})$ from $f_3(x_q)$. Therefore, the total effort to compute $f_3(x)$ at all points in D_e is dominated by $O(n^2 \log n)$, the effort needed to sort D_e. The total effort to minimize $f_3(x)$ over the entire network is $O(mn^2 \log n)$. The above bound reduces to $O(n^3 \log n)$ for tree networks. However, we will show how to reduce this bound to $O(n^2 \log^2 n)$, by solving the problem on each edge of the tree in $O(n \log^2 n)$ time.

Consider an edge of the tree $e = (u, v)$. Then, for each pair of nodes v_i, v_j, the function $w_i s_i d_i(x) - w_j s_j d_j(x)$ is linear over e. In particular, $f_3(x)$ is convex there. For $i = 1, \ldots, n$, suppose that the linear representation of the function $w_i s_i d_i(x)$ over e is given by $w_i s_i d_i(x) = \alpha_i x + \beta_i$. Then D_e consists of all points of the form $x_{i,j} = (\beta_j - \beta_i)/(\alpha_i - \alpha_j)$, $i, j = 1, \ldots, n$, $i \neq j$.

As mentioned above there is a point in D_e which is a minimizer of $f_3(x)$ over e, and it takes $O(n \log n)$ time to evaluate the function at any point x. Using the convexity of $f_3(x)$, and the special structure of the set D_e we can apply the search procedure in Megiddo and Tamir [24], with the modification in Cole [5], to find the minimum in $O(n \log^2 n)$ time. This will lead to an $O(n^2 \log^2 n)$ algorithm for finding the minimum of $f_3(x)$ on a tree network.

7. Maximizing the Lorenz measure

In this section we consider another equality index, called the Lorenz measure, which is quite common and useful in economics to define and measure equity in the income of a population. Maimon [21] has adapted this measure to location models. He argues that choosing the location of the server according to this measure ensures that the distance to the population is as much as possible homogenously distributed. (The reader is referred to Maimon [21] for additional characteristics of this criterion.) Maimon presents an $O(n^3 \log n)$ algorithm to find the optimal location on a tree with respect to the Lorenz measure. In an unpublished report, written more than ten years ago, Hansen and Zheng [11] gave an $O(n^2 \log n)$ improved algorithm for this model. We consider a weighted version of this model, where we replace the distances to the server by travel times. We focus on the algorithmic aspects of this generalized model. Specifically, we will present an $O(n^2 \log^2 n)$ algorithm to find an optimal solution to the generalized model on a tree. (For the case treated by Maimon [21] the complexity of our algorithm reduces to $O(n^2 \log n)$, which matches the improvement reported by Hansen and Zheng [11].) We also show how to extend the results to general networks.

To facilitate the discussion and introduce the Lorenz measure consider a point $x \in A(G)$, where a service facility is to be located. The travel time of a customer at v_i to x is $s_i d_i(x)$. Assume, without loss of generality, that $b_1(x) \leqslant b_2(x) \leqslant \cdots \leqslant b_n(x)$, where $b_i(x) = s_i d_i(x)$, $i = 1, \ldots, n$. For each $k = 1, \ldots, n$, define $W_k = w_1 + \cdots + w_k$. (Note that the latter definition depends on the ordering of $\{b_i(x)\}$. Also recall that with the notation introduced above $W_n = W$. Since in this section w_k is interpreted as the proportion of the population situated at v_k, we assume that $W_n = W = 1$, and define $W_0 = 0$.)

Following the derivation and expressions in Maimon, [21], we define

$$L(x) = \sum_{k=1}^{n} w_k \left(\sum_{j=1}^{k-1} w_j b_j(x) + w_k b_k(x)/2 \right).$$

Rearranging terms we obtain,

$$L(x) = \sum_{k=1}^{n} w_k (W - W_{k-1} - w_k/2) b_k(x).$$

The Lorenz measure, $f_5(x)$, is then defined by

$$f_5(x) = 2L(x)/z_m(x).$$

(We note that the model discussed by Maimon [21] corresponds to the case where $s_i = 1$ for $i = 1, \ldots, n$.) The objective is to find a point in $A(G)$ maximizing $f_5(x)$.

We first observe some useful properties of the functions involved. Consider an edge $e=(u,v)$ of the network $G=(V,E)$. The function $L(x)$ is clearly linear over each interval of e connecting two adjacent points of H_e. (See Section 2.) From the above discussion we also recall that the average function $z_m(x)$ is piecewise linear and concave on e. B_e is the set of breakpoints of $z_m(x)$. In particular, the Lorenz function, $f_5(x)$ is a ratio of two linear functions over each interval of e connecting two adjacent points of H_e. Therefore, there is a point in H_e which maximizes $f_5(x)$ over e. To find an optimum point it is sufficient to evaluate this objective at all points of H_e. The total effort to perform this task, is $O(n^2 \log n)$. It is very similar to the procedure described in the previous section to evaluate the function $f_3(x)$, and therefore we skip the details. We only note that the effort is dominated by the $O(n^2 \log n)$ time needed to compute and sort the $O(n^2)$ elements of H_e. It is easy to see that with the above expression for $L(x)$, it takes only constant time to compute $f_5(x)$ at each additional point of the sorted list of points obtained from H_e. To summarize, we conclude that in $O(mn^2 \log n)$ time we can locate a point of $A(G)$ maximizing the Lorenz measure $f_5(x)$.

For tree networks, the above bound reduces to $O(n^3 \log n)$, the bound reported by Maimon [21]. However, we will show how to reduce this bound to $O(n^2 \log^2 n)$, by solving the problem on each edge of the tree in $O(n \log^2 n)$ time. (The approach is again very similar to the one described above for optimizing $f_3(x)$. However, some of the ingredients are different.)

Consider an edge of the tree $e=(u,v)$. Then, for each node v_i, the function $b_i(x)=s_id_i(x)$ is linear over e. Moreover, the proof given in Maimon [21] for the case where $s_i=1$, for $i=1,\dots,n$, showing that $L(x)$ is concave and piecewise linear, extends directly to arbitrary nonnegative $\{s_i\}$. Thus, $L(x)$ is a piecewise linear and concave function over e. H_e is its set of breakpoints. Also, $z_m(x)$ is linear over e. Using Proposition 5.20 in Avriel et al. [1], we conclude that $f_5(x)$ is semistrictly quasiconcave over e. In particular, by Theorem 3.37 in [1], every local maximum of $f_5(x)$ on e is a global maximum over e. Thus, to find a maximizer of $f_5(x)$ over an edge, we can apply a binary search over H_e, and locate a maximum point in H_e, by evaluating the objective $f_5(x)$ at $O(\log n)$ points. We perform the search on H_e without explicitly generating this set of $O(n^2)$ cardinality.

For $i=1,\dots,n$, suppose that the linear representation of the function $b_i(x)=s_id_i(x)$ over e is given by $s_id_i(x)=\gamma_i x+\delta_i$. Then H_e consists of all points of the form $x_{i,j}=(\delta_j-\delta_i)/(\gamma_i-\gamma_j),\ i,j=1,\dots,n,\ i\neq j$.

As mentioned above there is a point in H_e which is a maximizer of $f_5(x)$ over e. It takes $O(n\log n)$ time to evaluate the function at any point x. Using the above quasiconcavity of $f_5(x)$, and the special structure of the set H_e we can apply the search procedure in Megiddo and Tamir [24], with the modification in Cole [5], to find the maximum in $O(n\log^2 n)$ time. This will lead to an $O(n^2\log^2 n)$ algorithm for finding the maximum of $f_5(x)$ on a tree network.

An improvement is possible for the model considered in Maimon, [21], i.e., $s_i=1$ for $i=1,\dots,n$. In this case the structure of H_e is simpler. Specifically, for $e=(u,v)$, let T_u (T_v) be the subtree containing the node u (v), obtained by cutting the edge e. Then for each node $v_i\in T_u$, we can assume that $b_i(x)=x+\delta_i$, where $\delta_i=d(v_i,u)$, and for each node $v_j\in T_v$, we can assume that $b_j(x)=-x+\delta_j$, where $\delta_j=d(v_j,u)$. Therefore, H_e consists of all points of the form $x_{i,j}=(\delta_j-\delta_i)/2,\ v_i\in T_u,\ v_j\in T_v$. Moreover, if we sort the coefficients $\delta_k=d(v_k,u),\ k=1,\dots,n$, then for each point x of H_e it takes only $O(n)$ time to sort $\{b_1(x),\dots,b_n(x)\}$. When the sorting is known, it follows from the above expressions that $f_5(x)$ can then be evaluated in $O(n)$ time. With these tools we can now directly apply the search procedure in Megiddo et al. [25], and Frederickson and Johnson [8], to find the maximum in $O(n\log n)$ time. This will lead to an $O(n^2\log n)$ algorithm for finding the maximum of $f_5(x)$ on a tree network in the case where $s_i=1,\ i=1,\dots,n$.

References

[1] M. Avriel, W.E. Diewart, S. Schaible, I. Zang, Generalized Concavity, Plenum, New York, 1988.

[2] O. Berman, Mean-variance location problems, Transp. Sci. 24 (1990) 287–293.

[3] O. Berman, E.H. Kaplan, Equity maximizing facility location schemes, Transp. Sci. 24 (1990) 137–144.

[4] R.E. Burkard, H. Dollani, Center problems with Pos/Neg weights on trees, European J. Oper. Res. 145 (2003) 483–495.

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

[6] H.A. Eiselt, G. Laporte, Objectives in location problems, in: Z. Drezner (Ed.), Facility Location: A Survey of Applications and Methods, Springer, Berlin, 1995, pp. 151–180.

[7] E. Erkut, Inequality measures for location problems, Location Sci. 1 (1993) 199–217.
[8] G.N. Frederickson, D.B. Johnson, The complexity of selection and ranking in $X+Y$ and matrices with sorted columns, J. Comput. System Sci. 24 (1982) 97–208.
[9] J. Halpern, O. Maimon, Accord and conflict among several objectives in locational decisions on tree networks, in: J.F. Thisse, H.G. Zoller (Eds.), Locational Analysis of Public Facilities, North-Holland, Amsterdam, 1983, pp. 301–314.
[10] P. Hansen, M. Zheng, An algorithm for the minimum variance point of a network, RAIRO Rech. Opér./Oper. Res. 25 (1991) 119–126.
[11] P. Hansen, M. Zheng, private communication, January 2002.
[12] J. Hershberger, S. Suri, Off-line maintenance of planar configurations, J. Algorithms 21 (1996) 453–475.
[13] R.K. Kincaid, O. Maimon, Locating a point of minimum variance on triangular graphs, Transp. Sci. 23 (1989) 216–219.
[14] R.K. Kincaid, O. Maimon, A note on locating a central vertex of a 3-cactus graph, Comput. Oper. Res. 17 (1990) 315–320.
[15] M.C. López-de-los-Mozos, J.A. Mesa, The 2-variance location problem in a tree network, Stud. Locational Anal. 11 (1997) 73–87.
[16] M.C. López-de-los-Mozos, J.A. Mesa, Heuristics for multiple facility location in a network using the variance criterion, J. Oper. Res. Soc. 51 (2000) 971–981.
[17] M.C. López-de-los-Mozos, J.A. Mesa, The variance location problem on a network with continuously distributed demand, RAIRO Oper. Res. 34 (2000) 155–181.
[18] M.C. López-de-los-Mozos, J.A. Mesa, The maximum absolute deviation measure in location problems on networks, European J. Oper. Res. 135 (2001) 184–194.
[19] M.C. López-de-los-Mozos, J.A. Mesa, The sum of weighted absolute differences location problem, 2001, submitted for publication.
[20] O. Maimon, The variance equity measure in locational decision theory, Ann. Oper. Res. 6 (1986) 147–160.
[21] O. Maimon, An algorithm for the Lorenz measure in locational decisions on trees, J. Algorithms 9 (1988) 583–596.
[22] M.T. Marsh, D.A. Schilling, Equity measurement in facility location analysis: a review and framework, European J. Oper. Res. 74 (1994) 1–17.
[23] N. Megiddo, Linear-time algorithms for linear programming in R^3 and related problems, SIAM J. Comput. 12 (1983) 759–776.
[24] N. Megiddo, A. Tamir, New results on the complexity of p-center problems, SIAM J. Comput. 12 (1983) 751–758.
[25] N. Megiddo, A. Tamir, E. Zemel, R. Chandrasekaran, An $O(n\log^2 n)$ algorithm for the kth longest path in a tree with applications to location problems, SIAM J. Comput. 10 (1981) 328–337.
[26] G.F. Mulligan, Equality measures and facility location, Papers Regional Sci. 70 (1991) 345–365.
[27] E.S. Savas, On equity in providing public services, Manage. Sci. 24 (1978) 800–808.
[28] A. Tamir, On the complexity of some classes of location problems, Transp. Sci. 26 (1992) 352–354.
[29] E. Zemel, An $O(n)$ algorithm for the linear multiple choice knapsack problem and related problems, Inform. Process. Lett. 18 (1984) 123–128.