Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

State of the Art—Location on Networks: A Survey. Part I: The p-Center and p-Median Problems

Barbaros C. Tansel, Richard L. Francis, Timothy J. Lowe,

To cite this article:
Barbaros C. Tansel, Richard L. Francis, Timothy J. Lowe, (1983) State of the Art—Location on Networks: A Survey. Part I: The p-Center and p-Median Problems. Management Science 29(4):482-497. https://doi.org/10.1287/mnsc.29.4.482

MANAGEMENT SCIENCE
Vol. 29, No. 4, April 1983
Printed in U.S.A.

State of the Art

LOCATION ON NETWORKS: A SURVEY. PART I: THE p-CENTER AND p-MEDIAN PROBLEMS*†

BARBAROS C. TANSEL,‡ RICHARD L. FRANCIS§ AND TIMOTHY J. LOWE**

Network location problems occur when new facilities are to be located on a network. The network of interest may be a road network, an air transport network, a river network, or a network of shipping lanes. For a given network location problem, the new facilities are often idealized as points, and may be located anywhere on the network; constraints may be imposed upon the problem so that new facilities are not too far from existing facilities. Usually some objective function is to be minimized. For single objective function problems, typically the objective is to minimize either a sum of transport costs proportional to network travel distances between existing facilities and closest new facilities, or a maximum of "losses" proportional to such travel distances, or the total number of new facilities to be located. There is also a growing interest in multiobjective network location problems.

Of the approximately 100 references we list, roughly 60 date from 1978 or later; we focus upon work which deals directly with the network of interest, and which exploits the network structure. The principal structure exploited to date is that of a tree, i.e., a connected network without cycles. Tree-like networks may be encountered when having cycles is very expensive, as with portions of interstate highway systems. Further, simple distribution systems with a single distributor at the "hub" can often be modeled as star-like trees. With trees, "reasonable" functions of distance are often convex, whereas for a cyclic network such functions of distance are usually nonconvex. Convexity explains, to some extent, the tractability of tree network location problems.
(FACILITIES/EQUIPMENT PLANNING—LOCATION)

1. Introduction

Network location problems occur when new facilities are to be located on a network. The network of interest may be a road network, an air transport network, a river network, or a network of shipping lanes. For a given network location problem, the new facilities are often idealized as points, and may be located anywhere on the network; constraints may be imposed upon the problem so that new facilities are not too far from existing facilities. Usually some objective function is to be minimized. For single objective function problems, typically the objective is to minimize either a sum of transport costs proportional to network travel distances between existing facilities and closest new facilities, or a maximum of "losses" proportional to such travel distances, or the total number of new facilities to be located. There is also a growing interest in multiobjective network location problems.

*Accepted by Marshall L. Fisher; received July 7, 1980. This paper has been with the authors 4 months for 2 revisions.

†We dedicate this paper to Jonathan Halpern, a pioneer in research on network location problems, whose untimely death was a great loss to the profession.

‡Georgia Institute of Technology.
§University of Florida.
**Purdue University.

0025-1909/83/2904/0482$01.25

TABLE 1

Frequency Distribution of Publication or Issue Date of Network Location References

Year	Number of Publications
1982	10
1981	14
1980	12
1979	11
1978	9
1977	10
1976	6
1975	5
1974	4
1973	2
1972	6
1971	3
1970	4
1969	1
1968	1
1967	2
1966	1
1965	1
1964	1
1963	0
1962	1
⋮	
1869	1

As Table 1 demonstrates, the literature on network location problems has grown rapidly since the appearance of Hakimi's seminal paper [42] on the "absolute center and median" problems in 1964. We shall use Figure 1 as a conceptual framework for our survey of this literature, so that Figure 1 provides an outline of our paper. We place primary emphasis on theoretical results, together with constructive solution approaches which exploit the network structure. We devote each separate section of our survey to the discussion of a particular problem type identified in Figure 1, pointing out relations between various types. We use the subheadings in each section to distinguish among special cases of a given problem, such as the case of a general network, a tree network, the single facility case, and the multifacility case.

For purposes of insight, the types of networks we shall consider can usually be visualized as road networks, with nodes representing intersections, and arcs (often straight line segments) representing portions of roads joining (adjacent) intersections. Other possible networks of interest include river, transport, and wiring networks.

Let N represent an imbedded planar network with a set of vertex locations $V = \{v_i : i \in I = \{1, \ldots, n\}\}$, where I is the set of vertex indices, and edge set E. We assume at most one edge joins any two distinct vertices, and E contains no loops. Also, we assume each edge has positive length, and is rectifiable, in the sense that there is a one-to-one correspondence between each edge and the interval [0, 1]. Hence if x is *any* given point on any edge, say the edge E of length δ_{st} joining v_s and v_t, then there is a unique number between zero and one, say $w_{st}(x)$, such that $w_{st}(x)\delta_{st}$ and $[1 - w_{st}(x)]\delta_{st}$ are the lengths along the edge between v_s and x, and x and v_t, respectively. It is now direct to define the *distance* $d(x, y)$ for any two points $x, y \in N$ to be the length of any shortest path in N joining x and y. This distance is a function $d(\cdot, \cdot)$ which satisfies the following metric properties for any $x, y \in N$: (i) (nonnegativity) $d(x, y) \geqslant 0$, with $d(x, y) = 0$ iff $x = y$; (ii) (symmetry) $d(x, y) = d(y, x)$; (iii) (triangle inequality) $d(x, y) \leqslant d(x, u) + d(u, y)$ for any $u \in N$.

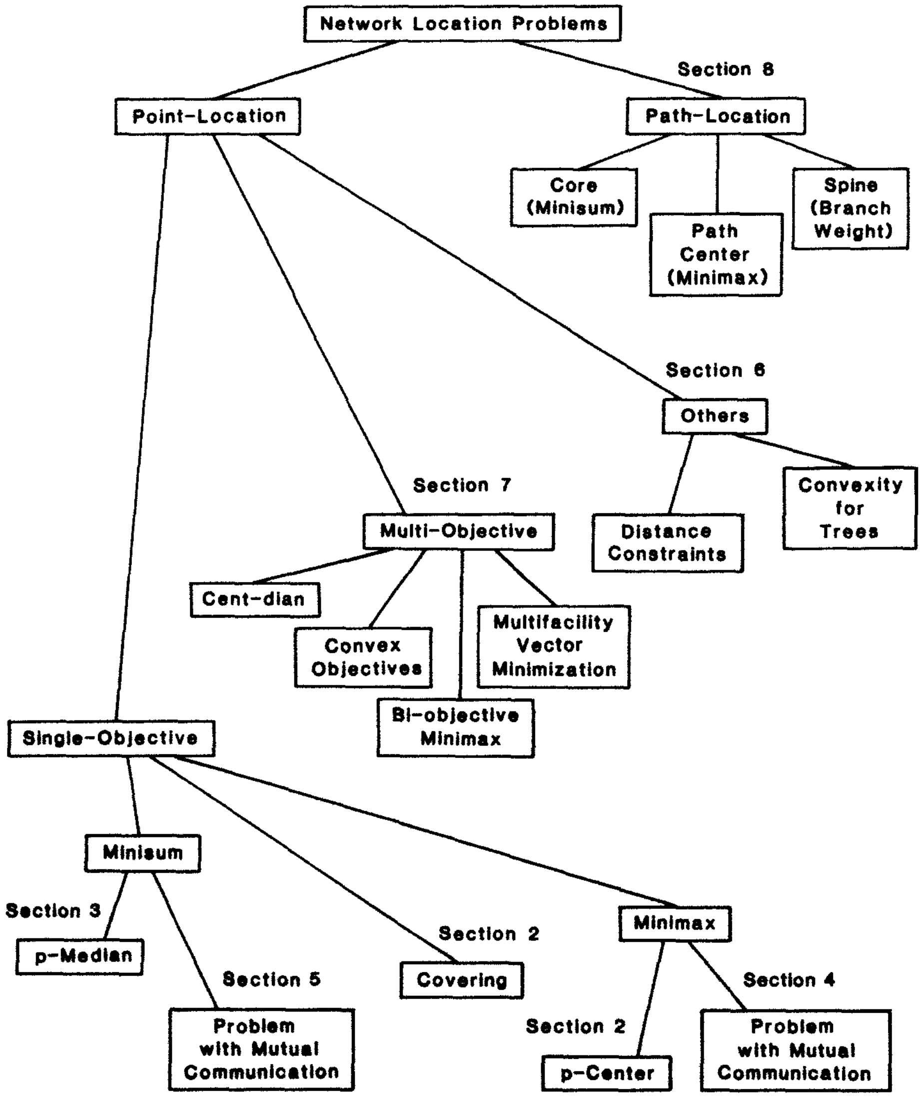

FIGURE 1. Family Tree for Network Location Problems.

We remark in passing that imbedded networks can be more general than planar imbeddings; see, for example, [17].

When N contains a unique shortest path between any x and y in N we call N a *tree*, and write T instead of N. A river network and its tributaries provide a natural example of a tree network. Tree-like networks may also be encountered in sparsely occupied regions or, alternatively, when having cycles is very expensive (e.g., in portions of interstate highway systems). Further, simple distribution systems with a single distributor at the "hub" can often be modeled as star-like trees.

Sometimes it is convenient to consider a location *vector* $X = (x_1, \ldots, x_m)$, with $x_j \in N$, $j = 1, \ldots, m$, in which case we say that X belongs to N^m (or T^m), the m-fold cartesian product of N (or T) with itself. (We note that for $X, Y \in N^m$, if we define the function $d_m(\cdot, \cdot)$ by $d_m(X, Y) = \sum\{d(x_i, y_i) : i = 1, \ldots, m\}$ then it is known that $d_m(\cdot, \cdot)$ is a metric, so that N^m becomes a metric space, with the customary metric space properties [34]: it is not unusual to find authors making the implicit assumption that N^1 is a metric space.)

For remaining notation, for any finite subsets Y and Z of N, we define $D(Y,Z) = \min\{d(y,z): y \in Y,\ z \in Z\}$. Also we denote by $|S|$ the cardinality of any finite set S.

Subsequently we shall consider a number of algorithms for solving location problems. Hence it is convenient to summarize the customary way of specifying the computational order of "effort" of such algorithms. Given a class of problems P_n, where n is some appropriately chosen measure of the "size" of the problems in the class, we say that an algorithm to solve the problem in P_n is of order $f(n)$, written $O(f(n))$, if $cf(n)$ is an *upper bound* on the amount of effort to solve *any* problem in P_n, where $f(n)$ is a real valued function, and c is some constant (usually unspecified). Note that $O(f(n))$ is a *worst-case* measure, as it is an upper bound on the amount of effort needed to solve the "most difficult" problem in P_n. It follows, if an algorithm has a low order of effort, e.g., $O(n \log n)$ or $O(n^2)$, that it is usually quite an efficient algorithm. However, an algorithm with a high order of effort may still be efficient in the sense that its *average* order of effort is not excessive. Unfortunately, the probability density function of the problems in a class is usually unknown, so that the average amount of effort can only be estimated, based on using the algorithm to solve *some* sample of the problems in the class. The statistical properties of such a sample are often unknown. However, in the case where the algorithm designer is the sampler, the reported results may not emphasize a sample of problems on which the algorithm performs badly.

We now give a brief overview of our survey. We devote §§2 through 5 to single objective location problems, while in §6 we discuss the "distance constraints" problem, and convexity. We discuss multiobjective location problems in §7. The results of §4, and the latter part of §7, exploit the results given in §6 for the distance constraints. We point out convexity concepts intrinsic to tree networks [19] throughout wherever relevant, and summarize convexity concepts at the end of §6. We discuss the covering problem in §2 in relation to the p-center problem. In §8 we give a brief discussion of path-location problems recently introduced by Slater [103]. In §9 we give a brief discussion of the state of the art, and of current research trends in the area of network location.

The p-center and p-median problems discussed in §2 and §3 provide a basis for the other network location problems that we survey. Part I of this paper is devoted to these two fundamental problems as well as notation that will be employed in Part II. We remark that virtually all of the work surveyed in Part II exploits network structure. Specifically, it is based on the assumption that the network is a tree.

In the interest of keeping the paper to a manageable length and giving it a well-defined network emphasis, we have omitted much of the literature dealing with integer programming (IP) location problems. We have included recent IP literature where the problems are equivalent to network location problems, and we give brief surveys of recent mathematical programming-based procedures for dealing with the IP problems. We have chosen to emphasize work which deals directly with a network, rather than transforming a network problem to an integer programming problem. Our omission of much of the IP literature is in no way a value judgement; rather, it is a reflection of our interests and an attempt to give the paper a focus. Readers interested in literature dealing with IP location problems are referred to the texts [12], [58], and [83] discussed below, and the review papers of Krarup and Pruzan [70], [71], McGinnis [78], and ReVelle *et al.* [96].

A recent text by Handler and Mirchandani [58] discusses extensively a portion of the network location literature which involves deterministic and probabilistic cases of minimax and minisum problems. Discussed also are the cent-dian and medi-center problems, which involve combinations of these two objectives for the case of a single new facility. The text is highly recommended, but most of the references date from 1977 or earlier, so that much recent work has necessarily been omitted. In addition, the text by Christofides [12] devotes one chapter to the p-center problem, and one to the p-median problem ($p \geqslant 1$). The text by Minieka [83] devotes a chapter to network location problems, with most of its emphasis on the 1-center and 1-median problems.

2. The p-Center Problem

Given functions f_i, $i \in I$, define the function f for $X \subset N$ by $f(X) = \max\{f_i(D(v_i, X)) : i \in I\}$.

The *p-center problem* is to find an *absolute p-center* $X^* = \{x_1^*, \ldots, x_p^*\}$ and the *p-radius* r_p for which

$$r_p \equiv f(X^*) = \min[f(X) : |X| = p, X \subset N].$$

The problem can be considered as one of locating centers either to minimize a maximum loss, or to provide good service. Supposing travel from centers to possible emergency locations specified by the v_i, $f_i(D(X, v_i))$ can be interpreted as the loss incurred during the time required to travel from the closest center to v_i, in which case $f(X)$ is the maximum loss incurred given the centers specified by X. Minimizing f thus minimizes the maximum loss: a conservative approach. Alternatively, $f_i(D(X, v_i))$ can be interpreted as the time to travel from v_i to the nearest "service center," in which case minimizing f minimizes the maximum time to travel from a vertex to a closest service center, thus providing good service.

If X is restricted to subsets of V, the problem will be referred to as the *vertex restricted* case. In the majority of the literature on this problem, the function $f_i(D(v_i, X))$ takes the form $w_i D(v_i, X) + a_i$, where the w_i are positive *weights* and the a_i are *addends*. In this case, when the w_i are all unity, we refer to the problem as the *unweighted* problem; otherwise, we refer to the problem as the *weighted* problem. In the case where each point y in N is a demand point, as opposed only to vertices, the definition of $f(X)$ will be $f(X) = \max[D(y, X) : y \in N]$ and the corresponding problem will be referred to as the (unweighted) *continuous p-center problem.*

In what follows, if we make no mention of addends with respect to a problem, then all addends are zero. If we mention addends with respect to a problem, then at least one a_i is nonzero.

1-*Center Problem on a General Network*

The absolute 1-center weighted problem was defined and solved by Hakimi [42] in 1964. To find the absolute center, Hakimi examines the function f on each edge, finds a best local minimum on that edge, and selects the best among $|E|$ such local minima. This method takes advantage of one important property of f, namely, that it is piecewise linear and continuous on each edge, with at most $n(n-1)/2$ break points. A local minimum always occurs either at a break point of f or at an end point of the edge. Hakimi, Schmeichel, and Pierce [45] showed that Hakimi's method can be implemented in $O(|E|n^2 \log n)$ effort, giving a computational refinement which reduces the effort to $O(|E|n \log n)$ for the unweighted case. Further refinements of the procedure were obtained by Kariv and Hakimi [66], resulting in an $O(|E|n \log n)$ algorithm for the weighted case and $O(|E|n)$ algorithm for the unweighted case. All these refinements focus on finding the break points and local minima of f in the most efficient manner. In [85], Minieka gives an $O(n^3)$ algorithm for solving the unweighted 1-center problem.

Frank [29] and Minieka [82] showed that the continuous 1-center problem can be reduced to a computationally finite one and proposed solution procedures similar to Hakimi's.

Minieka [84] considers "conditional" absolute, vertex-restricted, and continuous 1-center problems. Such problems occur when a center must be located not only with respect to vertices, but with respect to previously located centers as well. He shows that such problems reduce to the standard "unconditional" problems through an appropriate redefinition of distances.

1-*Center Problem on a Tree Network*

Goldman [38] solved the unweighted problem in the presence of addends. Goldman's algorithm is based on the repeated application of a "trichotomy theorem" that either determines the edge on which the absolute center lies, or reduces the search to one of the two subtrees obtained by removing all interior points of that edge. Halfin [46] refined Goldman's algorithm, improving it from $O(n^2)$ to $O(n)$.

For the unweighted case, Handler [52] presented an especially elegant algorithm. Handler's method finds any longest path in the tree and locates the absolute center at the midpoint of the path. To find a longest path, Handler chooses an arbitrary vertex v_i, finds a farthest vertex v_s from v_i, and then finds a farthest vertex v_t from v_s. The path $P(v_s, v_t)$ is a longest path and its midpoint is the unique absolute center of the tree. This procedure requires a computational effort of $O(n)$. Handler's algorithm was extended by Lin [73] to the unweighted case with addends. Lin showed that the absolute center of a general network N with vertex addends can be found by determining the absolute center of an expanded network N' whose vertex addends are all zero. Network N' is obtained from N by adding a new vertex adjacent to each old vertex, with the length of the edge connecting the two equal to the addend associated with the old vertex. For a tree network T, the resulting network is a tree T' and Handler's $O(n)$ algorithm [52] can be applied to T'.

For the weighted case with addends, Dearing and Francis [17] showed (for any network) that r_1 is bounded below by α, where

$$\alpha = \max\{(d(v_i, v_j) + a_i/w_i + a_j/w_j)/((1/w_i) + (1/w_j)) : i, j \in I\}. \tag{2.1}$$

Dearing and Francis proved that the lower bound is always attainable for the case of a tree network. The computation of α requires $O(n^2)$ operations and, once α is computed, two "critical" vertices defining α are identified. The absolute center can be readily located on the path joining the two critical vertices and its location is unique. Hakimi, Schmeichel, and Pierce [45] proved a theorem which reduces the computational effort for computing α. Their theorem states that if for some α_{st} it is true that $\max[\alpha_{si} : 1 \leq i \leq n] = \alpha_{st} = \max[\alpha_{it} : 1 \leq i \leq n]$ then α_{st} is the maximum of all α_{ij}. Kariv and Hakimi [66] gave an $O(n \log n)$ algorithm which confines the search to successively smaller subtrees until an edge is obtained: the absolute center is located at the local center (also the global center for a tree) on this edge using Hakimi's procedure for finding a local minimum.

A nonlinear version of the 1-center problem was considered and solved by Dearing [16], and by Francis [23]. In this version, each weight w_i is replaced by a strictly increasing and continuous function f_i of the distance $d(v_i, x)$. Both authors obtained a lower bound subsuming the one defined by (2.1). The bound is applicable to all networks and is always attainable for tree networks.

A "roundtrip" version of the problem was solved by Chan and Francis [7]. In this problem, each "demand point" is a pair of vertices (v_i, u_i) and $f(x)$ is the maximum of the roundtrip distances defined by $\rho_i(x) \equiv w_i[d(v_i, x) + d(x, u_i) + a_i]$. A lower bound similar to the one defined by (2.1) is obtained. The bound is not applicable to all networks, but is applicable and always attainable for tree networks. We note that the computation of such bounds [16], [23], [52] is a maximization problem (see (2.1)) and so may be viewed as a rudimentary duality. Indeed duality we shall discuss for the p-center problem reduces (when $p = 1$) to a dual problem identical to (2.1).

Vertex Restricted 1-Center Problem

The vertex restricted 1-center problem was considered as early as 1869 by Jordan [65] as a graph theoretic problem. Jordan pointed out that the set of 1-centers consists of either one vertex or two adjacent vertices. This problem can be solved in $O(n^3)$ by computing and then by examining the distance matrix of the network, as demonstrated by Hakimi [42]. Recently Hedetniemi, Cockayne, and Hedetniemi [59] have developed $O(n)$ algorithms for solving the unweighted 1-center problem with, and without, addends. Their algorithms have the lowest computational order to date of any for solving such problems, and are based on "an efficient data structure for representing a tree called a canonical recursive representation." Rosenthal, Hersey, Pino, and Coulter [99] introduced a generalized algorithm that solves a number of "eccentricity" problems on tree networks, one of which is the vertex restricted 1-center problem. In this case, the eccentricity of a vertex is defined to be the distance from that vertex to a farthest vertex. This generalized algorithm determines the eccentricity of each vertex by making only two traversals of the vertices. The vertex center is that vertex with the minimum eccentricity. Rosenthal [98] describes a very general "sensitivity analysis" algorithm which is applicable to several different types of vertex restricted one-facility location problems in trees, including the one-center problem. Slater [105] considered the problem of finding the vertex center of a network with respect to subnetworks. In this version of the problem, each "demand" is a known collection of vertices (or a subnetwork induced by the collection). The distance between a vertex and any such collection is the distance between a nearest element of the collection to that vertex. For a given vertex, the value of the objective function at that vertex is the maximum of the distances between that vertex and any such collection. Slater showed that a matrix D' can be constructed from the distance matrix D of the network, so that each entry of D' is a distance from a vertex to a nearest element of a collection. Slater demonstrated that the vertex center with respect to collections of vertices can be found by examining the matrix D'.

Slater [101], [102] considered the *security center problem* on a network. The concept of a security center is motivated in a competitive environment where it is desired to find a vertex, u, which maximizes the minimum of the differences between the number of vertices strictly closer to u and the number of vertices strictly closer to any vertex v. Slater showed that the search for a security center of a network N with unit edge lengths can be reduced to a search over a single *block* of N, where a block of N is any subnetwork S, of N, which is maximal with respect to the property that the removal of any vertex of S will not disconnect S. Slater showed that the security center of a tree is identical to the vertex restricted 1-center of a tree.

p-Center Problem on a General Network ($p \geq 2$)

The p-center problem was formulated by Hakimi [43]. Subsequently, a number of solution procedures have been suggested. A common characteristic of all these procedures is that they all rely on solving a sequence of set covering problems.

For completeness, we first define a set covering problem and an r-cover problem. At this point we depart from our practice of discussing only research working directly with the networks of interest, because early integer programming set covering research preceded, and appears to be the basis for, research working directly with the networks.

Let A be a matrix of zeros and ones, y a vector of zero-one variables y_i. The problem of minimizing $\sum_i y_i$ so that each row of Ay is greater than or equal to one is called the (minimal) *set covering* problem. Given the function $f(X) = \max\{w_i D(v_i, X) : i \in I\}$, the problem of minimizing $|X|$ so that $f(X) \leqslant r$ for some given value of r is called the *r-cover problem*.

Denoting by $q(r)$ the minimum value of the r-cover problem, it can be readily shown that, if $q(r) = p$ for some r, and $q(r') > p$ for any $r' < r$, then r is the p-radius and any X which solves the r-cover problem is an absolute p-center.

Minieka [81] considered the unweighted case on a general network and showed that the problem can be reduced to a computationally finite one. Minieka identifies a finite point set P' such that there exists an absolute p-center contained in $P = P' \cup V$. A point x on some edge is a member of P' if and only if x is the *unique* point on its edge such that $d(v_i, x) = d(x, v_j)$ for some two *distinct* vertices v_i and v_j. Based on this result, Minieka suggested a rudimentary algorithm that relies on solving a finite sequence of set covering problems. Based on the framework provided by Minieka, an exact algorithm, in which the number of columns may be reduced, was given by Garfinkel, Neebe, and Rao [31]. Using the results in [53] and [81], Handler [55], [57] proposed a relaxation approach, in which both the number of rows and columns may be reduced, which appears to perform well on large scale problems. As observed in [58], the above methods apply, with minor changes, to the weighted case.

For the weighted case, Christofides and Viola [13] gave a solution procedure which relies on solving a sequence of r-cover problems with successively increasing values of r. In the process, one also obtains the solutions for $n-1, n-2, \ldots, p+1$ center problems. The solution of each r-cover problem is obtained in two stages: first, all feasible solutions to the r-cover problem are obtained by finding all regions on the network that can be reached by a vertex within a radius r. Then, among all the feasible solutions, one with minimum cardinality is found by solving a set covering problem.

Kariv and Hakimi [66] showed that the p-center problem on a general network is NP-hard. They also showed that the weighted case can be reduced to a computationally finite one. Based on this finiteness property they gave an algorithm whose complexity is $O[|E|^p(n^{2p-1})(\log n)/(p-1)!]$. Also they gave an algorithm for the unweighted problem in which unity replaces $\log n$ in the foregoing order.

Hsu and Nemhauser [62] showed that finding an approximate solution to the vertex restricted or absolute p-center problem whose value is within either 100% or 50%, respectively, of the optimal value is NP-hard.

Minieka [82] considered a continuous p-center problem on a general network, assuming all points on each edge must be served by a single center. He showed that it can be reduced to a computationally finite one.

The vertex restricted p-center problem is considered by Toregas, Swain, ReVelle, and Bergman [115]. A solution procedure is given which relies on solving a sequence of set covering problems, each corresponding to a specified radius r.

A recent paper by Halpern and Maimon [51] suggests a comparative framework for analyzing p-center algorithms given in [12], [13], [57], [66], and [81], and shows how these algorithms fit into the framework.

p-Center Problem on a Tree Network

A number of solution procedures have been given for the p-center problem on a tree network. We now discuss these procedures.

Handler [56] considered the continuous and absolute p-center problems on a tree network for the special case of $p = 2$, and obtained two similar $O(n)$ algorithms. The gist of his approach is first to find the absolute 1-center of T, say x^*, and then to split the tree at x^*, obtaining two disjoint subtrees T_1 and T_2. Finding the absolute 1-center of each T_i, say x_1^* and x_2^*, determines an absolute 2-center of T.

An algorithm of complexity $O(n^2 \log n)$ is described by Kariv and Hakimi [66] for finding the absolute p-center of a vertex weighted tree network. They showed that r_p is one of $n(n-1)/2$ possible values, namely, the numbers $\alpha_{ij} = w_i w_j d(v_i, v_j)/(w_i + w_j)$ for each combination of vertices v_i, v_j. The algorithm computes all these numbers, arranges them in increasing order, and performs a binary search on this list of numbers. The search relies on an $O(n)$ algorithm which solves an r-cover problem for each value of r chosen from the ordered list $\{\alpha_{ij}\}$.

Tansel, Francis, Lowe, and Chen [113] considered the nonlinear p-center problem in the presence of distance constraints which impose upper bounds on the distance of any vertex to its nearest center. Nonlinearity is obtained by replacing each weight w_i by a strictly increasing and continuous function f_i of the distance $D(v_i, X)$. Provided the order of making certain inverse calculations involving the f_i can be ignored, they give an algorithm which solves the problem for all p, $1 \leq p \leq n$, of $O(n^4 \log n)$. The covering algorithm of [113] is $O(n^2)$, and was developed primarily for the purposes of establishing a number of duality results constructively. The use of an $O(n)$ covering algorithm would give an $O(n^3 \log n)$ algorithm to solve the p-center problem for all p.

Chandrasekaran and Daughety [8] gave a method to solve the continuous p-center problem on a tree network. First, they provided an $O(n)$ procedure for solving the r-cover problem. Then they provided a method to compute r_p. A further refinement of the method is given by Chandrasekaran and Tamir [10]. They proved that r_p is determined by one of the numbers $d(t, t')/2k$, where t and t' are any two vertices and k is any integer between 1 and p. The total computational effort for finding r_p and applying the covering algorithm is of $O((n \log p)^2)$.

A somewhat different approach, which relies on finding a clique cover of a related intersection graph, is given by Chandrasekaran and Tamir [9]. The intersection graph G_r, defined with respect to a given radius r, has nodes corresponding to demand points (which are not necessarily vertices) and arcs connecting pairs of nodes whenever the corresponding pairs of demand points can be jointly covered by a single center within a radius of r. Once G_r is formed, finding a clique covering of G_r provides a solution to the r-cover problem. The procedure is repeated for different values of r until a smallest value of r is found for which the clique cover solution generates at most p cliques. The computational effort is polynomial in the number of demand points and the number of potential center locations. In particular, the computational effort for finding a clique cover of G_r is polynomial due to the fact that G_r is *chordal* (i.e., for any circuit of order at least four there exists an arc, not of the circuit, which connects two nodes of the circuit). For chordal graphs, algorithms of linear order have been developed (see [32], [97]) for finding a clique cover.

Megiddo, Tamir, Zemel, and Chandrasekaran [79] developed a tree decomposition scheme to find the kth longest path in a tree in $O(n \log^2 n)$ time. Using their method, they improved the time complexity of the earlier algorithms to $O(n \log^2 n)$ for the cases where either the demands or the centers or both are restricted to vertices of a tree network. The bound is applicable for both the unweighted and weighted cases. For the continuous p-center problem, their algorithm is of $O(n \min(p \log^2 n, n \log p))$.

Tamir and Zemel [110] considered the unweighted p-center problem on a tree in a more general setting with "supply" and "demand" sets Σ and Δ, each consisting of a collection of finite number of disjoint, closed and connected subregions of T, some of which may possibly consist of just one point. They presented a polynomial algorithm which confines the search for r_p to a finite set R consisting of the distances between any "extreme" point of a subregion in Σ and any "extreme" point of a subregion in Δ. The algorithm is based on solving the related covering problem for various values of r chosen from R. Using the special data structure based on the decomposition scheme of [79], the computation of R is bypassed, resulting in an $O(n \log^2 n)$ bound for the case with both Σ and Δ discrete, and an $O(n \min\{p \log^2 n, n \log p\})$ bound if both Σ and Δ contain a full edge.

Duality for the p-Center and the Covering Problem

A number of duality results have been established in the literature on various versions of the covering problem and the p-center problem. The dual of the covering problem is to choose the maximum number of demand points no two of which are coverable by a common center. The dual of the p-center problem is to choose $p + 1$ demand points such that the minimum of the 1-radii computed with respect to all pairs of the chosen demands is as large as possible. For general networks, the primal objective value is always bounded below by the objective value of the corresponding dual problem. For the case of tree networks, equality is obtained at optimality (though this is not necessarily so for cyclic networks). Each dual problem can be given a physical interpretation as in [113]. In an attacker-defender context, the loss function version of the p-center problem can be interpreted as a (primal) defender's problem of locating troops so as to minimize the maximum loss, given some single vertex will be attacked. The (dual) attacker's problem can be interpreted as one of choosing a collection of vertices to threaten before attacking. The attacker's threat forces the defender to disperse his troops, as he cannot tell which vertex will be attacked until the attack occurs. Duality results on tree networks are a consequence of the property that the intersection graph of a family of subtrees of a tree is a chordal graph (see Buneman [5]).

The earliest duality result on the *covering* problem we know of was proven by Meir and Moon [80] in a graph-theoretic context for the case where both centers and demands are restricted to the vertices of a tree network of unit edge lengths, and where the cover radius is a nonnegative integer k for each vertex. Cockayne, Hedetniemi, and Slater [14] extended the results of [80] to the case when the cover radius for the ith vertex is a nonnegative integer k_i. Shier [100] generalized the result of [80] to the (unweighted) continuous problem where each point in the tree network is a demand point as well as a potential center location. Chandrasekaran and Tamir [9] arrived at the duality result for the weighted case with demands and centers restricted to finite subsets of the tree network; the set of demands is not necessarily equal to the set of potential centers. Further, it was shown in [9] that three other versions of the problem are special cases of this more general model. Tansel, Francis, Lowe, and Chen [113] extended the duality result on the cover problem to the nonlinear case in the presence of distance constraints of the form $D(X, v_i) \leqslant u_i$, $i \in I$. In their work, the cover radius for the ith vertex is either the value of the inverse function f_i^{-1} evaluated at r, or the upper bound u_i, whichever is smaller, and each point in the tree network is a potential center location. The method of proof in [113] is constructive.

Duals of various versions of the *p-center* problem have also been considered. Shier [100] is the first to discuss a dual problem to the continuous (unweighted) p-center problem. For this version, the dual problem becomes that of choosing $p + 1$ points in the tree network so that the closest two are as far apart as possible. Chandrasekaran and Tamir [10] established that Shier's duality result holds when one restricts the set of demands and potential centers to any subset S of T. Also, Chandrasekaran and Tamir

[9] extended the duality result to the weighted case. Tansel, Francis, Lowe and Chen [113] established duality results on the distance-constrained version of the p-center problem for the nonlinear case. Kolen [68] gave an extension for the nonlinear "roundtrip" version of the problem.

Shier [100] solved the dual of the continuous (unweighted) p-center problem for $p = 2, 3$. Shier's algorithms for solving the dual for $p = 2$ and 3 are quite similar to algorithms of Handler for solving primal 1-center [52] and 2-center [56] problems respectively. Chandrasekaran and Daughety [8] developed a solution procedure for the same dual problem for $p \geqslant 1$. They first solve the problem of locating the maximum number of points in T such that any two of them are at least a distance of λ apart. The procedure relies on using the algorithm for different values of λ until the number of points found is $p + 1$, and a slightly larger λ generates p or less points. Tansel, Francis, Lowe, and Chen [113] gave a solution procedure for the dual of the distance-constrained nonlinear p-center problem. Given the value of r_p, they identify the dual solution by applying their covering algorithm with a radius r which is "slightly" smaller than r_p.

Recently, Francis and Lowe have considered a (primal) distinct cover problem [24] for which a minimum number of centers are to be located so that some center covers each vertex, and pairs of vertices specified as "related" are covered by *distinct* centers. The motivation for studying the problem comes from the need to provide "extra" or "redundant" coverage for related vertices. They state a dual problem, which is to find a subset K of vertex indices of maximum cardinality such that for any unrelated v_p and v_q with $p, q \in K$, $d(v_p, v_q) > b_p + b_q$, where b_i is the cover radius for v_i, $i = p, q$. They identify two critical assumptions, say (A-1) and (A-2), as follows.

(A-1): Vertices can be partitioned into families such that any two vertices are related if and only if they are in the same family.

(A-2) Vertices can be numbered so that whenever v_s and v_t are in the same family and $s < t$, then for any v_p unrelated to v_s and v_t it is true that $d(v_s, v_p) \leqslant b_s + b_p$ implies $d(v_t, v_p) \leqslant b_t + b_p$.

Given (A-1) and (A-2), they prove a weak duality theorem for a general network: for a tree network they prove a strong duality theorem as well, and give $O(n^2)$ algorithms to solve each problem. When assumptions (A-1) and (A-2) are not made, they prove the primal problem is NP-complete-even for a tree network.

Tamir [107]–[109] and Kolen [68] have developed mathematical programming based approaches to the covering problem and related problems on a tree network. These methods exploit the "balancedness" of the constraint matrix associated with an integer programming formulation of the problems. The balancedness of the matrix guarantees that the linear programming relaxation of the resulting integer programs provides integer solutions.

3. The p-Median Problem

Let $g(X) = \sum\{w_i D(v_i, X) : i \in I\}$ for $X \subset N$. The p-median problem is to find a set X^* of p points for which $g(X^*) = \min[g(X) : |X| = p, X \subset N]$.

Any set X^* of p points that minimizes g is called an ***absolute p-median*** of N. If each member of X is restricted to a vertex location, the resulting problem is called a ***vertex restricted p-median problem***. Due to a result by Hakimi [42], [43] there exists an absolute p-median consisting entirely of vertices of N. For this reason, the distinction between the vertex restricted and unrestricted versions is insignificant.

The p-median problem arises naturally in locating plants/warehouses to serve other plants/warehouses or market areas. The problem is also motivated by ReVelle, Marks, and Liebman [96] as an example of a public sector location model where vertices represent population centers and facilities represent post offices, schools, public buildings, and the like: weights are typically proportional to the amount of "traffic" between medians and vertices.

The 1-*Median of a General Network*

Hakimi [42] appears to be the first to define an absolute median. The median can be found by summing each row of the weighted-distance matrix and choosing the vertex whose row sum is the minimum. This procedure takes $O(n^3)$ operations to compute the distance matrix followed by $O(n^2)$ operations to find the median.

The 1-*Median of a Tree Network*

For tree networks, more efficient algorithms can be devised to find a median. An $O(n)$ "tree-trimming" algorithm was given by Hua Lo-Keng and others [63]—in 1962—and independently by Goldman [36]. The algorithm reduces the search to successively smaller subtrees until a median is found. At each stage, one chooses an arbitrary tip vertex (a vertex of degree one) of the current tree. If the (modified) weight of the selected vertex is at least as large as half the sum of all weights, a median is found. Otherwise, that tip vertex is eliminated from further consideration together with the edge incident to it, and its weight is added to the weight of the adjacent vertex. The procedure is repeated with the new (reduced) tree. The algorithm does not require the use of arc lengths, but uses only the incidence relationships and weights.

Goldman's algorithm is based on a "localization theorem" proved by Goldman and Witzgall [41]. The theorem provides sufficient conditions for a subset of N to contain a median. Given a compact subset S of N, S contains at least one median if S satisfies the following two conditions: (i) S must be a "majority" set, meaning that the sum of the weights corresponding to vertices in S must be at least as large as half the sum of all weights; (ii) S must be "gated" in the sense that for each $t \in N - S$ there must exist a unique (closest) point t' in S to t such that for every $s \in S$ it is true that $d(t,s) = d(t,t') + d(t',s)$. (Condition (ii) is always satisfied when S is convex as well as compact, and illustrates the use of a convexity result in convex analysis, as is pointed out in [41].) Goldman's algorithm in essence is a repeated application of this theorem to a tree network. Goldman [37] also proposed an "approximate" localization theorem which somewhat relaxes the second condition and guarantees the existence of a point in S that approximates an actual median.

A median of a tree was shown to be the same as a "centroid" of the tree by Zelinka [117] for the unweighted case, and by Kariv and Hakimi [67] for the weighted case. To define a centroid, consider the subtrees $T_1, \ldots, T_{k_i}$ obtained by removing vertex v_i from T. Let $W(T_j)$ be the sum of the weights of the vertices in T_j, and define $\overline{W}(v_i)$ to be the maximum of $W(T_j)$ for $1 \leq j \leq k_i$. A vertex v_t which minimizes $\overline{W}(v_i)$ over all v_i in V is said to be a *centroid* of T. The location of a centroid is independent of the distances and can be found by using only the incidence relations. Goldman's earlier algorithm in essence finds a centroid of T. The generalized algorithm of Rosenthal, Hersey, Pino, and Coulter [99] also finds a centroid of T by making only two traversals of the vertices. All these algorithms are of $O(n)$, and solve the 1-median problem without having to compute the distance matrix.

We now consider some generalizations of the 1-median problem. Minieka [82] defined the *general absolute median* of a network to be any point on the network that minimizes the sum of (unweighted) distances from the point to the most distant point on each edge. Minieka showed that the general absolute median can be strictly interior to an edge; hence, the search cannot be confined solely to vertices of N. Minieka [84] also considered various versions of "conditional" 1-median problems, which are analogous both in formulation and solution approach to the conditional 1-center problems discussed earlier.

Goldman [39] took advantage of the convexity properties of trees to develop an $O(n)$ algorithm for localizing the optimum to a single edge of the tree for the case of the polynomial 1-median problem where the objective function is defined by the sum of polynomial functions of distances from any vertex to the median. His $O(n)$ algorithm finds the global minimum for the quadratic case (i.e., the sum of weighted squared distances). Goldman's algorithm is based on the computation of directional derivatives along the edges.

Slater [105] gave another generalization of the 1-median problem. In this generalization, each "demand" is a collection of vertices. The problem is to find a vertex such that the sum of the distances from that vertex to a nearest element of each collection is minimum. Slater showed that the set of vertices that solve this problem forms a connected path in T. For a general network, the problem can be solved by constructing a matrix that specifies the distances from each vertex to a nearest element of each collection, summing each row of this matrix and then choosing a vertex whose row sum is minimum.

p-Median of a Network and Vertex Optimality

Here we consider certain generalizations of Hakimi's vertex-optimality result for the p-median problem.

Levy [72] proved that the (vertex-optimal) result holds when the weights w_i are replaced by concave cost functions $c_i(\cdot)$ of the distance between v_i and its nearest median.

Goldman [35] generalized the result to the case of a "two-stage" commodity. More specifically, one distinguishes a vertex as being a source or a destination. Let (v_s, v_d) be a source-destination pair, and let x_i and x_j be the nearest medians to v_s and v_d, respectively. Then the cost of transferring the commodity from source v_s to destination v_d is the sum of three transport costs, namely, $w_{sd}d(v_s, x_i) + \tilde{w}_{sd}d(x_i, x_j) + w^*_{sd}d(x_j, v_d)$. In general, if $X = \{x_1, \dots, x_p\}$ is a median set, one does not know which median is the nearest to v_s or v_d; hence, the cost associated with a source-destination pair (v_s, v_d) is

$$g_{sd}(X) \equiv \min_{x_i, x_j \in X} \left[w_{sd}d(v_s, x_i) + \tilde{w}_{sd}d(x_i, x_j) + w^*_{sd}d(x_j, v_d) \right]$$

and the objective to be minimized is $g(X) = \sum[g_{sd}(X) : v_s, v_d \in V]$. Goldman showed that there exists an optimal X^* contained in V, and conjectured that the result holds for any multistage problem.

Hakimi and Maheshwari [44], in response to Goldman's conjecture, proved the vertex optimality result for the case of multiple commodities that go through multiple stages with the cost of transport from one stage to the next a concave nondecreasing function of the distance. In this general model, M_{sd} denotes the set of commodities to be transferred from source v_s to destination v_d, and $t(m)$ denotes the number of stages commodity $m \in M_{sd}$ is to go through. Given a median set $X = \{x_1, \dots, x_p\}$ let $y_m^r \in X$ be the location where stage r processing takes place for commodity m, $r = 1, \dots, t(m)$. The cost of transferring commodity m from v_s to v_d via the intermediate stages is given by

$$\hat{g}_m(Y) = C^0_{m,v_s}\left[d(v_s, y_m^1)\right] + C^1_{m,y_m^1}\left[d(y_m^1, y_m^2)\right] + \cdots + C^{t(m)}_{m,y_m^{t(m)}}\left[d(y_m^{t(m)}, v_d)\right],$$

where C^0_{m,v_s} and the C^i_{m,y_m^i} are concave and nondecreasing functions which depend on the commodity, the stage of processing, and the location of the stage of processing. Also it is assumed, for a fixed commodity, stage of processing, and distance to the next stage of processing, that the functions are invariant along any edge of the network. With $Y \subset X$, $|Y| = t(m)$, the minimum cost of transfer for commodity m is given by $g_m(x) = \min[\hat{g}_m(Y): Y \subset X, |Y| = t(m)]$. The cost of transferring all commodities from v_s to v_d is obtained by summing over all commodities, that is, $g_{sd}(X) = \sum[g_m(X): m \in M_{sd}]$. The total cost of the system is obtained by summing the cost $g_{sd}(\cdot)$ over all source-destination pairs, that is, $g(X) = \sum[g_{sd}(X): v_s, v_d \in V]$.

Wendell and Hurter [116] considered another form of the problem where the transportation cost functions are permitted to differ from edge to edge. The transport cost on any edge is a nondecreasing concave function of the distance. They proved that it is *sufficient* to consider the vertices of the network under such a cost structure. Furthermore, they obtained conditions under which it is *necessary* for the solution to occur at the vertices. In particular, they showed that nonvertex optimal locations can occur in any given edge only when transportation costs are linear with distance over that edge and in that case, when and only when the slopes of these linear cost functions are in a special relation.

Solution Approaches

Kariv and Hakimi [67] showed that the p-median problem on a general network is NP-hard. For the case of *tree* networks, however, algorithms of polynomial complexity have been developed. Matula and Kolde [76] suggested an $O(n^3p^2)$ algorithm for finding the p-median of a tree network. Kariv and Hakimi [67] proposed an $O(n^2p^2)$ algorithm for the same problem.

For general networks, a number of solution procedures have been developed recently, all based on the vertex-optimality result. Their common characteristic is that they all confine the search to vertex locations. The solution procedures tend to be based on mathematical programming relaxation and branch-and-bound techniques.

Successful implementation of dual-based techniques for the p-median problem and the related uncapacitated plant location problem have been reported recently. Cornuejols, Fisher, and Nemhauser [15] used a dual-based multi-phase approach for generating and verifying near-optimal solutions to both the p-median problem and the plant location problem. Narula, Ogbu, and Samuelsson [93] presented a branch-and-bound scheme which relies on obtaining the bounds by solving a Lagrangian relaxation of the p-median problem using subgradient optimization. By dualizing the p-median problem with respect to a different set of constraints than is done in [93], Mavrides [77] generates a Lagrangian relaxation of the problem which illustrates the connection between the plant location problem and the p-median problem (see also [91]).

Erlenkotter [21] developed a dual-ascent method for solving a Lagrangian relaxation of the plant location problem. Erlenkotter's approach appears to be the most computationally successful to date for solving the uncapacitated plant location problem. Independently, Bilde and Krarup [4] developed a related formulation of the problem. Galvao [30] modified Erlenkotter's approach to solve the p-median problem.

In a recent paper, Fisher [22] discusses Lagrangian relaxation methods for solving the plant location problem as well as other integer programming problems. Geoffrion and Graves [33] used Benders decomposition on a large scale distribution problem. Magnanti and Wong [75] consider alternative and related formulations of the plant location problem in the context of Benders decomposition.

Stochastic Networks

A number of probabilistic versions of the p-median problem have been considered. Various p-median problems have been modelled where the edge lengths are random variables and/or the demands for service (located at vertices) are random variables.

Mirchandani [87] stressed the importance of modelling stochastic elements in certain location decisions and introduced an underlying framework for these location decisions. Frank [27], [28] considered the one-median problem when the vertex weights are random variables but the edge lengths are deterministic. The absolute expected median is defined to be a point in the network which minimizes the sum of the expected weighted distances to the vertices. The maximum probability absolute median is defined to be a point in the network where, given a real number R, the probability that the sum of weighted distances exceeds R is minimal. Techniques to find the optimum points are discussed and it is shown that the maximum probability absolute center is not always at a vertex of the network. Mirchandani and Odoni [88], [89] considered the p-median problem where network edge lengths are random variables and vertex weights are deterministic.

They introduced a utility function and defined the concept of an "expected optimal p-median." They demonstrated that if the utility function is convex and nonincreasing (in travel times), then there exists an expected optimal p-median on the vertices of the network. Berman and Larson [1] extended the vertex optimality result to the case where the availability of servers is a random variable. Mirchandani and Oudjit [90] studied the two-median problem on a tree network with deterministic weights and random edge lengths. They showed that when the edge lengths are deterministic, the optimal one-median lies on a single path between the optimal two-median pair. They then showed that this result does not always hold when the edge lengths are random variables. They give a "link deletion" method for solving the two-median problem on the stochastic tree network.

Recently, Berman and Larson [2] and Berman and Odoni [3] have considered the case of mobile servers on a stochastic network. In [2] a single server problem is considered where demands for service arise at vertices of the network according to a Poisson process. As demand occurs, the server is to be dispatched to the demand. Two models are considered. In the first model, the demand is rejected if the server is busy. The objective is to minimize the weighted sum of mean travel time plus cost of rejection. It is shown that an optimal solution is obtained at a vertex of the network. In the second model, the demand enters a first-come-first-served queue if the server is busy. The objective is to minimize the mean queueing delay plus the mean travel time. In this model, the optimal solution may not be at a vertex.

In [3], the demands are deterministic but the travel times on the network edges are stochastic (Markovian). Servers can be relocated at a cost. The objective is to minimize weighted travel times and server relocation costs. It is shown that when the relocation costs are concave functions of the travel time to new locations, the problem has an optimal solution at the vertices of the network. The resulting location-relocation problem is modeled as an integer programming problem.

Mirchandani, Oudjit, and Wong [91] provide a generalization of the modelling framework given in [87] and introduce the concept of a "stochastic multidimensional network". Their model allows for the possibility of multiple services along with stochastic demands and stochastic travel times. They make use of the following result given in Oudjit [94]: if the travel time from a point interior to an edge of the network to an end vertex of the edge is proportional to the distance from the interior point to the end vertex, and if the transport costs are concave in travel time, then an optimal solution exists on the vertices of the network. Using this result, they give mathematical programming formulations of the problems.

Chiu [11] considers several interesting generalizations of the 1-median problem incorporating queueing aspects. In a problem he terms the stochastic loss problem, demands for service are generated at vertices of the network by a time-homogeneous Poisson process. A single facility is to be located to station n mobile servers. A call for service is "lost" at a (nonnegative) cost if, upon arrival, all n servers are busy. The facility is to be located so as to minimize the sum of expected travel time (the 1-median model) and the weighted cost of the loss. Chiu shows that the optimal server location coincides with a location that minimizes the expected travel time. In a second related problem, termed the stochastic queue median problem, if all servers are busy, then a call (instead of being lost) enters an infinite capacity queue operating on a first-come-first-serve basis. The facility is to be located so as to minimize the expected response time, defined as the sum of the expected travel time and the expected queueing delay time. For the case where the network is a tree, Chiu shows that the response time function is convex (when finite) and, for the tree case, exploits convexity to develop an efficient solution procedure. Chiu generalizes the stochastic queue median problem, for a tree network, to the allow demands to be continuously distributed on arcs (as well as discretely distributed on vertices) and obtains parallel results.[1]

[1] We would like to thank, collectively, the many colleagues who have contributed to our bibliography, and provided comments on our paper. Also we wish to apologize, in advance, to authors whose work we have overlooked. The field of Network Location Theory is growing so rapidly that continually updating our bibliography would postpone the publication of our paper indefinitely.

This research was sponsored in part by the National Science Foundation, NSF Grants ECS-8007104 and ECS-8007110.