Title

Mixed planar and network single-facility location problems

Permalink
https://escholarship.org/uc/item/06z4t4c4

Journal
Networks, 68(4)

ISSN
0028-3045

Authors
Drezner, Zvi
Scott, Carlton H
Turner, John

Publication Date
2016-12-01

DOI
10.1002/net.21698

Peer reviewed

Mixed Planar and Network Single-Facility Location Problems

Zvi Drezner
Steven G. Mihaylo College of Business and Economics
California State University - Fullerton
Fullerton, CA 92834
e-mail: zdrezner@fullerton.edu

Carlton H. Scott and John Turner
Paul Merage School of Business
University of California - Irvine
Irvine, CA 92697
e-mail: {chscott, john.turner}@uci.edu

Abstract

We consider the problem of optimally locating a single facility anywhere in a network to serve both on-network and off-network demands. Off-network demands occur in a Euclidean plane, while on-network demands are restricted to a network embedded in the plane. On-network demand points are serviced using shortest-path distances through links of the network (e.g., on-road travel), whereas demand points located in the plane are serviced using more expensive Euclidean distances. Our base objective minimizes the total weighted distance to all demand points. We develop several extensions to our base model, including: (i) a threshold distance model where if network distance exceeds a given threshold, then service is always provided using Euclidean distance, and (ii) a minimax model that minimizes worst-case distance. We solve our formulations using the "Big Segment Small Segment" global optimization method, in conjunction with bounds tailored for each problem class. Computational experiments demonstrate the effectiveness of our solution procedures. Solution times are very fast (often under one second), making our approach a good candidate for embedding within existing heuristics that solve multi-facility problems by solving a sequence of single-facility problems.

Key Words: Facility location; Network; Planar; On-network; Off-network; Optimal algorithm; Emergency medical services.

1 Introduction

We consider a mixed planar and network location problem that optimally locates a single facility anywhere on a network (e.g., beside a road) to serve both on-network and off-network demands.

Since heuristics for solving multiple-facility problems often rely on solving a sequence of single-facility problems, having a fast method for solving the single-facility variant is very important, and is the focus of our work. In our problem, we generate demands at two sets of points: one is located anywhere in the plane, and another is restricted to a network embedded in the plane (i.e., its nodes have planar (x, y) coordinates and links are straight lines connecting the nodes). We assume that the network is connected, meaning that a path exists between any two nodes of the network. Some, but not necessarily all, pairs of nodes are connected by links. The distance between a point on the network and a point in the first (off-network) set of demand points is the Euclidean distance between the two points, whereas the distance between any point on the network and a point in the second (on-network) set of demand points is the shortest-path distance along the network. As is customary in location problems such as the single facility Weber problem [36, 75, 76], its multiple facility version [12, 13, 49] and the p-median problem [21, 22, 28, 47, 61], a weight is associated with each demand point and the transportation cost of providing service from a facility to a demand point is the weight multiplied by the distance between that point and the facility. In addition, Euclidean travel is more expensive than traveling on the road network. Otherwise, there is no reason to use network travel.

In addition to our base model, which locates a facility anywhere on the network so that the total cost of serving both on-network and off-network demands is minimized, we explore several model variants. These variants allow for such practical considerations as (i) switching to use costlier Euclidean distances for reaching some on-network demand points when the facility is located far away (threshold distance model), and (ii) producing robust solutions that ensure no demand point receives disproportionately bad service (minimax model). In each case, we develop tailored bounds that are used by our optimization procedure. Note that, in all cases, the facility should be located on the network, otherwise service through links of the network is not possible.

A related problem is the vehicle routing allocation problem [3, 64, 72] where a tour is designed on the network and off-network demand is serviced at a close point on the route.

To the best of our knowledge, this particular problem has not been investigated. A few papers incorporate both distances along links of the network and off-network distances. Location on a network which consists of nodes and "mega-nodes" which are areas in the plane where demand points in such areas travel through a discrete set of gates at the periphery of the area was considered in [2]. The simultaneous location of a facility and a linear transit line was considered in [30, 31, 43]. A multiple-facility planar location problem in the presence of an alternative rapid transit system was considered in [16]. They showed that for polyhedral distances the problem is reduced to finding a p-median. A transportation network where customers residing off the network first travel off-network to an on-network bus stop using the Manhattan distance was investigated in [46]. Finally, similarities and differences between planar and network location problems were analyzed in [65, 66].

Our model has a number of applications; consequently, we review the literature in several relevant application domains. Consider the decision of where to build a hospital, and the transportation-related issues that must be taken into account. When a call for service is received, paramedics are dispatched from the hospital either by ambulance or by helicopter, depending on the severity level of the patient. Moreover, some calls may originate in remote areas where ground transportation cannot reach the patient, and thus a helicopter must be dispatched. Our model captures this key tradeoff between using helicopters or ground ambulances to transport patients to a hospital (generally speaking, helicopters travel faster than ground ambulances and can service patients over a wider geographic area, but are more expensive to operate). In particular, the threshold distance version of our problem, which we will present in Section 4.1, is directly applicable to emergency settings since it checks how far away from the facility each on-network demand point is located and switches to use the costlier off-network Euclidean distance metric when on-network distances are not within the time standard.

The emergency systems planning literature includes both continuous and discrete location models. Continuous models such as ours allow the emergency facility (e.g., hospital) to be located at any point in a continuous set (e.g., along a link of a road network). Continuous location is most applicable early on in the decision-making process to identify a brand new site that has not yet been built. Continuous emergency facility location models that maximize coverage within a service area include [29, 50, 67]; however, these papers only consider a single mode of transportation to/from the hospital. For additional reviews of emergency facility location the reader is referred to [54, 14].

In contrast, discrete models are provided with a set of candidate (x, y) locations, and it is the model's job to select one or more locations from this discrete set of points. Within the discrete location framework, Erdemir et al. [42] consider three coverage options: only ground coverage, only air coverage, or joint coverage of ground and air in which the patient is transferred from an ambulance to an emergency helicopter at a transfer point. They develop two model variants, the Set Cover Backup Model (SCBM) which is an extension of the Location Set Covering Problem (LSCP) of [70], and the Maximal Cover for a Given Budget Model (MCGBM) which is an extension of the Maximal Covering Location Problem (MCLP) of [23]. Both model variants use discrete sets of candidate locations for ground ambulances, helicopters, and transfer points. The resulting mixed-integer programming formulations are hard to solve at scale, for which the authors develop an efficient heuristic. A combination of hospital, ground ambulance, and helicopter location using a discrete location model, but without helicopter to ground ambulance transfer points was considered in [9]. To find good solutions, they use a mixed-integer linear program with an iterative switching heuristic. More recently, Cho et al. [19] simultaneously located trauma centers and heliports using a large-scale mixed-integer quadratic program, and observed a 10-15% increase in the number of patients receiving care in a timely manner.

In another application, a distribution center (for example, FEDEX, UPS, the U.S. Postal Service) needs to be established to serve a set of demand points [52]. Some demand points are served by air, while some are served by truck or rail. High-priority packages must be transported by air, while low-priority packages can be transported by truck or rail, which is cheaper. The existing literature tends to model the package depot location problem as a discrete location problem, in which a small number of depots are chosen from a candidate set and specific tours (e.g., one per truck) are assigned to each depot. In such a context, the package depot location problem can be viewed as a location routing problem, for which [64] provide a survey of the literature and relevant models. Wasner and Zäpfel [74] is a representative example, in which the authors develop a large mixed-integer program and related heuristic to solve a multi-depot hub-location routing problem for a mid-sized Austrian parcel service. In contrast, our focus is on a continuous (location anywhere on the network) model, which is most appropriate for deciding where to put a new depot when no candidate sites have been pre-selected by other means. Thus, our model contributes to the literature on package depot location at the strategic level.

The use of aerial drones to deliver packages is a new and exciting technology [26, 63, 73]. At present, no company is doing this, but both Amazon and Google have speculated that they may start by 2017 (see [63] and various reports on the web). Consider two classes of consumers that need an item urgently: those that drive to the store to pick the item up themselves (or pay for a courier service to deliver the item to them), and those that request drone delivery. Shipments that are time-critical (e.g., hot cooked food, medical supplies like vaccines or anti-venom for treating snake bites, replacement parts for a broken machine) are most suited for this case. Although we anticipate that the cost of a drone delivery, at the margin, may still exceed that of an on-road delivery, our model trades off the cost of using network versus Euclidean distances, and can be used to model this application regardless of how the future economics of drone delivery plays out. Moreover, our model can easily be extended to the case where, at each demand location, there are customers that are heterogeneous in their disutility (cost) of driving to a store to pick up their order (simply co-locate a number of demand nodes with differing magnitudes of on-network costs, and apply the model presented in this paper).

In yet another application, a military base (temporary or permanent) needs to be located. The operations conducted from this base are performed either by air (planes, helicopters) or by ground transportation depending on the operation. Some operations must be done by air and some must be done on the ground. Once again, there are pros and cons of modeling military base location as a discrete versus continuous location problem. The literature appears to be sparse in either case. Three separate discrete location model variants to select aircraft alert sites for the defense of important national areas of interest identified by the U.S. Department of Defense are discussed in [4]. We could not find any continuous location examples in the context of military base location. However, we expect that the continuous location modeling approach would be most appropriate when the strategic decision of where to place the base must be completed well in advance of the operational decisions that describe the specific missions (by ground or air) that will be run from the base. In this context, the mass of each demand point in our model can be viewed as a probability of needing to run a mission in that vicinity, and the objective minimizes expected operational costs.

The paper is organized as follows. In Section 2 we formulate the base model, and in Section 3 we present algorithms for finding an optimal solution. In Section 4, we describe some important extensions to our base model, as well as how bounds and solution approaches can be tailored for these cases. Finally, we present computational experiments in Section 5 and conclude with a discussion and suggestions for future research in Section 6.

2 Formulation

Demand is generated at two sets of points, with points in the set N_N restricted to locations on the network, and points in the set N_E located anywhere in the plane. We wish to locate a facility at a point $X = (x, y)$ on the network, which may be at a node or anywhere along a link. We denote the shortest-path distance from X to a demand point $i \in N_N$ as $d_i^N(X)$, and the Euclidean (straight-line) distance from X to a demand point $i \in N_E$ as $d_i^E(X)$. So that we may express distances in general terms, we also define the general distance $d_i(X)$ between X and any demand point $i \in N = N_N \cup N_E$, such that $d_i(X) = d_i^E(X)\ \forall i \in N_E$ and $d_i(X) = d_i^N(X)\ \forall i \in N_N$. We use the weight w_i to denote the population (mass) of demand point i, and define c as the ratio between the transportation cost per unit of Euclidean distance and the transportation cost per unit of network distance. In typical applications $c > 1$ because it is more expensive, on a per-unit of distance basis, to travel from X to a demand point in N_E, than to a demand point in N_N (for example, transporting a patient by helicopter is more expensive than by ground ambulance). The objective function, which minimizes total cost by finding the best location X anywhere on the network, is

$$F(X) = \sum_{i \in N} w_i d_i(X) = \sum_{i \in N_N} w_i d_i^N(X) + c \sum_{i \in N_E} w_i d_i^E(X). \tag{1}$$

It is sometimes convenient to decompose $F(X)$ into two components: the total network and total Euclidean transportation costs, which we denote as $F_N(X) = \sum\limits_{i \in N_N} w_i d_i^N(X)$ and $F_E(X) = c \sum\limits_{i \in N_E} w_i d_i^E(X)$, respectively. This allows us to write (1) as $F(X) = F_N(X) + F_E(X)$.

If $N_E = \emptyset$, i.e., all demand points are on the network and only network distances are used, an optimal solution to (1) is on a node of the network [47, 48]. However, this may not be the case when $N_E \neq \emptyset$. Consider the example in Figure 1 of a network consisting of an equilateral triangle with all sides of length of 1 and demand points located at its vertices as well as its center. The vertices of the triangle are members of N_N, while the demand point at the center of the triangle is a member of N_E. All demand points have a weight of 1 and some ratio c is used. The value of the objective function at each node (vertex) is $2 + \frac{\sqrt{3}}{3}c$, and its value at the center of each side of the triangle is $2.5 + \frac{\sqrt{3}}{6}c$. The value of the objective function at a center of a side is lower than its value at a node when $c > \sqrt{3}$, which proves that for $c > \sqrt{3}$ the solution is not on a node. It can be shown that the optimal solution is in the interior of a link for $c > \frac{2}{3}\sqrt{3}$, but the more complicated analysis is not necessary to illustrate our point.

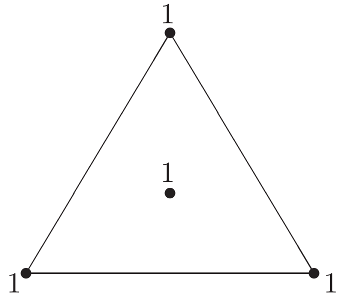


Figure 1: An example showing that the optimal solution may be in the interior of a link.

3 Solution Procedure

The optimal location of a facility placed anywhere on the network can be found by the "Big Segment Small Segment" global optimization algorithm proposed by [6]. This is a branch-and-bound procedure, and can be used to find the optimal solution to any given relative accuracy $\epsilon > 0$. The algorithm is as follows:

The Big Segment Small Segment Algorithm

Step 1 (Initialization): Initialize a set of segments S, with one segment for each link in the network. For each segment $s \in S$, calculate an upper bound (UB) and a lower bound (LB) on the optimal value that can be achieved when the solution X is restricted to lying on the segment s. Denote the smallest (tightest) UB as UB^*, and the segment on which this best bound is achieved as s^*.

Step 2 (Pruning): Discard all segments for which $LB \geq UB^*(1 - \epsilon)$.

Step 3 (Termination Criterion): If no segments remain ($S = \emptyset$), terminate. The best solution found has value UB^* and is within the desired relative accuracy of ϵ from the global minimum.

Step 4 (Refinement / Branching): Otherwise, bisect the "big segment" s^* at its midpoint into two equally-sized "small-segments" s_1 and s_2. Update the list of segments by replacing the big segment with the two new small segments: $S \leftarrow S \cup \{s_1, s_2\} \setminus \{s^*\}$. Calculate LB and UB for the two new segments, and update UB^* and s^* accordingly. Go to Step 2.

In order to implement the algorithm, we require upper and lower bounds for the optimal value of (1) when X is restricted to a segment. Such an upper bound is easy to establish, since the value of the objective function at any point along a segment is an upper bound. For each segment, it is efficient to define its upper bound as the minimum value of the objective function evaluated at its two endpoints. In Step 1 of the algorithm, the endpoints of each segment (link) are nodes of the network. Once the values of the objective function at all nodes of the network are calculated, evaluating the upper bound on a link requires only taking the minimum among two values. We can also save for each segment the two values of the objective function at its endpoints, so that in Step 4, when the big segment is divided into two small segments and the value of the objective function needs to be computed at the center of the big segment, we already have available the values of the objective function at the endpoints of the small segments.

To establish a lower bound on the optimal value of (1) when X is restricted to a segment, we propose a bound based on DC-optimization (see for example, [8] and the references therein).

The shortest-path distance from any point along a segment to a point on the network is a concave function (see, for example, [6]), while the Euclidean distance from any point along a segment to a point in the plane is a convex function. Therefore, $F_N(X)$ is concave while $F_E(X)$ is convex, and $F(X) = F_N(X) + F_E(X)$ is neither. However, if we replace each convex Euclidean distance function $d_i^E(X)$ in $F_E(X) = c \sum\limits_{i \in N_E} w_i d_i^E(X)$ with a suitable concave underestimate, we can produce a modified objective which is a concave lower bound for $F(X)$. Using the fact that a concave function evaluated over a segment obtains its minimum at an endpoint, for any X that lies on a particular segment we then use as a lower bound the minimum value of this modified objective evaluated at the segment's two endpoints. It remains to discuss how to construct concave underestimates for $F_E(X)$. A simple concave underestimate for $F_E(X)$ can be produced by, for each segment s, pre-computing the point X_s at which $F_E(X)$ is minimized and has value v_s, and then taking $\{v_s$ if X is on segment $s\}$ as our underestimate for $F_E(X)$. Since this expression is a constant when restricted to each segment, it is concave on each segment. Although intuitive to specify, this particular underestimate for $F_E(X)$ is expensive to produce, since an optimization problem must be solved for each segment s as a pre-processing step. Instead, we suggest underestimating each $d_i^E(X)$ with its tangent line taken at any point along the segment, say at its center. Then, (a) since $d_i^E(X)$ is convex its tangent line is a valid lower bound, and (b) the tangents are all linear, and thus concave. Therefore, the modified function (1) following the substitution of Euclidean distances to points in N_E by their tangent lines is a suitable concave underestimating function, as required.

3.1 Implementation

Consider a link connecting a "left" node $L = (x_L, y_L)$ and a "right" node $R = (x_R, y_R)$ of length ℓ. A point $X(\theta)$ on the link is defined by $\theta \in [0, 1]$ as follows:

$$X(\theta) = (x(\theta), y(\theta)) = ((1-\theta)x_L + \theta x_R, (1-\theta)y_L + \theta y_R). \tag{2}$$

The point $X(\theta)$ is at distance $\theta\ell$ from the left node and $(1-\theta)\ell$ from the right node. The distance between $X(\theta)$ and a demand point X_i at some node $i \in N_N$ of the network is calculated as follows. Let d_L, d_R be the distances between X_i and the left and right nodes, respectively. These distances are calculated along the shortest path between X_i and the two nodes. The distances between

X_i and all the nodes of the network can be calculated in the preamble of the algorithm and are available. The distance between X_i and $X(\theta)$ is

$$d_i^N(X(\theta)) = \min\left\{d_L + \theta\ell, d_R + (1-\theta)\ell\right\} \text{ for } i \in N_N. \tag{3}$$

This distance is a concave function of θ, since it is a minimum of two linear functions.

The distance between $X(\theta)$ and $X_i = (x_i, y_i)$ for $i \in N_E$ is calculated directly as the Euclidean distance between the two points, i.e.,

$$d_i^E(X(\theta)) = \sqrt{\left((1-\theta)x_L + \theta x_R - x_i\right)^2 + \left((1-\theta)y_L + \theta y_R - y_i\right)^2} \text{ for } i \in N_E. \tag{4}$$

The tangent line $T_i(\theta)$ to the Euclidean distance at a point $X(\theta_0)$ is

$$T_i(\theta) \quad = \quad d_i^E(X(\theta_0)) + \left.\frac{\partial d_i^E(X(\theta))}{\partial\theta}\right|_{\theta=\theta_0} (\theta - \theta_0), \tag{5}$$

where

$$\left.\frac{\partial d_i^E(X(\theta))}{\partial\theta}\right|_{\theta=\theta_0} \quad = \quad \frac{(x_L - x_i)(x_R - x_L) + (y_L - y_i)(y_R - y_L) + \theta_0\left[(x_R - x_L)^2 + (y_R - y_L)^2\right]}{d_i^E(X(\theta_0))}.$$

By the convexity of Euclidean distance, $T_i(\theta) \le d_i^E(X(\theta))$. A segment in our algorithm is either a link or part of a link, and is defined by $0 \le \theta_1 < \theta_2 \le 1$. The tangent line is calculated at the center of the segment $\theta_0 = \frac{\theta_1+\theta_2}{2}$. The lower bound for a point $X(\theta)$ along the segment $\theta \in [\theta_1, \theta_2]$ is

$$LB = \min\left\{\sum_{i\in N_N} w_i d_i^N(X(\theta_1)) + c\sum_{i\in N_E} w_i T_i(\theta_1), \sum_{i\in N_N} w_i d_i^N(X(\theta_2)) + c\sum_{i\in N_E} w_i T_i(\theta_2)\right\}, \tag{6}$$

and the upper bound for a point $X(\theta)$ along the segment $\theta \in [\theta_1, \theta_2]$ is

$$UB = \min\left\{\sum_{i\in N_N} w_i d_i^N(X(\theta_1)) + c\sum_{i\in N_E} w_i d_i^E(X(\theta_1)), \sum_{i\in N_N} w_i d_i^N(X(\theta_2)) + c\sum_{i\in N_E} w_i d_i^E(X(\theta_2))\right\}. \tag{7}$$

As we can see, only six sums are required to calculate both LB and UB for a segment.

4 Extensions

The following are extensions of the base model presented in Section 2.

4.1 The Threshold Distance Problem

We assume that Euclidean distances are more expensive ($c > 1$), otherwise there is no point in considering the option of using network distances at all. This is the case in many practical examples since air transportation is often more costly than ground transportation. Suppose that beyond a threshold distance (usually time converted to distance), even a point in N_N should get its service using the more expensive Euclidean distance. For example, in many applications there is a time standard of service that must be maintained. This may mean that patients must be transported to a hospital within some established time threshold, or packages must be delivered before a fixed due date. Depending on the facility location $X = (x, y)$, a demand node i could be near X or far from X, making it difficult to a priori classify nodes as needing the fast, expensive (Euclidean) mode of transport. In this section, we describe how our model can be extended to handle the case where some demand nodes $i \in N_N$ get faster, more expensive service by using Euclidean distances rather than on-network distances whenever X is located beyond a given threshold distance from i.

For this model extension, the upper bounds are straightforward while the lower bounds become more complex. As before, the upper bound on a segment is easily established using any point along the segment (e.g., the minimum value of the objective function at the two endpoints). The upper bound is easy to calculate since the value of the objective function at a given location remains easy to calculate. We know the distances to all demand points, and we can simply check which demand points exceed the threshold distance. In contrast, we modify the lower bound as follows. For each segment of length s, we compute the distances $d_i(L)$ and $d_i(R)$ to the left (L) and right (R) ends of the segment. Since network distance is a concave function, the lowest possible distance is the minimum of these two distances. The maximum distance between demand point i and any point in the segment is [6, 60]:

$$d_{\max}(i) = \frac{1}{2}[d_i(L) + d_i(R) + s]. \tag{8}$$

When both the minimum and maximum distances are either smaller or larger than the threshold distance, the same mode of service (network or Euclidean) is used for the service of that demand point. When the threshold distance is between the minimum and the maximum distances, the cheaper mode of service (network distance) is assumed for that demand point yielding a lower bound. When the segment is small, it is expected that very few demand points, if any, will fall in this category and therefore the quality of the lower bound is mostly maintained.

Note that when a point in N_N is served using Euclidean distance, the weight w_i should be multiplied by c to reflect the higher cost of Euclidean distance service.

4.2 The Minimax Objective

In many emergency facility location problems, minimizing the maximum distance (minimax) objective is employed [41, 58, 40], which provides the best possible service to the farthest demand point. To model the minimax objective in our framework, we first separate the objective function into two components, which measure the maximum network and Euclidean distances from the facility X to any demand point i:

$$F_N(X) = \max_{i \in N_N} \left\{ d_i^N(X) \right\}; \qquad F_E(X) = \max_{i \in N_E} \left\{ d_i^E(X) \right\}. \tag{9}$$

Two weights, f_N and f_E, are given. The minimax objective can be formulated as minimizing

$$\max \left\{ f_N F_N(X), f_E F_E(X) \right\}. \tag{10}$$

It is not immediately clear how to calibrate the two parameters f_N and f_E. It is possible that at the solution point of one objective, $F_N(X)$ or $F_E(X)$, the value of the objective function for the other function is lower if the weights are not calibrated properly. The only "interesting" case is when such situations do not occur. We prefer to have constant f_N and f_E values rather than generating the efficient frontier of the two objectives, which does not result in a unique location solution.

We propose an objective similar to the concept of minimax regret which was proposed in location models such as [1, 27, 35]. Let X_N^* and X_E^* be the locations on the network that minimize $F_N(X)$ and $F_E(X)$ with optimal values F_N^* and F_E^*, respectively. When a common location to the two objectives is sought, we cannot minimize both objectives simultaneously (unless $X_N^* = X_E^*$, which is very unlikely). We propose a compromise objective where the maximum percentage increase in the values of the objective functions is minimized:

$$\min_X \left\{ \max \left\{ \frac{F_N(X)}{F_N^*}, \frac{F_E(X)}{F_E^*} \right\} \right\}. \tag{11}$$

This means selecting $f_N = \frac{1}{F_N^*}$ and $f_E = \frac{1}{F_E^*}$.

Note that by selecting these weights, the solution is a compromise solution between the two objectives unless $X_N^* = X_E^*$.

4.2.1 Solving the Minimax Problem

To solve the minimax problem (11), we carry out the following steps:

1. Solve a location problem using only network distances and demand points $i \in N_N$ to produce the minimax solution X_N^* with optimal value F_N^*.

2. Solve a location problem using only Euclidean distances and demand points $i \in N_E$ to produce the minimax solution X_E^* with optimal value F_E^*.

3. Compute $f_N = \frac{1}{F_N^*}$ and $f_E = \frac{1}{F_E^*}$.

4. Solve a location problem with both network and Euclidean distances for all demand points $i \in N_N \cup N_E$ using the objective function defined by (10). Return this solution.

Finding F_N^*, F_E^* and solving the location problem with minimax objective (10) can be done by the Big Segment Small Segment algorithm in a similar fashion to the approach used for solving the minisum problem. The value of the objective function at any point in the segment or the minimum value at the two ends of the segment are upper bounds. We define the lower bounds for $F_N(X)$, $F_E(X)$, and (10) below.

Lower bound for $F_N(X)$: Since network distances are concave, a lower bound for each distance is obtained by evaluating the distances to each of the endpoints of the segment and selecting the smaller one. The maximum of these lower bounds is a lower bound for the maximum distance to any point of the segment and thus is a lower bound to $F_N(X)$ in the segment.

Lower bound for $F_E(X)$: Consider a link of length ℓ with distances to demand point i from its two endpoints of $D_1 \leq D_2$ depicted in Figure 2. By the Cosine Theorem, $\cos\alpha = \frac{D_1^2+\ell^2-D_2^2}{2\ell D_1}$.

Also, $x = D_1 \cos\alpha$. It can be shown that d_i as defined below is the shortest distance from demand point $i \in N_E$ to all the points in the segment:

$$d_i = \sqrt{D_1^2 - x^2} \quad \text{where} \quad x = \max\left\{\frac{D_1^2 + \ell^2 - D_2^2}{2\ell}, 0\right\}. \tag{12}$$

The lower bound on the segment is therefore $\max\limits_i \{d_i\}$.

Lower bound for (10): For a given segment, we first compute the lower bounds LB_N and LB_E of the functions $F_N(X)$ and $F_E(X)$ respectively, as described above. Then, the value of $\max\{f_N LB_N, f_E LB_E\}$ is a lower bound for the minimax objective (10) on the segment.

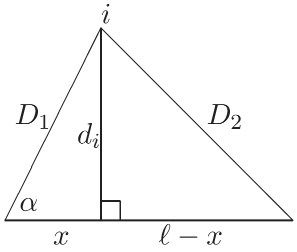


Figure 2: The shortest distance to a segment.

4.2.2 Alternative Approaches

As discussed in the computational experiments section, solving these problems by the Big Segment Small Segment algorithm is very efficient. Regardless, there may be more efficient ways to find F_N^* and F_E^*. For completeness, we discuss such approaches here.

The minimax problem on the network is well known and can be solved by the algorithm proposed by [58].

Solving the Euclidean minimax problem anywhere in the plane is equivalent to finding the smallest circle that encloses all demand points and is one of the oldest location problems [68, 69]. Chrystal [20] proposed a solution method for this problem later refined by [51]. As reported by [32], the most effective algorithm is the one proposed by [41] whose average complexity is about linear. Drezner and Shelah [38] constructed a contrived example with complexity of $O(n^2)$ where

n is the number of demand points. Megiddo [55] suggested an algorithm to solve this problem in linear time. To the best of our knowledge, no one has previously investigated the problem of finding the best location anywhere on the network that minimizes the maximum Euclidean distance to a set of points in the plane.

The objective function on a link is convex as it is a maximum of two convex functions. Therefore, a local minimum on a link is the global minimum on that link. The minimum value of the objective function on a link can be found by the golden section search (see, for example, [77]). The proposed algorithm consists of the following steps:

1. Find the lower bound on each link $\max_i \{d_i\}$ where d_i is found by (12).
2. Sort the links in increasing order of the lower bound.
3. Consider the links sequentially according to the order found in Step 2.
4. Find the minimum value of the objective function on the link by the golden section search.
5. Once the lower bound of the next link is not lower than the lowest value of the objective function found so far, stop. The best solution found is the optimal solution on the network.

Note that rather than sorting the vector of lower bounds, the more efficient binary heap [15, 45, 57] can be used.

4.3 Other Extensions

The base model, its threshold extension and the minimax objective can be extended to conditional problems. In this context, one or more facilities already exist in the area and we wish to add a new facility. Each customer should receive service from their closest facility, whether existing or new. See [5, 7, 17, 18, 59].

Let D_i be the shortest distance between demand point i and the existing facilities. D_i is a constant independent of X which can be easily pre-computed. We replace the distance $d_i(X)$ by $\min\{d_i(X), D_i\}$. For example, to model and solve the base model, we employ the limited distance idea suggested by [37]. Problem (1) is converted to minimizing

$$F_{cond}(X) = \sum_{i \in N} w_i \min\left\{d_i(X), D_i\right\}. \tag{13}$$

Note that for demand points using Euclidean distances w_i is multiplied by c.

The value of the objective at a given facility location is calculated by (13). Therefore, the upper bound is easily established as the value of the objective function at any point on the segment, for which we use the minimum value from the segment's two endpoints. In order to establish a lower bound, we first replace for every $i \in N_E$ the Euclidean distance with its tangent plane. The distance $d_i(X)$ and its tangent plane $T_i(X)$ are both concave and since D_i is a constant, the minimum of a concave function and a constant is concave. Consequently, the minimum value of the converted function at the endpoints of the segment is a lower bound.

5 Computational Experiments

Programs were compiled by an Intel 11.1 Fortran Compiler with no parallel processing using double precision arithmetic and run on a desktop with an Intel 870/i7 2.93GHz CPU Quad processor and 8GB RAM. Only one thread was used.

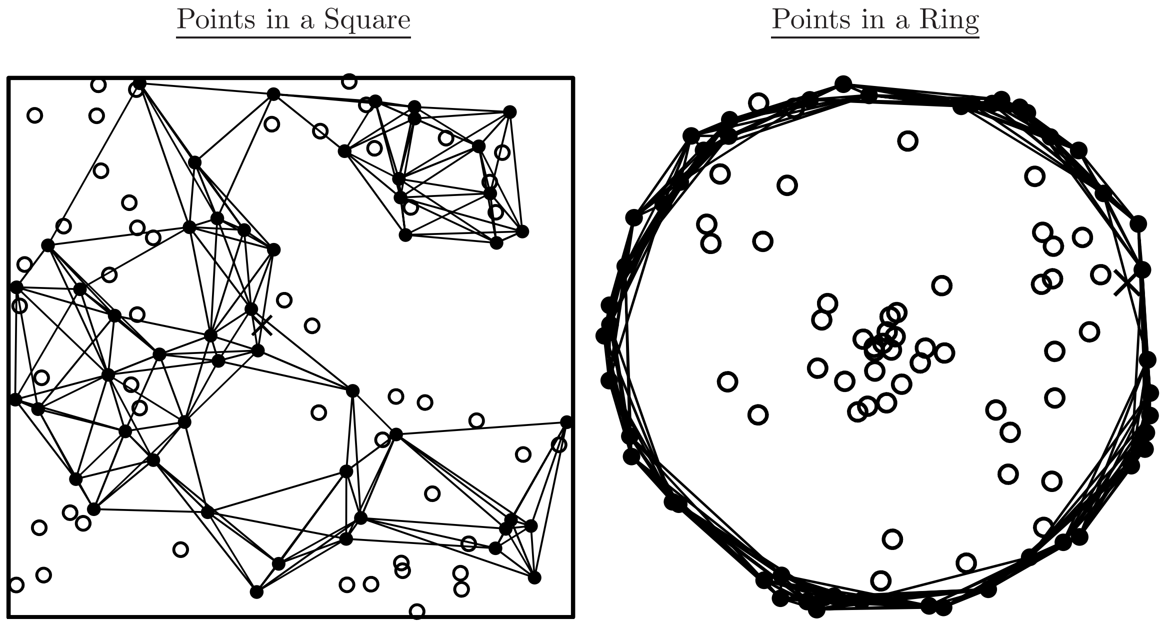
5.1 Experiments With the Base Problem With and Without a Threshold

For our first batch of test instances, we generated demand points (both N_N and N_E) uniformly at random from a unit square, linking some nodes in N_N as follows. Starting from a network with no links at all, we add six links for each demand point (node) $i \in N_N$ such that the links we add directly connect i with its six closest neighbor nodes (as measured by Euclidean distance). Each link we add is assigned a length equal to the Euclidean distance between the two nodes it connects. Many of the links are selected twice. This happens when two points are close to one another such that each node is among the six closest nodes to the other one. Such duplication is eliminated by deleting one of the links. In all cases we found the networks we generated were connected. When we generated only five links for each node, there was one case where the network was not connected.

We used a cost ratio (between Euclidean and network distances) of $c = 5$ and applied equal weight to all demand points.

In all experiments we generated an equal number of demand points in N_N and N_E for a total of n demand points. This is done to avoid excessively large tables that do not enhance the assessment of the procedures. We used a relative accuracy of $\epsilon = 10^{-10}$ in the Big Segment Small Segment algorithm, and solved each set 10 times. For threshold distance instances, a threshold distance of 0.5 was used.

Figure 3: Square and Ring Instances



• network nodes; ○ off-network points; × base problem optimal solution.

Since the solution to instances where nodes are generated uniformly in a square tends to be in the center of the square, we also generated a second batch of instances to verify the robustness of our algorithm. For these instances, off-network demand points were generated in a disc of radius 1, and on-network demand points were generated in a ring along the outside edge of the disc. The ring shares the same center as the disc, and has an inner radius of 0.9 and an outer radius of 1. By construction, the road network is in the ring and thus the optimal solution must lie in the ring.

This distribution of demand is realistic when there is an area where ambulances cannot reach like a lake, a mountain, or a forest surrounded by demand points and a road system. Distress calls can come from either the ring (where ambulances can be used) or the interior disc where service can only be provided using Euclidean distances. The solution to such a problem can be anywhere on the network and, unlike our first batch of instances, the value of the objective function is flat with many local optima. Finally, for these instances we constructed the network by connecting each node to its closest ten points rather than the closest six points as in our first batch of instances. This is because using only six closest points caused many instances to have disconnected networks. Consequently, the number of network links in our second batch of instances is larger than in our first batch of instances. Figure 3 provides an example from both batches of instances (generated uniformly in a unit square, and in/around a disc).

In Table 1 we report the results for our two data sets. For each data set, we first report for each n the characteristics of the instances: averages for the number of links, and the time in seconds required to generate the matrix of network distances using Floyd's shortest path algorithm [44]. We then report for the no-threshold and threshold cases the averages of: the number of iterations, max number of segments during Step 4 of the Big Segment Small Segment algorithm following the initial pruning of links, and run time in seconds per instance for the full application of the algorithm. The full application of the algorithm includes: (i) finding the value of the objective function at each network node to establish an upper bound, (ii) scanning all links and calculating a lower bound for each one to establish the initial set of segments (those links whose lower bound is smaller than the upper bound) for the branch-and-bound phase, and (iii) running the branch-and-bound phase until all segments are pruned. The variability of the reported values for the first data set is so small that there is no need to report other characteristics such as maximum run times. For example, the time for solving the largest problem in a square with threshold ranged between 1.29 and 1.36 seconds with an average of 1.32 seconds.

It is interesting that our second more "difficult" data set did not require much more computational effort. Actually, the increase in run time is mainly due to the increase in the number of links. The base problem was solved equally well but there were a few exceptions in the threshold

Table 1: Average performance of the algorithm

n	# Links	†	No Threshold			Having a Threshold		
			Itera- -tions	Max Seg.	Time (sec.)	Itera- -tions	Max Seg.	Time (sec.)
Nodes in a Square								
100	183	0.00	172	21	0.00	332	39	0.00
200	360	0.01	193	12	0.00	347	46	0.00
300	541	0.01	192	16	0.00	1,268	101	0.01
400	715	0.02	180	13	0.00	333	43	0.01
500	888	0.04	197	14	0.01	350	41	0.01
600	1,071	0.09	198	14	0.02	358	53	0.02
700	1,240	0.15	196	10	0.01	384	52	0.02
800	1,426	0.31	188	15	0.02	327	51	0.03
900	1,598	0.59	201	14	0.03	327	48	0.04
1,000	1,783	0.98	178	13	0.03	290	50	0.04
2,000	3,547	11.73	179	13	0.15	471	76	0.18
3,000	5,326	43.87	186	12	0.36	373	75	0.43
4,000	7,086	109.73	168	14	0.67	521	87	0.82
5,000	8,854	219.15	175	12	1.04	506	109	1.32
Nodes in a Ring								
100	278	0.00	318	65	0.00	5,540	1,045	0.05
200	565	0.00	234	35	0.00	4,098	736	0.05
300	842	0.01	197	26	0.01	14,286	2,589	0.25
400	1131	0.02	235	19	0.00	215	34	0.01
500	1409	0.03	234	18	0.01	119	33	0.02
600	1693	0.07	250	19	0.02	6,142	1,078	0.18
700	1983	0.13	243	15	0.03	95	25	0.04
800	2269	0.29	241	16	0.03	262	40	0.05
900	2561	0.58	207	15	0.04	170	29	0.06
1,000	2864	0.96	209	17	0.05	255	40	0.07
2,000	5896	11.71	197	12	0.24	109	34	0.28
3,000	8824	43.83	201	14	0.56	1,763	336	0.85
4,000	11,670	109.46	207	12	1.04	108	35	1.23
5,000	14,609	218.68	193	12	1.61	54	22	1.96

† Time in seconds for calculating network distances by computing shortest paths.

problems. When there are many near-optimal solutions, it is more common to encounter many iterations. The most difficult case was one of the $n = 300$ problems that required 113,058 iterations with a maximum of 20,624 segments and was solved in 2.07 seconds, which is still very fast.

The algorithm is very fast and efficient. It is interesting that the vast majority of the time was spent finding the shortest path distances between pairs of nodes (which has a complexity of $O(n^3)$) and only a very small portion of the run time was spent on the algorithms. The pre-processing to compute the all-pairs shortest paths distances only needs to be computed once, even when our algorithm would be run repeatedly by a multi-facility metaheuristic. Consequently, the run times that exclude pre-processing are more relevant when considering the use of our method within the context of such a multi-facility metaheuristic. The largest tested problem of $n = 5,000$ demand points was solved by the Big Segment Small Segment algorithm in only 1.04 seconds without a threshold and 1.32 seconds with it for the first data set that has 8,854 links, and in 1.61 seconds without a threshold and 1.96 seconds with it for the second data set that has 14,609 links. Indeed, the efficiency of our method makes it a good candidate to use as part of broader heuristics that solve multiple-facility location problems by solving a sequence of single-facility problems.

5.2 Experiments With Minimax Problems

All three components of the solution process (finding F_N^*, F_E^*, and solving (10)) were performed using the Big Segment Small Segment algorithm. Since run times are so short, there is no reason to test more specifically-tailored algorithms for finding F_N^* and F_E^* in an attempt to shorten the run time. Moreover, due to the similarity in the run times we previously observed for our first and second batches of instances, for this section we only report run times for our first batch of instances.

The algorithms are very efficient. The largest problem was solved in an average time of 1.12 seconds.

6 Discussion and Suggestions for Future Research

We analyzed and solved the problem of locating a facility anywhere on a network when demand is generated at two sets of points: one is located anywhere in the plane, and another is restricted to a network embedded in the plane. Service distance to demand points at nodes is the shortest path distance through links of the network, while service distance to demand points located anywhere in the plane is the more expensive per unit distance and is measured by the Euclidean norm. The

Table 2: Average performance of the minimax algorithms

	Finding F_N^*			Finding F_E^*			Solving (10)			Objec-	Total
n	Itera-	Max	Time	Itera-	Max	Time	Itera-	Max	Time	-tive	Time
	-tions	Seg.	(sec.)	-tions	Seg.	(sec.)	-tions	Seg.	(sec.)	Value	(sec.)
100	95	9	0.00	285	32	0.00	37	10	0.00	1.05650	0.00
200	79	8	0.00	194	22	0.00	18	8	0.00	1.03954	0.00
300	114	7	0.00	217	24	0.00	23	8	0.00	1.02365	0.00
400	92	7	0.00	263	28	0.00	26	9	0.01	1.03001	0.01
500	98	8	0.00	247	26	0.01	25	10	0.00	1.03146	0.01
600	117	8	0.00	223	25	0.00	24	11	0.01	1.01363	0.01
700	90	5	0.00	201	20	0.01	26	10	0.01	1.02346	0.02
800	97	9	0.00	214	23	0.01	18	7	0.01	1.01581	0.03
900	95	7	0.01	242	23	0.01	16	7	0.02	1.01184	0.03
1,000	108	8	0.01	225	23	0.02	22	10	0.02	1.01703	0.04
2,000	100	8	0.06	224	24	0.05	28	10	0.06	1.01155	0.17
3,000	89	6	0.12	215	23	0.11	25	10	0.15	1.00573	0.38
4,000	107	8	0.24	218	23	0.20	23	9	0.28	1.00670	0.71
5,000	84	8	0.37	180	21	0.30	28	11	0.45	1.00850	1.12

base model involves the minimization of the total weighted distance to all demand points. Two extensions to the base model were analyzed and solved: (i) the threshold distance model where if the network distance exceeds a given threshold, then service is always provided using Euclidean distance, and (ii) a minimax version of the base model. The problems are solved by the Big Segment Small Segment global optimization method [6]. Computational experiments demonstrate the effectiveness of the solution procedures. Problems with 5,000 demand points were solved in a little over one second of computer time, which illustrates the effectiveness of the approaches. It verifies the usefulness of the "Big Segment Small Segment" procedure for locating a facility anywhere on the network. No other paper investigated this particular problem so it is difficult to compare run times with existing results.

6.1 Suggestions for Future Research

Optimal algorithms for solving multiple-location versions of our problems are not expected to be efficient when there are a large number of facilities because these problems are NP-hard. The planar p-median Euclidean distance problem, which is a special case of our problem, is known to be NP-hard [56]. For small values of p, such as $p = 2$, the Big Segment Small Segment algorithm can be executed in a nested approach as suggested in [6].

For larger values of p, heuristic approaches can be implemented. Many heuristic approaches that were implemented for solving the planar p-median Euclidean distance problem (for recent reviews, consult [10, 11, 33, 34]) are based on optimally solving a sequence of single-facility problems and can be applied to the mixed planar and network problem. The alternating procedure by [24, 25] is a location-allocation procedure that proceeds as follows. A starting solution is randomly generated and demand points are assigned to their closest facility. Next, p single-facility problems are solved and a new allocation of demand points to facilities is produced. This process alternates between solving p independent single-facility problems and allocating demand to facilities until the solution stabilizes. This alternating procedure was improved by [33] replacing the single-facility solution procedure by the conditional single-facility solution procedure described in Section 4.3. Overall, multi-facility heuristics that employ single-facility solutions can be applied in a multi-start fashion, and are practical because of the very fast single-facility optimal procedure presented in this paper. Many metaheuristic procedures such as local search, tabu search and simulated annealing [62], variable neighborhood search [12], and genetic algorithms [33] have been applied to the planar p-median Euclidean distance problem, and make use of optimal single-facility solutions. We expect that such metaheuristics applied to the mixed planar and network problem would work well, and leave comprehensive testing of such approaches for future work.

Second, we also may consider an extension to the base problem presented in this paper that has a fixed budget. In that case, some on-network demands could be provided by the faster and more expensive Euclidean distance service as long as the budget is not exceeded.

Third, we may also be interested in relaxing the requirement that the optimal solution be on the network. This may be useful in cases where new roads could be built to connect an off-network location to an existing road network. We expect the Big Triangle Small Triangle technique [39] could be employed here, in conjunction with similar bounds to the ones we have derived in this paper.

Finally, it would be of interest to extend the location-routing problem [53, 71] to incorporate both on-network and off-network distances similar to the model suggested by [63].

References

[1] I. Averbakh and O. Berman, Minmax regret median location on a network under uncertainty, INFORMS J Computi 12 (2000), 104–110.

[2] R. Batta and U.S. Palekar, Mixed planar/network facility location problems, Comput & Oper Res 15 (1988), 61–67.

[3] J. Beasley and E. Nascimento, The vehicle routing-allocation problem: A unifying framework, Top 4 (1996), 65–86.

[4] J.E. Bell, S.E. Griffis, W.A. Cunningham III, and J.A. Eberlan, Location optimization of strategic alert sites for homeland defense, Omega 39 (2011), 151–158.

[5] O. Berman and Z. Drezner, A new formulation for the conditional p-median and p-center problems, Oper Res Lett 36 (2008), 481–483.

[6] O. Berman, Z. Drezner, and D. Krass, Big segment small segment global optimization algorithm on networks, Networks 58 (2011), 1–11.

[7] O. Berman and D. Simchi-Levi, The conditional location problem on networks, Transportation Sci 24 (1990), 77–78.

[8] R. Blanquero and E. Carrizosa, Solving the median problem with continuous demand on a network, Comput Optim Appl 56 (2013), 723–734.

[9] C.C. Branas and C.S. Revelle, An iterative switching heuristic to locate hospitals and helicopters, Socio-Economic Planning Sciences 35 (2001), 11–30.

[10] J. Brimberg, Z. Drezner, N. Mladenović, and S. Salhi, A new local search for continuous location problems, Eur J Oper Res 232 (2014), 256–265.

[11] J. Brimberg, P. Hansen, N. Mladenović, and S. Salhi, A survey of solution methods for the continuous location-allocation problem, Int J Oper Res 5 (2008), 1–12.

[12] J. Brimberg, P. Hansen, N. Mladenović, and E. Taillard, Improvements and comparison of heuristics for solving the uncapacitated multisource Weber problem, Oper Res 48 (2000), 444–460.

[13] J. Brimberg and N. Mladenović, Degeneracy in the multi-source Weber problem, Math Program 85 (1999), 213–220.

[14] L. Brotcorne, G. Laporte, and F. Semet, Ambulance location and relocation models, Eur J Oper Res 147 (2003), 451–463.

[15] S. Carlsson, Improving worst-case behavior of heaps, BIT Numer Mathemati 24 (1984), 14–18.

[16] E. Carrizosa and A.M. Rodríguez-Chía, Weber problems with alternative transportation systems, Eur J Oper Res 97 (1997), 87–93.

[17] R. Chen, Conditional minisum and minimax location-allocation problems in Euclidean space, Transportation Sci 22 (1988), 157–160.

[18] R. Chen and G.Y. Handler, The conditional p-center in the plane, Naval Res Logist 40 (1993), 117–127.

[19] S.H. Cho, H. Jang, T. Lee, and J. Turner, Simultaneous location of trauma centers and helicopters for emergency medical service planning, Oper Res 62 (2014), 751–771.

[20] G. Chrystal, On the problem to construct the minimum circle enclosing n given points in the plane, Proc Edinburgh Math Soc 3 (1885), 30–33.

[21] R.L. Church, COBRA: A new formulation of the classic p-median location problem, Ann Oper Res 122 (2003), 103–120.

[22] R.L. Church, BEAMR: An exact and approximate model for the p-median problem, Comput & Oper Res 35 (2008), 417–426.

[23] R.L. Church and C.S. ReVelle, The maximal covering location problem, Papers Regional Sci Association 32 (1974), 101–118.

[24] L. Cooper, Location-allocation problems, Oper Res 11 (1963), 331–343.

[25] L. Cooper, Heuristic methods for location-allocation problems, SIAM Revi 6 (1964), 37–53.

[26] R. D'Andrea, Can drones deliver?, IEEE Trans Automation Sci Eng, 11 (2014), 647–648.

[27] M. Daskin, S. Hesse, and C. Revelle, α-reliable p-minimax regret: A new model for strategic facility location modeling, Location Sci 5 (1997), 227–246.

[28] M.S. Daskin, Network and discrete location: Models, algorithms, and applications, John Wiley & Sons, New York, 1995.

[29] M.S. Daskin and E.H. Stern, A hierarchical objective set covering model for emergency medical service vehicle deployment, Transportation Sci 15 (1981), 137–152.

[30] J.M. Díaz-Báñez, M. Korman, P. Pérez-Lantero, and I. Ventura, "The 1-center and 1-highway problem," Computational geometry, Springer, 2012, pp. 155–165.

[31] J.M. Díaz-Báñez, M. Korman, P. Pérez-Lantero, and I. Ventura, "Locating a service facility and a rapid transit line," Computational geometry, Springer, 2012, pp. 126–137.

[32] Z. Drezner, "Continuous center problems," Foundations of location analysis, H.A. Eiselt and V. Marianov (Editors), Springer, 2011, pp. 63–78.

[33] Z. Drezner, J. Brimberg, S. Salhi, and N. Mladenović, New heuristic algorithms for solving the planar p-median problem, Comput Oper Res 62 (2015), 296–304.

[34] Z. Drezner, J. Brimberg, S. Salhi, and N. Mladenović, New local searches for solving the multi-source Weber problem, Ann Oper Res (2015), DOI: 10.1007/s10479-015-1797-5.

[35] Z. Drezner and J. Guyse, Application of decision analysis techniques to the Weber facility location problem, Eur J Oper Res 116 (1999), 69–79.

[36] Z. Drezner, K. Klamroth, A. Schöbel, and G.O. Wesolowsky, "The Weber problem," Facility location: Applications and theory, Z. Drezner and H.W. Hamacher (Editors), Springer, Berlin, 2002, pp. 1–36.

[37] Z. Drezner, A. Mehrez, and G.O. Wesolowsky, The facility location problem with limited distances, Transportation Sci 25 (1991), 183–187.

[38] Z. Drezner and S. Shelah, On the complexity of the Elzinga-Hearn algorithm for the 1-center problem, Math Oper Res 12 (1987), 255–261.

[39] Z. Drezner and A. Suzuki, The big triangle small triangle method for the solution of non-convex facility location problems, Oper Res 52 (2004), 128–135.

[40] Z. Drezner and G. Wesolowsky, Minimax and maximin facility location problems on a sphere, Naval Res Logist Q 30 (1983), 305–312.

[41] J. Elzinga and D. Hearn, Geometrical solutions for some minimax location problems, Transportation Sci 6 (1972), 379–394.

[42] E.T. Erdemir, R. Batta, P.A. Rogerson, A. Blatt, and M. Flanigan, Joint ground and air emergency medical services coverage models: A greedy heuristic solution approach, Eur J Oper Res 207 (2010), 736–749.

[43] I. Espejo and A.M. Rodríguez-Chía, Simultaneous location of a service facility and a rapid transit line, Comput & Oper Res 38 (2011), 525–538.

[44] R.W. Floyd, Algorithm 97: Shortest path, Commun ACM 5 (1962), 345.

[45] H.N. Gabow, Z. Galil, T. Spencer, and R.E. Tarjan, Efficient algorithms for finding minimum spanning trees in undirected and directed graphs, Combinatorica 6 (1986), 109–122.

[46] D.R.P. Groß, H.W. Hamacher, S. Horn, and A. Schöbel, Stop location design in public transportation networks: Covering and accessibility objectives, Top 17 (2009), 335–346.

[47] S.L. Hakimi, Optimum locations of switching centres and the absolute centres and medians of a graph, Oper Res 12 (1964), 450–459.

[48] S.L. Hakimi, Optimum distribution of switching centers in a communication network and some related graph theoretic problems, Oper Res 13 (1965), 462–475.

[49] P. Hansen, N. Mladenović, and É. Taillard, Heuristic solution of the multisource Weber problem as a p-median problem, Oper Res Lett 22 (1998), 55–62.

[50] K. Hogan and C. Revelle, Backup coverage concepts in the location of emergency service, Manage Sci 32 (1986), 1434–1444.

[51] S. Jacobsen, An algorithm for the minimax Weber problem, Eur J Oper Res 6 (1981), 144–148.

[52] D. Kim, C. Barnhart, K. Ware, and G. Reinhardt, Multimodal express package delivery: A service network design application, Transportation Sci 33 (1999), 391–407.

[53] G. Laporte, The vehicle routing problem: An overview of exact and approximate algorithms, Eur J Oper Res 59 (1992), 345–358.

[54] V. Marianov and C. ReVelle, "Siting emergency services," Facility location: A survey of applications and methods, Z. Drezner (Editor), Springer Series in Operations Research, 1995, pp. 199–223.

[55] N. Megiddo, Linear-time algorithms for linear programming in R^3 and related problems, SIAM J Computi 12 (1983), 759–776.

[56] N. Megiddo and K.J. Supowit, On the complexity of some common geometric location problems, SIAM J Computi 13 (1984), 182–196.

[57] K. Mehlhorn and P. Sanders, Algorithms and data structures: The basic toolbox, Springer, 2008.

[58] E. Minieka, The centers and medians of a graph, Oper Res 25 (1977), 641–650.

[59] E. Minieka, Conditional centers and medians on a graph, Networks 10 (1980), 265–272.

[60] E. Minieka, Anticenters and antimedians on a network, Networks 13 (1983), 359–364.

[61] N. Mladenović, J. Brimberg, P. Hansen, and J.A. Moreno-Perez, The p-median problem: A survey of metaheuristic approaches, Eur J Oper Res 179 (2007), 927–939.

[62] J. Moreno-Perez, J. Moreno-Vega, and N. Mladenovic, "Tabu search and simulated annealing in p-median problem," Proceedings of the Canadian Operational Research Society Conference, Montreal, 1994.

[63] C.C. Murray and A.G. Chu, The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery, Transportation Res C 54 (2015), 86–109.

[64] G. Nagy and S. Salhi, Location-routing: Issues, models and methods, Eur J Oper Res 177 (2007), 649–672.

[65] B. Pfeiffer and K. Klamroth, Bilinear programming formulations for Weber problems with continuous and network distances, J Oper Res Soc Jpn 48 (2005), 123–134.

[66] B. Pfeiffer and K. Klamroth, A unified model for Weber problems with continuous and network distances, Comput & Oper Res 35 (2008), 312–326.

[67] F. Silva and D. Serra, Locating emergency services with different priorities: The priority queuing covering location problem, J Oper Res Soc 59 (2008), 1229–1238.

[68] J. Sylvester, A question in the geometry of situation, Q J Math 1 (1857), 79.

[69] J. Sylvester, On Poncelet's approximate linear valuation of Surd forms, Philosophical Magazine 20 (Fourth series) (1860), 203–222.

[70] C. Toregas, R. Swain, C. ReVelle, and L. Bergman, The location of emergency service facilities, Oper Res 19 (1971), 1363–1373.

[71] P. Toth and D. Vigo, Vehicle routing: Problems, methods, and applications, second edition, SIAM, 2014.

[72] L. Vogt, C. Poojari, and J. Beasley, A tabu search algorithm for the single vehicle routing allocation problem, J Oper Res Soc 58 (2007), 467–480.

[73] X. Wang, S. Poikonen, and B. Golden, The vehicle routing problem with drones: Several worst-case results, Optim Lett (2016), 1–19.

[74] M. Wasner and G. Zäpfel, An integrated multi-depot hub-location vehicle routing model for network planning of parcel service, Int J Production Economics 90 (2004), 403–419.

[75] A. Weber, Über den standort der industrien, 1. teil: Reine theorie des standortes. English translation: On the location of industries, University of Chicago Press, Chicago, IL, 1929, Originally published in Tübingen, Germany in 1909.

[76] G.O. Wesolowsky, The Weber problem: History and perspectives, Location Sci 1 (1993), 5–23.

[77] W.I. Zangwill, Nonlinear programming: A unified approach, Prentice-Hall, Englewood Cliffs, NJ, 1969.