J Glob Optim (2007) 37:305–319
DOI 10.1007/s10898-006-9051-y

ORIGINAL PAPER

A general global optimization approach for solving location problems in the plane

Zvi Drezner

Received: 19 November 2005 / Accepted: 1 June 2006 /
Published online: 6 July 2006

Abstract We propose a general approach for constructing bounds required for the "Big Triangle Small Triangle" (BTST) method for the solution of planar location problems. Optimization problems, which constitute a sum of individual functions, each a function of the Euclidean distance to a demand point, are analyzed and solved. These bounds are based on expressing each of the individual functions in the sum as a difference between two convex functions *of the distance*, which is not the same as convex functions of the location. Computational experiments with nine different location problems demonstrated the effectiveness of the proposed procedure.

Keywords Planar location · Global optimization · Big triangle Small triangle · Single facility

1 Introduction

Hansen et al. (1981) suggested the "Big Square Small Square" (BSSS) technique for finding the global optimum for two dimensional location problems. The method is a branch and bound approach. We describe it for a maximization objective. A square containing the optimal solution point is constructed, and the process splits each square into four smaller squares. A lower bound (LB) and an upper bound (UB) for the value of the objective function for all the points in the square are calculated for each square. Squares whose UB does not exceed the best found solution are discarded from the search until the sizes of the remaining squares are all less than a tolerance ϵ. This approach was improved to the "Generalized Big Square Small Square" (GBSSS) algorithm by Plastria (1992). Drezner and Suzuki (2004) proposed to replace the

Z. Drezner(✉)
College of Business and Economics, California State University-Fullerton, Fullerton, CA 92834, USA
e-mail: zdrezner@fullerton.edu

squares with triangles hence the name “Big Triangle Small Triangle” (BTST). This global optimization approach is shown to be more efficient than BSSS and GBSSS.

Two problems were solved in Drezner and Suzuki (2004): an obnoxious facility location problem (Hansen et al. 1981) and the Weber location problem with some negative weights (Tellier and Polanski 1989; Drezner and Wesolowsky 1991; Maranas and Floudas 1994; Tuy et al. 1995; Krarup 1998). This method was applied successfully to the solution of numerous location problems. Berman et al. (2003) considered the weighted minimax (1-center) location problem in the plane when the weights are not given but rather drawn from independent uniform distributions. In Drezner and Drezner (2004) the Huff competitive location problem was optimally solved using BTST. In Drezner and Drezner (2006a, accepted for publication) two equity objectives are solved: minimizing the variance of the distances to the facility and minimizing the range of the distances to the facility. In Drezner and Drezner (2006b) the Huff competitive location problem is modified to account for lost demand. The acceleration–deceleration distance is defined in Drezner et al. (2006) and the single facility location problem based on this distance is optimally solved using BTST. This problem is especially difficult because every demand point is a local minimum and, for example, a problem with 10,000 demand points has at least 10,000 local optima. In Drezner and Scott (2006) a queueing-location model is analyzed when some of the demand is lost. In Drezner et al. (2003) an inventory-location model in the plane is investigated. The gradual cover problem is defined and solved in Drezner et al. (2004). In standard covering problem there is a well defined covering radius. In the gradual covering problem there are minimum and maximum cover radii. A point is fully covered within the minimum radius and it is not covered at all beyond the maximum radius. Between these two radii coverage declines linearly.

The successful application of BTST requires good bounds on the value of the objective function in a triangle. The purpose of this paper is to propose a general method for deriving good bounds for many common location problems. The objective function is either a minimization or maximization of a sum of functions, each function associated with a demand point. These functions are functions of the Euclidean distance to a demand point. This general approach was tested on many of the problems mentioned above and computational results are reported.

The paper is organized as follows. In Sect. 2, we summarize the BTST algorithm. In Sect. 3, we construct the general bounds and prove their properties. In Sect. 4, we investigate the implementation of these general bounds to eleven location problems. In Sect. 5, issues of implementation are discussed and in Sect. 6 computational experiments with nine different location problems are reported. We conclude the paper in Sect. 7.

2 The big triangle small triangle method

The framework of the BTST approach is summarized below. The complete details are given in Drezner and Suzuki (2004). A feasible region, which consists of a finite number of convex polygons is given. The algorithm is detailed as follows:

Phase 1: Each convex polygon is triangulated using the Delaunay triangulation (Drezner and Suzuki 2004). The vertices of the triangles are the demand points and the vertices of the convex polygon. The union of the triangulations is the initial set of triangles.

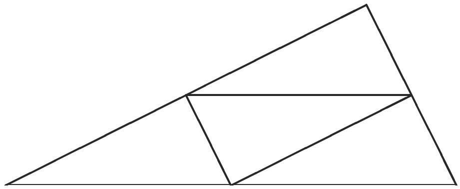

Fig. 1 The split of a triangle into four small triangles

Phase 2: Calculate a LB, and an UB, for each triangle. Let the largest LB be $\overline{\text{LB}}$. Discard all triangles, for which UB $\leq \overline{\text{LB}}(1+\epsilon)$.

Phase 3: Choose the triangle with the largest LB and split it into four small triangles by connecting the centers of its sides as depicted in Fig. 1. Calculate LB and UB for each triangle, and update $\overline{\text{LB}}$ if necessary. The large triangle and all triangles, for which UB $\leq \overline{\text{LB}}(1+\epsilon)$ are discarded.

Stopping Criterion: The branch and bound is terminated when there are no triangles left. The solution $\overline{\text{LB}}$ is within a relative accuracy of ϵ from the optimum.

Since the triangulation is based on the demand points as vertices, no demand point is in the interior of a triangle. This property is maintained throughout the algorithm because the interiors of small triangles constructed by splitting a large triangle are part of the interior of the large triangle. This property is helpful in constructing the bounds.

3 General upper and lower bounds

3.1 Notation and preliminary discussion

T	A given triangle
T_k	Vertex k of triangle T, for $k = 1,2,3$
n	The number of demand points
X	The unknown location of the new facility
$d_i(X)$	The Euclidean distance between demand point i and X
$d(X, Y)$	The Euclidean distance between points X and Y
$\phi(d)$	A known function of the distance
$\phi_i(d)$	The function associated with demand point i
$F(X)$	The objective function $= \sum_{i=1}^{n} \phi_i[d_i(X)]$.

We construct UB and LB in a triangle T based on three vertices T_1, T_2, T_3. The objective function in the triangle is $F(X) = \sum_{i=1}^{n} \phi_i[d_i(X)]$ for $X \in T$. For clarity of presentation we treat the case of maximizing $F(X)$. If the problem is a minimization problem, we can maximize $-F(X)$.

Most expressions involve a particular demand point. For simplicity of notation, the index i (indicating demand point i) is omitted from many of these expressions.

The UB and LB are applicable to functions $\phi(d)$, which can be expressed as a difference between two convex functions in d. This condition is not the customary condition in DC-optimization (Tuy et al. 1995; Tuy 1998) when the assumption is that $\phi(d(X))$, is a difference between convex functions in the location X. The distance function $d(X)$ is a convex function in X. However, a convex function of a convex function is not necessarily convex. $\phi(d(X))$ is convex in X when $\phi(d)$ is a *monotonically increasing* convex function of d. So, for example, $\phi(d) = \mathrm{e}^{-d}$ is convex but monotonically decreasing in d and $e^{-d(X)}$ is not convex in X. It is usually not difficult to express a function of one variable as a difference between two convex functions. For example, if the second derivative of $\phi(d)$ exists in the triangle and is bounded such that $\partial^2\phi/\partial d^2 \geq -M$, then $\phi(d) = [\phi(d) + Md^2/2] - Md^2/2$ is a difference between two convex functions. It is recommended however, if possible, to express $\phi(d)$ as a difference between two convex functions, which have no artificially large values.

Consider a point Y, which is not in the interior of the triangle. As data, we use the three distances between Y and the vertices d_1, d_2, d_3, and the sides of the triangle d_{12}, d_{13}, d_{23} (d_{ij} is the length of the side connecting T_i and T_j).

Let the minimum and maximum distance from a point Y to all points in the triangle be $d_{\min}(Y)$ and $d_{\max}(Y)$, respectively. This means that $d_{\min}(Y) \leq d(X, Y) \leq d_{\max}(Y)$ for any point X in the triangle.

Since, the distance function is convex, $d_{\max}(Y) = \max\limits_{i=1,2,3} \{d_i\}$. The calculation of the minimum distance is described in Drezner and Drezner (2004). We suggest a different way to calculate $d_{\min}(Y)$.

3.2 Finding the shortest distance to all the points of a triangle

Since, point Y is not in the interior of the triangle, the shortest distance must be to a point X on the boundary of the triangle. It is possible that the shortest distance is to one of the three vertices. We need to find whether there is a point $X \in T$, which is closer than the closest vertex. We first find the shortest distance from a point Y to all points on the side connecting T_i and T_j. Point Y, T_i, and T_j form a triangle whose sides are d_i, d_j and d_{ij} (see Fig. 2). If one of the angles opposite d_i or d_j is greater than 90°, then the minimum distance from point Y to this side is obtained to one of the two vertices. We are therefore interested in the shortest distance to the side only when both angles do not exceed 90°. This condition is equivalent to:

$$d_i^2 \leq d_j^2 + d_{ij}^2 \quad \text{and} \quad d_j^2 \leq d_i^2 + d_{ij}^2.$$

These conditions can be summarized as

$$\left| d_i^2 - d_j^2 \right| \leq d_{ij}^2. \tag{1}$$

Define $\theta_{ij} = \left(d_i^2 - d_j^2\right)^2 / d_{ij}^4$, then condition (1) is equivalent to $\theta_{ij} \leq 1$.

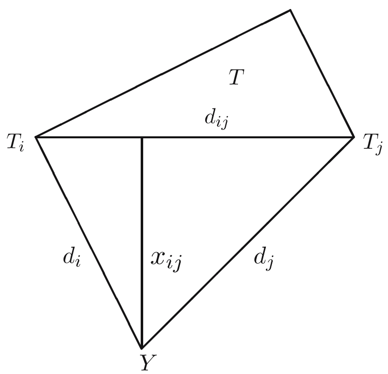

Fig. 2 The shortest distance to a triangle

Let x_{ij} be the shortest distance to the side T_iT_j of the triangle (assuming that $\theta_{ij} \leq 1$) (see Fig. 2). x_{ij} is the height of the triangle perpendicular to the segment T_iT_j. Then,

$$\sqrt{d_i^2 - x_{ij}^2} + \sqrt{d_j^2 - x_{ij}^2} = d_{ij}$$

or

$$\sqrt{d_i^2 - x_{ij}^2} = d_{ij} - \sqrt{d_j^2 - x_{ij}^2}.$$

Simple algebraic manipulations lead to:

$$x_{ij}^2 = d_j^2 - \frac{\left(d_i^2 - d_j^2 - d_{ij}^2\right)^2}{4d_{ij}^2} = \frac{d_i^2 + d_j^2}{2} - \frac{d_{ij}^2}{4} - \frac{\left(d_i^2 - d_j^2\right)^2}{4d_{ij}^2},$$
$$= \frac{d_i^2 + d_j^2}{2} - \frac{(1+\theta_{ij})d_{ij}^2}{4} \tag{2}$$

only for $\theta_{ij} \leq 1$. If $\theta_{ij} > 1$ define $x_{ij} = \infty$. Note that when $\theta_{ij} = 1$, $x_{ij} = \min\left\{d_i, d_j\right\}$. These calculations yield

$$d_{\min}^2(Y) = \min\left\{\min_{1\leq i\leq 3}\left\{d_i^2\right\}, \min_{1\leq i<j\leq 3}\left\{x_{ij}^2\right\}\right\}. \tag{3}$$

Note, that calculating $d_{\min}^2(Y)$ and $d_{\max}^2(Y)$ does not require a square root operation when all the original data distances are squared.

3.3 Bounds on convex functions

In this section, we assume that $\phi(d)$ is convex and, we are interested in bounds for $d_{\min} \leq d \leq d_{\max}$.

Lemma 1

$$\phi(d) \leq \phi(d_{\min}) + \frac{\phi(d_{\max}) - \phi(d_{\min})}{d_{\max} - d_{\min}}\left[d - d_{\min}\right] = K_1 d + L_1, \tag{4}$$

where

$$K_1 = \frac{\phi(d_{\max}) - \phi(d_{\min})}{d_{\max} - d_{\min}}, \quad L_1 = \phi(d_{\min}) - K_1 d_{\min}.$$

Proof The Lemma follows the property that the function $\phi(d)$ is below the line connecting the endpoints of the segment $[d_{\min}, d_{\max}]$. □

Let $d_c = \frac{d_{\min}+d_{\max}}{2}$ be the center of the segment.

Lemma 2

$$\phi(d) \geq \phi(d_c) + \frac{\partial\phi}{\partial d}(d_c)\left[d - d_c\right] = K_2 d + L_2, \tag{5}$$

where

$$K_2 = \frac{\partial\phi}{\partial d}(d_c), \quad L_2 = \phi(d_c) - K_2 d_c.$$

Proof The function $\phi(d)$ is above the tangent line at d_c. □

3.4 Properties of the bounds

Let $\epsilon = d_{\max} - d_{\min}$. We assume that the second derivative of $\phi(d)$ by d exists and is bounded by M. The second derivative must be positive because $\phi(d)$ is convex. The error in the inequality of Lemma 1 is

$$e_1(d) = \phi(d_{\min}) + \frac{\phi(d_{\max}) - \phi(d_{\min})}{d_{\max} - d_{\min}} \left[d - d_{\min}\right] - \phi(d). \tag{6}$$

Lemma 3 $0 \le e_1(d) \le M\frac{\epsilon^2}{2}$

Proof By Lemma 1 $e_1(d) \ge 0$. By the Taylor expansion:
$\phi(d_{\max}) = \phi(d_{\min}) + \frac{\partial\phi}{\partial d}(d_{\min})\left[d_{\max} - d_{\min}\right] + \frac{\partial^2\phi}{\partial d^2}(\xi_1)\frac{(d_{\max}-d_{\min})^2}{2}$
for $d_{\min} \le \xi_1 \le d_{\max}$.
$\phi(d) = \phi(d_{\min}) + \partial\phi/\partial d(d_{\min}) \times \left[d - d_{\min}\right] + \partial^2\phi/\partial d^2(\xi_2) \times (d - d_{\min})^2/2$
for $d_{\min} \le \xi_2 \le d$.
Substituting into (6):

$$e_1(d) = \frac{\partial^2\phi}{\partial d^2}(\xi_1)\frac{(d - d_{\min})(d_{\max} - d_{\min})}{2} - \frac{\partial^2\phi}{\partial d^2}(\xi_2)\frac{(d - d_{\min})^2}{2} \le M\frac{\epsilon^2}{2}.$$

□

The error in the inequality of Lemma 2 is

$$e_2(d) = \phi(d) - \phi(d_c) - \frac{\partial\phi}{\partial d}(d_c)\left[d - d_c\right]. \tag{7}$$

Lemma 4 $0 \le e_2(d) \le M\frac{\epsilon^2}{8}$

Proof By Lemma 2 $e_2(d) \ge 0$. By the Taylor expansion:

$$\phi(d) = \phi(d_c) + \frac{\partial\phi}{\partial d}(d_c)\left[d - d_c\right] + \frac{\partial^2\phi}{\partial d^2}(\xi)\frac{(d - d_c)^2}{2} \quad \text{for } d_{\min} \le \xi \le d_{\max}.$$

Substituting into (7):

$$e_2(d) = \frac{\partial^2\phi}{\partial d^2}(\xi)\frac{(d - d_c)^2}{2} \le M\frac{\epsilon^2}{8}$$

because $|d - d_c| \le \frac{\epsilon}{2}$. □

3.5 The proposed upper bound

Suppose, that the function $\phi(d)$ can be expressed as a difference between two convex functions $\phi(d) = \phi_1(d) - \phi_2(d)$. In this section, we assume that the problem is a maximization problem. If the problem is a minimization problem, we simply reverse the role of $\phi_1(d)$ and $\phi_2(d)$.

A LB for the maximum possible value of the objective function in a triangle is the value of the objective function at any point in the triangle. We propose to use the value of the objective function at the center of gravity of the triangle: $\phi(T_0)$ for $T_0 = (T_1 + T_2 + T_3)/3$. We construct a general UB for all the points $X \in T$.

By applying Lemma 1 to $\phi_1(d)$, we get

$$\phi_1(d) \le K_1 d + L_1.$$

By applying Lemma 2 to $\phi_2(d)$, we get

$$\phi_2(d) \geq K_2 d + L_2.$$

Therefore,

$$\phi(d) = \phi_1(d) - \phi_2(d) \leq [K_1 - K_2]\, d + L_1 - L_2. \tag{8}$$

By applying Eq. 8 to demand point i not in the interior of the triangle, we define

$$\phi_i(d_i(X)) \leq K^{(i)} d_i(X) + L^{(i)} \tag{9}$$

yielding

$$F(X) = \sum_{i=1}^{n} \phi_i(d_i(X)) \leq \mathrm{UB}(X) = \sum_{i=1}^{n} \left[K^{(i)} d_i(X) + L^{(i)}\right] \quad \text{for } X \in T. \tag{10}$$

The maximum of $\mathrm{UB}(X)$ in the triangle is an UB for $F(X)$ in the triangle. We separate the terms in $\mathrm{UB}(X)$ to those with $K^{(i)} \geq 0$ and those with $K^{(i)} < 0$. Define $\mathrm{UB}_1(X) = \sum_{K^{(i)} \geq 0} K^{(i)} d_i(X)$; $\mathrm{UB}_2(X) = \sum_{K^{(i)} < 0} K^{(i)} d_i(X)$; $C = \sum_{i=1}^{n} L^{(i)}$, then $\mathrm{UB}(X) = \mathrm{UB}_1(X) + \mathrm{UB}_2(X) + C$.

The function $\mathrm{UB}_1(X)$ is convex, and the function $\mathrm{UB}_2(X)$ is concave. We further bound $\mathrm{UB}_2(X)$ as follows. Since, $\mathrm{UB}_2(X)$ is concave, it is below the tangent plane at $T_0 = (T_1 + T_2 + T_3)/3 = (x_0, y_0)$. This yields for $X = (x, y)$:

$$\begin{aligned}\mathrm{UB}_2(X) &\leq \mathrm{UB}_3(X) \\ &= \sum_{K^{(i)} < 0} K^{(i)} \left\{ d_i(T_0) + \frac{(x - x_0)(x_0 - x_i) + (y - y_0)(y_0 - y_i)}{d_i(T_0)} \right\}.\end{aligned} \tag{11}$$

The $\mathrm{UB}_3(X)$ is linear and therefore convex. Therefore, $\mathrm{UB}_1(X) + \mathrm{UB}_3(X) + C$ is convex and obtains its maximum value in the triangle at one of the three vertices. The upper bound in the triangle UB is:

$$\mathrm{UB} = \max_{k=1,2,3} \{\mathrm{UB}_1(T_k) + \mathrm{UB}_3(T_k)\} + C. \tag{12}$$

Let ϵ be the largest distance between two points in triangle T. It is clear by the triangle inequality that $d_{\max} - d_{\min} \leq \epsilon$ for any point Y. We prove that $\mathrm{UB} \leq \max_{X \in T} \{F(X)\} + O(\epsilon^2)$.

Theorem 1 $UB - \max_{X \in T} \{F(X)\} \leq O(\epsilon^2)$

Proof By Lemmas 3 and 4 and definition (10) $\mathrm{UB}(X) - \max_{X \in T} \{F(X)\} \leq O(\epsilon^2)$. To complete the proof, we need to show that $\mathrm{UB}_3(X) - \mathrm{UB}_2(X) \leq O(\epsilon^2)$. Consider, the error associated with demand point i

$$e_3(X) = d_i(T_0) + \frac{(x - x_0)(x_0 - x_i) + (y - y_0)(y_0 - y_i)}{d_i(T_0)} - d_i(X).$$

By the Taylor expansion, there exist a point (ξ, η) in the triangle such that

$$\begin{aligned}d_i(X) &= d_i(T_0) + \frac{(x - x_0)(x_0 - x_i) + (y - y_0)(y_0 - y_i)}{d_i(T_0)} \\ &\quad + \frac{(\eta - y_i)^2 (x - x_0)^2 + (\xi - x_i)^2 (y - y_0)^2 - 2(\xi - x_i)(\eta - y_i)(x - x_0)(y - y_0)}{2 d_i^3(T_0)}.\end{aligned}$$

The last term is equal to:

$$\frac{[(\eta - y_i)(x - x_0) - (\xi - x_i)(y - y_0)]^2}{2d_i^3(T_0)}.$$

Therefore,

$$e_3(X) = \frac{[(\eta - y_i)(x - x_0) - (\xi - x_i)(y - y_0)]^2}{2d_i^3(T_0)} = O(\epsilon^2). \qquad \square$$

Theorem 1 shows that if a triangle is split into four triangles, the error in the UB for each of the four smaller triangles is reduced by about four fold. Five splits of triangles reduces the error by about 1,000 times.

4 Examples

In this section, we compare the general approach with bounds suggested in other papers.

4.1 Obnoxious facility location

The problem is minimizing $\sum_{i=1}^{n} \frac{w_i}{d_i^2(X)}$ for positive weights ($w_i > 0$) (Drezner and Suzuki 2004). Since, it is a minimization problem, $\phi_1(d) = 0$; $\phi_2(d) = w_i/d^2$. By (5) $K_2 d + L_2 = w_i(3d_c - 2d)/d_c^3$. Since, $K_2 < 0$, the approximation (11) will not be applied and the LB is based on $\text{UB}_1(X) + C$. In Drezner and Suzuki (2004) a similar approach was taken by using $\phi(x) = 1/x$ and substituting $x = d_i^2(X)$ in the resulting tangent line leading to: $w_i(2d_c^2 - d^2)/d_c^4$ rather than $w_i 3d_c - 2d/d_c^3$.

Note that $(3d_c - 2d)/d_c^3 = (3d_c^2 - 2d_c d + d^2 - d^2)/d_c^4 = (2d_c^2 - d^2 + (d_c - d)^2)/d_c^4 \geq (2d_c^2 - d^2)/d_c^4$. Therefore, the bound proposed here by the general approach is actually better than the specific bound used in Drezner and Suzuki (2004). We also mention that in Drezner and Suzuki (2004) d_c was selected as the distance to the center of gravity rather than $(d_{\max} + d_{\min})/2$ thus avoiding the need to calculate $d_{\min}$ and $d_{\max}$.

4.2 Weber problem with some negative weights

The problem is minimizing $\sum_{i=1}^{n} w_i d_i(X)$ when some of the weights may be negative (Drezner and Suzuki 2004). Since, $\phi(d)$ is linear, the problem is unaltered by (4) and (5). The UB (12) is the same as suggested in Drezner and Suzuki (2004).

4.3 Huff competitive location

The problem is to maximize $\sum_{i=1}^{n} \frac{b_i}{1+h_i d^{\lambda}}$ (Drezner and Drezner 2004). For this maximization problem $\phi(d) = b/(1 + hd^{\lambda})$ where h and b are positive constants and $\lambda \geq 1$. $\partial^2\phi/\partial d^2 \geq -b\lambda(\lambda - 1)hd^{\lambda-2}$. Therefore, $\phi(d) + bhd^{\lambda}$ is convex and, we can use $\phi_1(d) = \phi(d) + bhd^{\lambda}$; $\phi_2(d) = bhd^{\lambda}$. The LB suggested in Drezner and Drezner (2004) is much more contrived.

4.4 Stochastic weighted minimax

The minimization objective function can be written using $\phi(d) = \ln d - \ln(T - ad)$ (Berman et al. 2003) which is a difference between two concave functions. One can use $\phi_1(d) = -\ln(T - ad)$; $\phi_2(d) = -\ln d$. A more complicated bound is suggested in Berman et al. (2003).

4.5 Minimizing variance

The objective is to minimize:

$$\frac{\sum\limits_{i=1}^{n} w_i d_i^2(X)}{\sum\limits_{i=1}^{n} w_i} - \left(\frac{\sum\limits_{i=1}^{n} w_i d_i(X)}{\sum\limits_{i=1}^{n} w_i} \right)^2 .$$

Drezner and Drezner 2006a, accepted, which is a difference between two convex functions of the distance $d_i(X)$. A modification is convenient in this case: express $\phi(d) = \phi_1(d) - \left[\phi_2(d)\right]^2$. Also, the simplifications suggested in Drezner and Drezner (2006a, accepted for publication) can simplify the solution of this particular problem. However, the structure of this problem is not suitable for the direct application of the general approach suggested in the present paper. Special bounds were constructed in Drezner and Drezner (2006a, accepted for publication) successfully solving this problem using BTST.

4.6 Minimizing range

The objective is to minimize: $\max_{1 \le i \le n} \{d_i(X)\} - \min_{1 \le i \le n} \{d_i(X)\}$ (Drezner and Drezner 2006a, accepted for publication), which is a difference between convex functions of the distances. Since, the objective function is not a sum, the procedure needs to be adjusted for this particular problem as is done in Drezner and Drezner (2006a, accepted for publication).

4.7 Unserviced demand

The problem is transformed to maximizing the sum based on $\phi(d) = e^{-d}$ or $\phi(d) = 1/(1 + d)$ which are both convex functions in d (Drezner and Scott 2006). In this case apply $\phi_1(d) = \phi(d)$ and $\phi_2(d) = 0$. This leads to the same bound proposed in Drezner and Scott (2006).

4.8 Inventory-location problem

The model leads to minimizing

$$\sum_{i=1}^{n} \left(\alpha_i d_i(X) + w_i \sqrt{A_i d_i^2(X) + B_i d_i(X) + C_i} \right)$$

(Drezner et al. 2003). The function $\phi(d) = \alpha d + w\sqrt{Ad^2 + Bd + C}$ is concave in d for the specific parameters of the problem (Drezner et al. 2003). We propose to apply

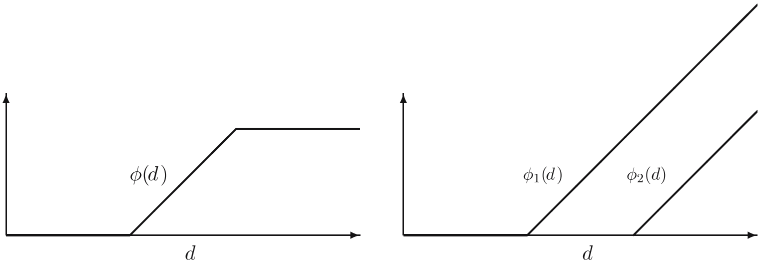

Fig. 3 The gradual covering functions

$\phi_1(d) = -\phi(d)$ and $\phi_2(d) = 0$. This model was solved heuristically in Drezner et al. (2003) and no global optimization technique was used.

4.9 Gradual covering

For this minimization problem: $\phi(d) = \begin{cases} 0 & d \le l \\ w(d-l) & l \le d \le u \\ w(u-l) & d \ge u \end{cases}$. (Drezner et al. 2003)

The function $\phi(d)$ is neither convex nor concave and is depicted in the left graph in Figure 3. It can easily be expressed as a difference between two convex functions as depicted in the right graph of the figure.

$$\phi_1(d) = \begin{cases} 0, & d \le l, \\ w(d-l), & d \ge l, \end{cases}$$

$$\phi_2(d) = \begin{cases} 0, & d \le u, \\ w(d-u), & d \ge u. \end{cases}$$

A specific lower bound which is different from the one proposed here was constructed in Drezner et al. (2003).

4.10 Lost demand

4.10.1 Model 1

Maximize $\sum_{i=1}^{n} b_i e^{-\lambda d_i(X)}$ (Drezner and Drezner 2006b). The function $\phi(d) = \mathrm{e}^{-\lambda d}$ is convex. This model is similar to the unserviced demand model in Sect. 4.7.

4.10.2 Model 2

Maximize $\sum_{i=1}^{n}(\alpha_i + \beta_i \mathrm{e}^{-\lambda d_i(X)})\gamma_i + \mathrm{e}^{-\lambda d_i(X)}/\delta_i + \mathrm{e}^{-\lambda d_i(X)}$ (Drezner and Drezner 2006b). All coefficients are non-negative and $\delta_i \ge \gamma_i$. As pointed out in Drezner and Drezner (2006b) the function $\phi(d) = (\alpha + \beta e^{-\lambda d})(\gamma + \mathrm{e}^{-\lambda d}/\delta + \mathrm{e}^{-\lambda d})$ is not necessarily convex. Following algebraic manipulations

$$\phi(d) = \alpha + (\gamma - \delta)\beta + \left\{\frac{(\delta-\gamma)\alpha + \beta\delta\gamma}{\delta^2}\right\}\mathrm{e}^{-\lambda d} + (\delta-\gamma)(\beta\delta-\alpha)\left\{\frac{1}{\mathrm{e}^{-\lambda d}+\delta} + \frac{\mathrm{e}^{-\lambda d}}{\delta^2}\right\}.$$

The function inside the braces of the last term is convex. Therefore, depending on the sign of $\beta\delta - \alpha$, the function $\phi(d)$ is expressed as either a sum of two convex functions and thus convex or a difference between two convex functions. The LB suggested in Drezner and Drezner (2006b) are the same ones suggested here.

4.11 The acceleration-deceleration distance

For this minimization problem $\phi(d) = \begin{cases} 2\sqrt{dd_0} \mid d \le d_0 \\ d + d_0 \mid d \ge d_0 \end{cases}$ for a given parameter $d_0 > 0$ (Drezner et al., 2006). As is proven in Drezner et al. (2006), $\phi(d)$ is concave. Therefore, apply $\phi_1(d) = -\phi(d)$ and $\phi_2(d) = 0$.

5 Implementation

In order to implement the general approach proposed in this paper, one needs to code four functions: $\phi_1(d)$, $\phi_2(d)$, $\partial\phi_1/\partial d(d)$, and $\partial\phi_2/\partial d(d)$. All these functions may be different for different demand points. We must make sure that $\phi_1(d)$ and $\phi_2(d)$ are convex. All the bounds can be calculated based on these four functions, and other values such as $d_{\min}$ and $d_{\max}$ do not depend on the specific functions used. One can code two different programs, one for minimization and one for maximization. It is easier to code just one approach (e.g. maximization) and apply $-\phi(d)$ for minimization problems.

We coded such a program[1] and for each particular objective function (which is a sum of individual functions each based on a distance to one demand point) we need to code only these four functions. Note, that such a general program may require a bit more overhead in cases where not all features of the general approach are utilized. For example, if $\phi_2(d) = 0$, there is no need to "call" this function and its derivative and unnecessary additions of zeros will be included in the program.

6 Computational experiments

A program was coded in Fortran, double precision arithmetic, compiled by Compaq Fortran 6.6 and ran on a 2.8 GHz computer. The solutions were found to a relative accuracy of 10^{-10}. We experimented with nine different problems as described in Sect. 4. Demand points for all problems were randomly generated in a unit square. We found a solution to these problems in the convex hull of the demand points. All these problems conform to the special structure required for our general approach and the bounds in the original papers were calculated in a different way or with less overhead as follows.

1. The obnoxious facility location problem (equal weights).
2. The Weber problem with some negative weights (weights randomly generated in [-1, 1]).
3. The Huff competitive location problem (ten existing stores, $\lambda = 2$, and all buying power values and attractiveness of facilities are equal).
4. The stochastic weighted minimax location problem (using $T = 2$, $a = 1$).

[1] We thank Atsuo Suzuki for his Fortran program that finds the triangulation based on Sugihara and Iri (1994) subroutines first developed in Ohya et al. (1984).

5. The inventory location problem (using α_i, w_i, A_i, C_i generated in [0,1] and $B_i = 2A_i + C_i$).
6. The unserviced demand problem (using $\phi(d) = e^{-d}$).
7. The unserviced demand problem (using $\phi(d) = 1/(1 + d)$).
8. The gradual covering problem (using $l = 0.1, u = 0.3, w = 5$).
9. The acceleration–deceleration distance Weber problem (with $d_0 = 1$).

In Tables 1–3 we report computational results with these problems. Each problem was run ten times for each value of n. We report the results (minimum, maximum, and average) for the number of iterations, the maximum number of triangles during the branch and bound phase, and the run time in seconds.

6.1 discussion of results

Six of the nine problems (see Tables 1–3) were solved very efficiently in all cases. All $n = 10,000$ problems were solved in less than 100 s with very little variance. The

Table 1 Computational results (first set of three problems)

n	Iterations			Max triangles			Time (seconds)		
	Min	Max	Ave	Min	Max	Ave	Min	Max	Ave
Obnoxious facility									
10	601	1,682	926.1	54	132	74.4	0.00	0.03	0.01
20	597	1,778	1336.3	54	98	77.9	0.01	0.05	0.03
50	1,298	2,357	1857.7	75	131	93.7	0.07	0.12	0.10
100	1,208	4,106	2623.2	83	170	117.8	0.14	0.43	0.28
200	1,093	5,657	3697.2	98	208	150.5	0.25	1.17	0.77
500	2,201	10,581	5355.0	126	293	191.2	1.23	5.35	2.80
1,000	1,656	12,410	6658.7	102	293	209.3	2.14	12.68	7.09
2,000	4,982	13,199	7500.8	169	307	243.5	11.78	28.00	16.70
5,000	4,372	25,483	11725.3	156	615	355.6	33.74	136.74	69.76
10,000	3,991	23,321	11122.3	160	600	350.4	89.09	275.08	157.29
Weber with some negative weights									
10	33	49	39.3	6	13	10.2	0.00	0.00	0.00
20	33	75	45.9	7	20	12.4	0.00	0.00	0.00
50	33	88	46.6	10	21	14.3	0.00	0.02	0.00
100	31	65	44.5	7	19	11.0	0.00	0.02	0.01
200	32	68	53.7	5	17	11.3	0.03	0.05	0.03
500	32	66	50.2	12	22	16.5	0.14	0.18	0.16
1000	36	85	57.2	9	25	15.8	0.58	0.63	0.60
2000	35	63	46.5	11	21	16.3	2.22	2.28	2.25
5000	30	69	57.8	7	24	13.2	13.51	13.80	13.69
10000	38	67	53.3	7	23	15.4	53.86	54.25	54.05
Huff competitive location problem									
10	580	8501	2553.8	127	1,723	543.9	0.01	0.14	0.04
20	969	27897	5980.1	274	4,837	1302.8	0.02	0.99	0.18
50	2,517	38375	10219.4	749	6,288	1917.4	0.16	2.81	0.67
100	5,168	16367	9905.4	762	4,029	2257.8	0.59	1.92	1.15
200	8,054	71394	22952.9	2,067	9,125	4413.6	1.85	18.02	5.47
500	1,7980	47297	27252.8	2,912	9,569	5253.0	10.19	26.98	15.30
1,000	1,3879	113251	36439.1	2,474	14,596	6123.8	15.70	128.83	41.07
2,000	1,4574	71753	39192.3	1,849	10,796	5885.3	33.51	158.17	87.84
5,000	1,6310	86444	50891.5	2,557	12,076	7000.0	101.25	482.83	290.21
10,000	19,295	168686	57944.7	2,695	38,635	9663.8	260.83	1870.78	680.90

Table 2 Computational results (second set of three problems)

n	Iterations			Max triangles			Time (seconds)		
	Min	Max	Ave	Min	Max	Ave	Min	Max	Ave
Stochastic weighted minimax									
10	77	136	101.2	47	84	62.4	0.00	0.02	0.00
20	76	120	92.8	45	67	55.6	0.00	0.02	0.00
50	67	108	82.7	45	62	52.3	0.00	0.02	0.01
100	68	115	90.3	40	64	46.7	0.01	0.04	0.02
200	75	100	88.0	41	67	51.5	0.06	0.08	0.06
500	76	97	87.2	36	57	42.7	0.26	0.30	0.29
1,000	66	108	82.4	32	56	42.1	0.95	1.03	0.98
2,000	69	118	86.7	25	67	43.5	3.60	3.75	3.65
5,000	68	106	81.6	25	52	36.3	21.53	21.84	21.65
10,000	52	138	85.1	29	52	38.5	84.77	86.23	85.28
The inventory location problem									
10	112	260	156.8	11	34	22.5	0.00	0.02	0.00
20	112	273	160.9	17	37	24.2	0.00	0.02	0.00
50	107	156	130.4	19	27	21.9	0.00	0.02	0.01
100	91	245	124.4	16	41	22.4	0.01	0.05	0.02
200	90	207	121.9	18	35	22.7	0.03	0.06	0.05
500	91	150	113.9	18	26	21.0	0.19	0.22	0.21
1,000	85	163	108.7	17	29	20.8	0.65	0.75	0.69
2,000	80	206	127.8	15	35	25.4	2.40	2.69	2.51
5,000	74	111	89.2	16	25	19.3	14.28	14.49	14.37
10,000	71	122	91.4	17	28	20.2	56.86	57.44	57.11
The unserviced demand problem with $\phi(d) = e^{-d}$									
10	133	432	215.9	15	38	27.2	0.00	0.02	0.00
20	119	199	162.4	20	36	26.9	0.00	0.02	0.01
50	130	510	197.7	22	64	31.5	0.01	0.06	0.02
100	133	216	161.6	22	34	26.8	0.03	0.06	0.04
200	113	191	145.5	20	31	25.5	0.08	0.11	0.09
500	88	205	145.8	19	37	28.0	0.31	0.42	0.36
1,000	95	275	147.3	22	49	28.7	1.10	1.42	1.19
2,000	93	215	124.3	19	38	23.7	3.98	4.42	4.10
5,000	82	299	140.4	20	61	29.8	23.52	25.45	24.03
10,000	87	148	109.6	20	31	24.2	92.59	93.67	92.97

number of iterations and the maximum number of triangles were very small. This is especially impressive since an accuracy of $\epsilon = 10^{-10}$ was applied to all problems while the customary accuracy in the literature is $\epsilon = 10^{-5}$ or 10^{-6}.

The gradual covering problem (see Table 3) had very few cases (for $n = 10$ and $n = 50$) where more iterations were required and the maximum number of triangles was quite high. This is caused by the discontinuity in the first derivative for this particular function ("jumping" from 0 to 5 and back), which necessitated more triangle splits when the discontinuity occured inside a triangle. Increasing ϵ to 10^{-6} shortens the run time for these few cases.

The obnoxious facility location problem (see Table 1) requires more iterations, triangles, and run time than the seven problems mentioned above. In one case of $n = 10,000$ the computer run time was close to 5 min. The reason is that the function $1/d^2$ diverges to infinity when d approaches zero. When a demand point is a vertex of a triangle, $d_{\min} = 0$ and the bounds have large errors and $\phi(d_{\min})$ does not even

Table 3 Computational results (third set of three problems)

n	Iterations			Max triangles			Time (seconds)		
	Min	Max	Ave	Min	Max	Ave	Min	Max	Ave
The unserviced demand problem with $\phi(d) = 1/1 + d$									
10	135	434	231.3	19	40	29.6	0.00	0.02	0.00
20	138	486	210.8	19	74	32.3	0.00	0.02	0.01
50	138	493	208.6	25	61	33.2	0.00	0.03	0.01
100	131	305	186.1	20	46	30.1	0.01	0.03	0.03
200	127	239	165.5	22	36	28.9	0.05	0.07	0.06
500	124	227	174.2	25	43	32.3	0.18	0.24	0.21
1,000	104	309	164.9	23	58	31.9	0.59	0.79	0.66
2,000	111	223	135.5	23	43	26.9	2.16	2.37	2.21
5,000	100	311	158.1	25	62	33.8	12.57	13.53	12.80
10,000	91	164	119.5	23	32	27.3	48.81	49.51	49.09
The gradual covering problem									
10	38	20,277	3323.8	15	2,426	488.3	0.00	0.37	0.06
20	82	6,337	757.4	24	826	112.9	0.00	0.14	0.02
50	50	22,766	2346.2	24	3,964	426.9	0.00	1.36	0.14
100	49	92	67.6	27	57	37.9	0.00	0.02	0.01
200	43	82	60.8	20	46	33.0	0.03	0.05	0.04
500	36	76	60.5	16	30	23.4	0.15	0.18	0.17
1,000	39	91	61.6	10	25	18.8	0.58	0.64	0.61
2,000	38	81	53.3	13	26	19.1	2.22	2.32	2.26
5,000	35	78	52.3	8	23	16.9	13.55	13.73	13.63
10,000	36	84	51.0	10	19	13.2	53.67	54.16	53.85
The acceleration–deceleration distance weber problem									
10	33	49	39.3	6	13	10.2	0.00	0.00	0.00
20	33	75	45.9	7	20	12.4	0.00	0.00	0.00
50	33	88	46.6	10	21	14.3	0.00	0.02	0.00
100	31	65	44.5	7	19	11.0	0.00	0.02	0.01
200	32	68	53.7	5	17	11.3	0.03	0.05	0.03
500	32	66	50.2	12	22	16.5	0.14	0.18	0.16
1,000	36	85	57.2	9	25	15.8	0.58	0.63	0.60
2,000	35	63	46.5	11	21	16.3	2.22	2.28	2.25
5,000	30	69	57.8	7	24	13.2	13.51	13.80	13.69
10,000	38	67	53.3	7	23	15.4	53.86	54.25	54.05

exist. We therefore replaced d^2 by $d^2 + 10^{-10}$. Even though $\phi(0) = 10^{10}$ may not be good enough, it solved this issue and the resulting run times (reported in Table 1) are reasonable. Note, that at the solution point all distances are relatively large because a small distance results in a high value of the objective function. Therefore, adding 10^{-10} to each d^2 hardly affect the optimal value of the objective function.

The competitive Huff location problem performed the worst. In one case of $n =$ 10, 000 demand points it took over half an hour to solve the problem, required almost 170,000 iterations and 40,000 triangles. The reason is that when a demand point is close to an existing facility, the value of h_i may be very large. The function $b_i/(1 + h_i d^2)$ is actually negligible and does not contribute much to the value of the objective function. However, our "trick" of adding $b_i h_i d^2$ to convexify the objective function (and subtracting it as $\phi_2(d)$) did not work well and caused a large error in the upper bound. One should try to find a better convexification scheme. The contrived upper bound in Drezner and Drezner (2004) does not suffer from this deficiency and works better.

7 Conclusions

We proposed a general approach to optimally solve a certain type of facility location problems in the plane. The location problem is a minimization or maximization of a sum of functions, each a function of the Euclidean distance between the facility and a demand point. Such a function is expressed as a difference between two convex functions *of the distance*. It should be emphasized that a convex function of the distance is a weaker condition than requiring a convex function *of the location*. We propose bounds to be used in the BTST method (Drezner and Suzuki 2004) for the solution of such problems. The general procedure was tested on nine problems. All nine problems were solved very efficiently, demonstrating the effectiveness of the proposed general approach.

References

Berman O., Wang, J., Drezner, Z., Wesolowsky, G.O.: A probabilistic minimax location problem on the plane. Ann. Oper. Res. **122**, 59–70 (2003)

Drezner T., Drezner, Z.: Finding the optimal Solution to the Huff competitive location model. Comput. Manage. Sci. **1**, 193–208 (2004)

Drezner T., Drezner, Z.: Equity models in planar location. Comput. Manage. Sci. (2006a) (accepted for publication)

Drezner T., Drezner, Z.: Lost demand in a competitive environment. in review (2006b)

Drezner Z., Drezner, T., Wesolowsky, G.O.: Location with acceleration-deceleration distance. in review (2006)

Drezner Z., Scott, C.H.: Locating a service facility with some unserviced demand. IMA J Manage Math. in press (2006)

Drezner Z., Scott, C.H., Song, J.S.: The central warehouse location problem revisited. IMA J Manage Math **14**, 321–336 (2003)

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

Drezner Z., Wesolowsky, G.O.: The Weber problem on the plane with some negative weights. Inf. Syst. Oper. Res. **29**, 87–99 (1991)

Drezner Z., Wesolowsky, G.O., Drezner, T.: The Gradual covering problem. Nav. Res Logist. **51**, 841–855 (2004)

Hansen P., Peeters, D., Thisse, J.: On the location of an obnoxious facility. Sistemi Urbani **3**, 299–317 (1981)

Krarup J.: On a complementary problem of Courant and Robbins. Location Sci. **6**, 337–354 (1998)

Maranas C.D., Floudas, C.A.: A global optimization method for Weber's problem with attraction and repulsion. In Hager, W.W., Hearn, D.W., Pardalos, P.M. (eds.) Large Scale Optimization: State of the Art. Kluwer, Dordrecht, The Netherlands (1994)

Ohya T., Iri, M., Murota, K.: Improvements of the incremental method of the Voronoi diagram with computational comparison of various algorithms. J. Oper. Res. Soc. Jpn **27**, 306–337 (1984)

Plastria F.: GBSSS, the generalized big square small square method for planar single facility location. Euro. J. Oper. Res. **62**, 163–174 (1992)

Sugihara K., Iri, M.: A robust topology-oriented incremental algorithm for Voronoi diagram. Int. J. Comput Geome. Appli. **4**, 179–228 (1994)

Tellier L.N., Polanski, B.: The Weber problem: frequency and different solution types and extension to repulsive forces and dynamic processes. J. Reg. Sci. **29**, 387–405 (1989)

Tuy H.: Convex Analysis and Global Optimization. Kluwer Academic Publishers, Dordrecht (1998)

Tuy H., Al-Khayyal, F., Zhou, F.: A DC optimization method for single facility location problems. J. Global Optim. **7**, 209-227 (1995)