FACTA UNIVERSITATIS (NIŠ)
Ser. Math. Inform. Vol. 28, No 3 (2013), 271–284

ALGORITHM FOR CONSTRAINED WEBER PROBLEM WITH FEASIBLE REGION BOUNDED BY ARCS

Lev A. Kazakovtsev

Abstract. The author proposes a heuristic algorithm for a special case of constrained continuous planar facility location problem with the connected region of feasible solutions bounded by arcs. The algorithm implements a modified Weiszfeld procedure and a special procedure for searching for the closest feasible solution for a given point. Example problems were solved. The convergence of the algorithm is proved experimentally for randomly generated problems. The results were compared with the results of the sample search.

1. Introduction

The location problem stated by Alfred Weber in 1909 [22, 24, 6] is a continuous optimization problem of searching for such a point $X^* \in \mathbb{R}^n$ that

$$X^* = \arg\min_{X \in \mathbb{R}^n} f(X) = \arg\min_{X \in \mathbb{R}^n} \sum_{i=1}^{N} w_i \|A_i - X\|. \tag{1.1}$$

Here, $A_i \in \mathbb{R}^n, i \in \left\{\overline{1, N}\right\}$ are some known points called demand points, $w_i \in \mathbb{R}, w_i \geqslant 0$ are some weight coefficients, $\|\,.\,\|$ is some norm $\mathbb{R}^n \to \mathbb{R}$ [15].

Examples of the Weber problems include the warehouse location [5, 6], positioning computer and communication networks [14], locating base stations of wireless networks [20]. They are also useful in approximation theory, statistical estimation problem solving [17], signal and image processing and other engineering applications.

Problem (1.1) was originally formulated by Weber (see [22, 24]) with Euclidean norm ($\|\,.\,\| = l_2(.)$) but generalized to l_p norms and other metrics [24, 3, 20]. In [18] and [13], authors consider norm approximation and approximated solution for Weber problems with an arbitrary metric. In [19], authors solve approximately

Received September 6, 2013.; Accepted December 17, 2013.
2010 *Mathematics Subject Classification.* Primary 90B85; Secondary 90C27, 90B80

the Weber problem with a special metric induced by measuring distances between points of an arbitrary surface in R^3.

In [11, 9, 25], the authors propose methods for the constrained Weber and minimax location problems. Algorithms for the constrained Weber problem with polyhedral feasible region are proposed in [11, 10] and many other papers.

The metric used in practically important location problems depends on various factors including properties of the transportation means. In the case of public transportation systems, the price usually depends on the distance. However, some minimum price is usually defined. For example, the initial fare of the taxi cab may include some distance, usually 1-5 km. Having rescaled the distances so that this distance included in the initial price is equal to 1, we can define the price function d_P as

$$d_P(X,Y) = \max\{\|X-Y\|, 1\} \forall X, Y \in \mathbb{R}^n \tag{1.2}$$

where $\|\,.\,\|$ is some norm.

In this paper, we consider $\|\,.\,\|$ as Euclidean norm in $\mathbb{R}^2$ only ($\|\,.\,\|_2$).

The very similar metric, the Radar Screen [1] metric is a norm metric with the distance function defined as

$$d_{rs}(X,Y) = \|X-Y\|_{rs} = \min\{1, \|X-Y\|_2\} \forall X, Y \in \mathbb{R}^n. \tag{1.3}$$

The Weber problem with the Radar Screen metric is a special case of the problem considered in [7]. Unlike (1.3), the distance function if (1.2) is convex.

However, in both cases, (1.2) and (1.3), the problem is decomposed into series of constrained location problems with Euclidean metric where the area of the feasible solutions is bounded by arcs. Each of the problems has the feasible region equal to same intersection of the discs (see Fig. 1.1) with centres in the demand points.

The the sub-problems in the location problem (1.1) with both distance metrics (1.2) and (1.3) include the constraints

$$\|X - A_i\|_2 \leqslant 1 \forall i \in \mathcal{S}_< \tag{1.4}$$

and

$$\|X - A_i\|_2 \geqslant 1 \forall i \in \mathcal{S}_>, \tag{1.5}$$

where $\mathcal{S}_<, \mathcal{S}_> \in \{\overline{1,N}\}$, $\mathcal{S}_< \cap \mathcal{S}_> = \varnothing$ are some subsets of the set of demand point indexes.

In [7, 4], authors propose an approach based on the convex Mixed-Integer Nonlinear Program (MINLP).

We propose a much simpler method based on the Weiszfeld procedure [23] and prove its convergence experimentally.

The paper is organized as follows. In Chapter 2, we restate some basic definitions and describe existing algorithms and investigate some features of the objective function. In Chapter 3, we restate the algorithm for the Weber problem with the connected feasible region bounded by arcs. In chapter 4, we give simple examples and describe methods used for proving the convergence of the algorithm.

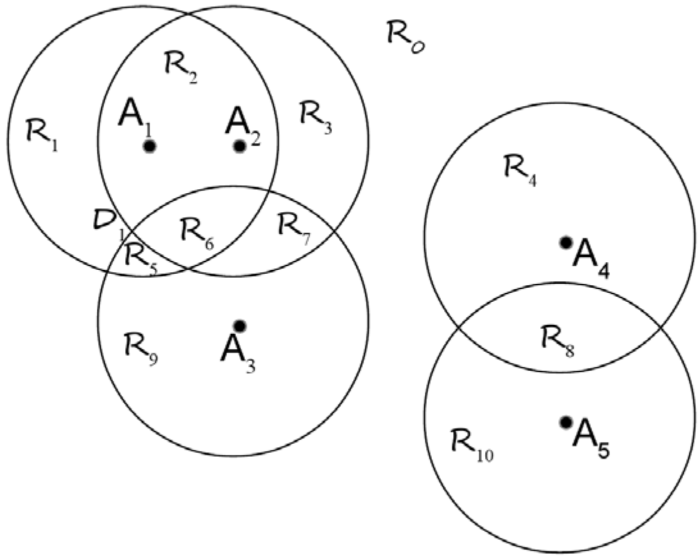

FIG. 1.1: Illustration of a set of problems (1.1) with constraints (1.4)-(1.5). Each region R_i has its own set of constraints.

2. Preliminaries

The most common algorithm for Weber problem with the metrics induced by the l_p norms is Weiszfeld procedure [23, 6] described as follows.

Algorithm 2.1. Weiszfeld procedure.

Require: Coordinates and weights of the demand points $A_i = (a_1^i, a_2^i)$, w_i, $i = \overline{1, N}$, number of iterations N_{iter}.

Step 1: Calculate the initial point $X^* = (x_1^*, x_2^*) : x_r = \frac{\sum_{i=1}^{N} a_r^i w_i}{\sum_{i=1}^{N} w_i}$ $\forall\ r \in \{1, 2\}$; $n_{iter} = 0$.

Step 2: while $n_{iter} < N_{iter}$ do:

Step 2.1: $n_{iter} = n_{iter} + 1$; $d_{denom} = \sum_{i=1}^{N} w_i / \|A_i - X^*\|_2$.

Step 2.2: $x_r^{**} = \sum_{i=1}^{N} \frac{x_r^* w_i}{\|X^* - A_i\|_2 \cdot d_{denom}}$ $\forall i \in \{1, 2\}$; $X^{**} = (x_1^{**}, x_2^{**})$.

Step 2.3: $X^* = X^{**}$.

Step 2.4: Continue Step 2.

Step 3: STOP, return X^*.

In Step 2 of the Algorithm 2.1, other stop condition can be used [5]:

Algorithm 2.2. Weiszfeld procedure, other stop condition.

Require: Coordinates and weights of the demand points $A_i = (a_1^i, a_2^i), w_i, i = \overline{1, N}$, prespecified tolerance ε.

Step 1: Calculate the initial point (see Algorithm 2.1); $\Delta = +\infty$.

Step 2: while $\Delta > \varepsilon$ do:

Step 2.1: $n_{iter} = n_{iter} + 1$; $d_{denom} = \sum_{i=1}^{N} w_i / \|A_i - X^*\|_2$.

Step 2.2: $x_r^{**} = \sum_{i=1}^{N} \frac{x_r^* w_i}{\|X^* - A_i\|_2 \cdot d_{denom}}\ \forall r \in \{1, 2\}$; $X^{**} = (x_1^{**}, x_2^{**})$.

Step 2.3: $\Delta = \|X^* - X^{**}\|_2$; $X^* = X^{**}$.

Step 2.4: Continue Step 2.

Step 3: STOP, return X^{**}.

The feasible set of our problem given by the constraints (1.4) and (1.5) is generally non-convex, the objective function $f(X)$ in (1.1) is convex [10].

For constrained optimization problems with convex objective functions, the solution coincides with the solution of the unconstrained problem or lays on the border of the forbidden region [9] (moreover, the solution of the constrained problem is said to be visible from the solution of the unconstrained problem).

Corollary 2.1. *If X^* is a solution of the constrained problem (1.1) with constraints (1.4) and (1.5) then it is the solution of the unconstrained problem (1.1) or* $\exists i \in \{\overline{1, N} : \|A_i - X^*\|_2 = 1\}$.

The algorithms above solve the unconstrained problem only. Step 1 is optional, any point can be chosen as initial. However, choosing one of demand points as initial leads to extremely slow convergence [2, 21]. Choosing the median point improves the convergence in most cases.

Step 2.2 of Algorithms 2.1,2.2 can lead to generating new point X^{**} outside the feasible region given by (1.4)-(1.5). Let us denote this region $\mathcal{R}_f$. We assume that $\mathcal{R}_f \neq \emptyset$.

For an arbitrary point $X \in \mathbb{R}^2$, let us denote the closest point in $\mathcal{R}_f$:

$$C(X) = \arg\min_{X' \in \mathcal{R}_f} \|X - X'\|_2 = \begin{cases} X, & X \in \mathcal{R}_f, \\ \arg\min_{X' \in \mathcal{R}_f} \|X - X'\|_2, & X \notin \mathcal{R}_f. \end{cases} \tag{2.1}$$

Convergence of Weiszfeld procedure is proved for Weber problems with l_p norms [6] where $p \in [1, 2]$. Despite slow convergence problems in special cases [2, 21], this simple procedure based on the first-order necessary conditions is efficient for planar unconstrained location problems.

The algorithm which we propose is based on the hypothesis below.

Hypothesis 1. If Step 2.2 of Algorithms 2.1,2.2 generates new interim solution $X^{**} \notin \mathcal{R}_f$ then, having added to Step 2.2 a substitution of this new solution with $C(X^{**})$, we have an algorithm for the constrained problem (1.1),(1.4)-(1.5). Let us denote the sets of points

$$\mathcal{R}_{Ai} = \begin{cases} \{X \in \mathbb{R}^2 |\ \|X - A_i \leqslant 1\|_2\}, & i \in \mathcal{S}_< \\ \{X \in \mathbb{R}^2 |\ \|X - A_i \geqslant 1\|_2\}, & i \in \mathcal{S}_> \end{cases} \quad \forall i \in \mathcal{S}_< \cup \mathcal{S}_>.$$

Obviously,

$$\mathcal{R}_f = \bigcap_{i\in(\mathcal{S}_< \cup \mathcal{S}_>)} \mathcal{R}_{Ai}.$$

For given $i \in \mathcal{S}_< \cup \mathcal{S}_>$, let us denote $C_{Ai}(X)$ the point of the set $\mathcal{R}_{Ai}$ closest to a given point X.

Lemma 2.1. *If $X^{**} \notin \mathcal{R}_{Ai}$, $i \in \mathcal{S}_< \cup \mathcal{S}_>$, $X^{**} \neq A_i$ then the point $C_{Ai}(X^{**})$ closest to X^{**} is an intersection point of the circle with its center in A_i and a line segment connecting A_i and X^{**}.*

Proof. The distance function $D(X) = \|X - X^{**}\|_2$, $X \in \mathbb{R}^2$ is strictly convex. Therefore, its minimum point in $\mathcal{R}_{Ai}$ coincides with the unconstrained minimum point or belongs to the borderline of the region $\mathcal{R}_{Ai}$. Since the unconstrained minimum point is obviously X^{**} and $X^{**} \notin \mathcal{R}_{Ai}$,

$$\arg\min_{X\in\mathcal{R}_{Ai}} D(X) = \arg\min_{X\in\mathbb{R}^2:\|X-A_i\|_2=1} \|X - X^{**}\|_2. \tag{2.2}$$

Let X'' be a minimum point of the problem (2.2).

If X^{**} is outside the circle, from the triangle inequality,

$$D(X'', X^{**}) \geqslant D(X'', A_i) - D(A_i, X^{**}) = D(A_i, X^{**}) - 1,$$

$$D(X'', X^{**}) = D(X'', A_i) - D(A_i, X^{**}) = D(A_i, X^{**}) - 1$$

if and only if points X^{**}, X'' and A_i are collinear.

Thus, the minimum of $D(X'', X^{**}) = D(A_i, X^{**}) - 1$ is attained when X'' is the intersection of the line segment connecting A_i and X^{**} with the circle.

If X^{**} is inside the circle,

$$D(X'', X^{**}) \geqslant D(A_i, X^{**}) - D(X'', A_i) = 1 - D(A_i, X^{**}).$$

Thus, the minimum of $D(X'', X^{**}) = 1 - D(A_i, X^{**})$ is attained when X'' is the intersection of the line segment connecting A_i and X^{**} with the circle. □

For given $i \in (\mathcal{S}_< \cup \mathcal{S}_>)$, let us denote a function on $\mathbb{R}^2$

$$\mathcal{G}_{Ai}(X) = \begin{cases} 0, & X \in \mathcal{R}_{Ai} \\ \|X - C_{Ai}(X)\|_2, & X \notin \mathcal{R}_{Ai} \end{cases}. \tag{2.3}$$

Lemma 2.2. *Let us chose two indexes $i, j \in \mathcal{S}_< \cup \mathcal{S}_> : \mathcal{R}_{Ai} \cap \mathcal{R}_{Aj} \neq \emptyset$, $A_i \neq A_j$ and an arbitrary point $X' \notin \mathcal{R}_{Ai} \cap \mathcal{R}_{Aj}$. The point $C_{Ai,Aj}(X') \in \mathcal{R}_{Ai} \cap \mathcal{R}_{Aj}$ closest to the point X' coincides with $C_{Ai}(X')$, $C_{Aj}(X')$ or an intersection point of two circles of radius 1 with centers in A_i and A_j.*

Proof. Since the function $f_R(X) = \|X - X'\|$ is convex and $X' \notin \mathcal{R}_{Ai} \cap \mathcal{R}_{Aj}$, its minimum in $\mathcal{R}_{Ai} \cap \mathcal{R}_{Aj}$ belongs to the borderline of the region $\mathcal{R}_{Ai} \cap \mathcal{R}_{Aj}$ which consists of tho arcs of the circles with centers in A_i and A_j joint in two intersection points of the circles. Let us denote them $I_{i,j}$ and $I'_{i,j}$.

If $C_{Ai,Aj}(X')$ belongs to the circle with center in A_i ($\|X' - A_i\| = 1$) and $C_{Ai,Aj}(X') \notin \{I_{i,j}, I'_{i,j}\}$ then, from Lemma 2.1, $C_{Ai,Aj}(X') = C_{Ai}(X')\}$.

Analogously, If $C_{Ai,Aj}(X')$ belongs to the circle with center in A_j then

$$C_{Ai,Aj}(X') \notin \{I_{i,j}, I'_{i,j}\}$$

and

$$C_{Ai,Aj}(X') = C_{Aj}(X').$$

□

Let us denote the penalty function on $\mathbb{R}^2$

$$\mathcal{G}(X) = \max_{i \in \mathcal{S}_< \cup \mathcal{S}_>} \mathcal{G}_{Ai}(X). \tag{2.4}$$

Obviously, $\mathcal{G}(X) \geqslant 0 \forall X \in \mathbb{R}^2$ and $\mathcal{G}(X) = 0 \forall X \in \mathcal{R}_f$.

From convexity of each of functions $\mathcal{G}_{Ai}(\cdot)$, function $\mathcal{G}(\cdot)$ is convex.

Thus, the problem of searching for a point $X' \in \mathcal{R}_f$ can be restated as problem of minimizing $\mathcal{G}(X)$.

3. Algorithm for the constrained Weber problem

Based on Hypothesis 1, we propose the algorithm below.

Algorithm 3.1. Weiszfeld procedure for problem (1.1) with constraints (1.4) and (1.5).

Require: Coordinates and weights of the demand points $A_i = (a_1^i, a_2^i), w_i, i = \overline{1, N}$, prespecified tolerance ε, constraints (1.4) and (1.5) specified by sets $\mathcal{S}_<$ and $\mathcal{S}_>$.

Step 1: Calculate the initial point $X^* \in \mathcal{R}_f$ (here, $\mathcal{R}_f$ is the feasible set bounded by constraints); $\Delta = +\infty$.

Step 2: while $\Delta > \varepsilon$ do:

Step 2.1: $n_{iter} = n_{iter} + 1$; $d_{denom} = \sum_{i=1}^{N} w_i / \|A_i - X^*\|_2$.

Step 2.2: $x_r^{**} = \sum_{i=1}^{N} \frac{x_r^* w_i}{\|X^* - A_i\|_2 \cdot d_{denom}}$ $\forall r \in \{1, 2\}$; $X^{**} = (x_1^{**}, x_2^{**})$.

Step 2.3: if $\mathcal{G}(X^{**}) > 0$ then $X^{***} = C(X^{**})$ else $X^{***} = X^{**}$.

Step 2.4: $\Delta = \|X^* - X^{***}\|$; $X^* = X^{***}$.

Step 2.5: Continue Step 2.

Step 3: STOP, return X^{**}.

For this algorithm, we must explain two supplementary procedures for searching for an initial point $X^* \in \mathcal{R}_f$ and for searching for the closest point $\mathcal{C}(X^{**}) \in \mathcal{R}_f$.

We use the initial feasible point closest to the median (see algorithm below).

Algorithm 3.2. Calculate the initial point $X^* \in \mathcal{R}_f$ (Step 1 of Algorithm 3.1).

Require: Coordinates and weights of the demand points $A_i = (a_1^i, a_2^i), w_i, i = \overline{1,N}$, constraints (1.4) and (1.5) specified by sets $\mathcal{S}_<$ and $\mathcal{S}_>$.

Step 1: $x_r = \frac{\sum_{i=1}^N w_i a_r^i}{\sum_{i=1}^N w_i} \forall r \in \{1,2\}$. We assume that $X = (x_1, x_2)$.

Step 2: $X^* = \mathcal{C}(X)$.

Step 3: STOP, return X^*.

Based on Lemma 2.1 and Lemma 2.2, we use the algorithm below for calculating $\mathcal{C}(X)$.

Algorithm 3.3. Calculate $\mathcal{C}(X)$ (Step 2 of Algorithm 3.2 and Step 2.3 of Algorithm 3.1).

Require: Coordinates of the demand points $A_i = (a_1^i, a_2^i), i = \overline{1,N}$, constraints (1.4) and (1.5) specified by sets $\mathcal{S}_<$ and $\mathcal{S}_>$, initial point X.

Step 1: Calculate $g_i = \mathcal{G}_{Ai}(X) \forall i \in \mathcal{S}_< \cup \mathcal{S}_>$; $G = \sum_{i \in \mathcal{S}_< \cup \mathcal{S}_>} g_i$.

Step 2: while $G > 0$ do:

Step 2.1: Choose two indexes $i' = \arg \max_{i \in \mathcal{S}_< \cup \mathcal{S}_>} g_i$; $i'' = \arg \max_{i \in (\mathcal{S}_< \cup \mathcal{S}_>) \setminus \{i'\}} g_i$.

Step 2.2: if $g_{i''} = 0$ then $X = C_{Ai'}(X)$; Calculate $g_i = \mathcal{G}_{Ai}(X) \forall i \in \mathcal{S}_< \cup \mathcal{S}_>$; $G = \sum_{i \in \mathcal{S}_< \cup \mathcal{S}_>} g_i$; Continue Step 2.

Step 2.3: Calculate coordinates of two intersection points of the circles in $A_{i'}$ and $A_{i''}$: $I_1, I_2 = \mathcal{P}_{intersect}(A_{i'}, A_{i''})$; Calculate $C_1 = C_{Ai'}(X)$; $C_2 = C_{Ai''}(X)$.

Step 2.4: $X = \arg \min_{X \in \{C_1, C_2, I_1, I_2\}} \mathcal{G}(X)$; $G = \mathcal{G}(X)$.

Step 2.5: Continue Step 2.

Step 3: STOP, return X.

In Algorithm 3.3, we use formulas (2.3) and (2.4). To obtain $C_{Ai}(X)$ (an intersection point of the line connecting X and A_i and the circle with center in A_i) for given $X' = (x_1', x_2')$ and $A_i = (a_1^i, a_2^i)$, we use equations below (from equations of a line and circle).

$$b = \frac{x_1' - a_1^i}{x_2' - a_2^i}, \tag{3.1}$$

$$a = a_1^i - b a_2^i. \tag{3.2}$$

From the equation

$$(x_1 - a_1^i)^2 + (x_2 - a_2^i)^2 = 1$$

of the circle with center in A_i, we have an equation, we have an equation

$$(1 + b^2)x_2^2 + (2b(a - a_1^i) - 2a_2^i)x_2 + (a_2^i)^2 + (a - a_1^i)^2 - 1 = 0.$$

Having solving it, assuming $C_{Ai}(X') = (c_1, c_2)$, we obtain:

$$c_2' = -(2b(a - a_1^i) - 2a_2^i) + D, \quad c_2'' = -(2b(a - a_1^i) - 2a_2^i) + D, \tag{3.3}$$

$$c_1' = a_1^i + \frac{(c_2' - a_2^i)(x_1' - a_2^i)}{x_2' - a_2^i} + a_1^i, \quad c_1'' = a_1^i + \frac{(c_2'' - a_2^i)(x_1' - a_2^i)}{x_2' - a_2^i} + a_1^i, \tag{3.4}$$

$$(c_1, c_2) = \arg \min_{X \in \{(c_1', c_2'), (c_1'', c_2'')\}} \|X - X'\| \tag{3.5}$$

where

$$D = \sqrt{(2b(a - a_1^i) - 2a_2^i)^2 - 4(1 + b^2)(a_2^i)^2 + (a - a_1^i)^2 - 1}. \tag{3.6}$$

For calculating (3.1)-(3.6), we must take into consideration two possible special cases. When $X' = A_i$, distances from X' to all points on the circle with center in X' are equal. In this case, we can use any of them as the minimum point. We use the point

$$(c_1, c_2) = (x_1', x_2' - 1).$$

In the case when $a_2^i = x_2'$, we use formulas

$$c_1' = c_1'' = x_1',$$

$$c_2' = x_2' + 1, \quad c_2'' = x_2' - 1$$

and choose (c_1, c_2) in accordance with (3.5).

4. Numerical examples

Let us consider problem shown in Fig. 4.1, Case 1. In this problem, the coordinates and weights of the demand points are: $A_1 = (0, 0.75)$, $w_1 = 3$, $A_2 = (0.3, 0.5)$, $w_2 = 2$, $A_3 = (0.6, 0.5)$, $w_3 = 3$, $A_4 = (1, 2)$, $w_4 = 6$. The feasible region $\mathcal{R}_f$ is bounded by four constraints

$$\|X - A_1\|_2 \leqslant 1, \quad \|X - A_2\|_@ \geqslant 1,$$

$$\|X - A_3\|_2 \geqslant 1, \quad \|X - A_4\|_2 \leqslant 1.$$

I.e., $\mathcal{S}_< = \{1, 4\}$, $\mathcal{S}_> = \{2, 3\}$.

We implement Algorithm 3.1 with $\varepsilon = 0.00001$.

The median point (Step 1 of Algorithm 3.2) is

$$X^{**} = \frac{\sum_{i=1}^{4} A_i w_i}{\sum_{i=1}^{4} w_i} = (0.6, 1.19642857).$$

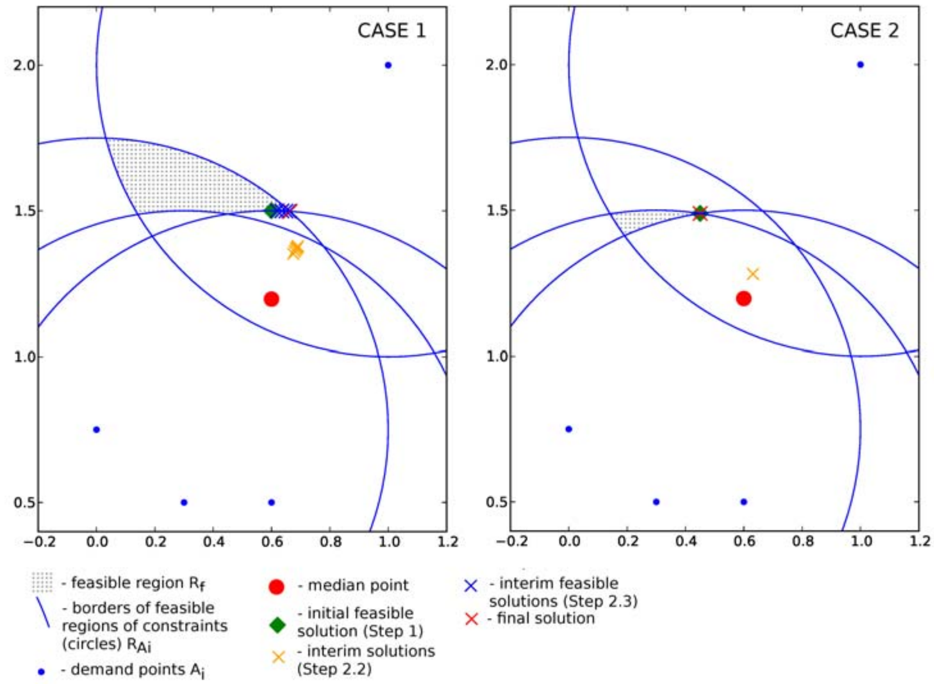

FIG. 4.1: Example problems and solution process

This median point X^* is outside $\mathcal{R}_f$. To obtain the closest feasible point, Algorithm 3.3 is implemented. At Step 1 of Algorithm 3.3, $g_1 = 0$, $g_4 = 0$, $g_2 = 0.241704046$, $g_3 = 0.30357143$, $G = 0.545275476 > 0$. Thus, Step 2 of Algorithm 3.3 is performed.

At Step 2.1 of Algorithm 3.3, $i' = 3$, $i'' = 2$. At Step 2.2, $g_{i''} = g_2 \neq 0$.

At Step 2.3 of Algorithm 3.3, the points of intersection of the circles with centers in A_2 and A_3 are $I_1 = (0.45, 1.4886856)$ and $I_2 = (0.45, -0.4886856)$. The points closest to the circles with centers in A_2 and A_3 are $C_1 = C_{A3} = (0.6, 1.5)$ and $C_2 = C_{A2} = (0.6956239, 1.41841261)$.

At Step 2.4 of Algorithm 3.3, $\mathcal{G}(C_1) = 0$, $\mathcal{G}(C_2) = 0.07662269$, $\mathcal{G}(I_i) = 0$, $\mathcal{G}(I_2) = 1.86662994$. Therefore, the new point is $X = C_1 = (0.6, 1.5)$. Since $G = 0$, this is the returned value of Algorithm 3.3 and Algorithm 3.2. This value is used as $X^* = (0.6, 1.5)$ at Step 1 of Algorithm 3.1.

At Steps 2.1 and 2.2 of Algorithm 3.1,

$$d_{denom} = 17.4095535,$$

$$X^{**} = (0.67463657, 1.35220493).$$

This point is outside $\mathcal{R}_f$.

Table 4.1: Some table here.

n_{iter}	X^*	d_{denom}	X^{**}	Δ
1	(0.6, 1.5)	17.4095535	(0.67463657, 1.35220493)	0.01937229
2	(0.619371390, 1.49981236)	17.5361094	(0.67960047, 1.36051086)	0.01628994
3	(0.63565516, 1.49936415)	17.64033129	(0.68362543, 1.367251455)	0.01360489
4	(0.64924779, 1.49878659)	17.7251354	(0.68687416, 1.37269171)	0.01129761
5	(0.66052837, 1.49816648)	17.79357911	(0.689489, 1.37706744)	0.003214
6	(0.66373616, 1.49796678)	17.81267195	(0.69021857, 1.3782875)	0

At Steps 2.3 and 2.4, after implementing Algorithm 3.3,

$$X^{***} = (0.61937139, 1.49981236),$$

$$\Delta = 0.01628994.$$

The interim results of further iterations are shown in Table 4.1.

The problem has been solved in 6 iterations.

If the initial feasible point of the problem is an intersection point of 2 circles with centers in the demand points, just one iteration is needed. Let us solve a problem with same coordinates of the demand points and weights but other feasible region (see Fig. 4.1, Case 2). In this case, the constraints are

$$\|X - A_1\|_2 \leqslant 1, \quad \|X - A_2\|_2 \leqslant 1,$$

$$\|X - A_3\|_2 \geqslant 1, \quad \|X - A_4\|_2 \leqslant 1,$$

$$\mathcal{S}_< = \{1, 2, 4\}, \quad \mathcal{S}_> = \{3\}$$

.

We have the same median point (0.6, 1.196428571428) outside the new feasible region.

This median point $X^* = (0.6, 1.19642857)$ is outside $\mathcal{R}_f$. After Algorithm 3.2, the initial feasible point is $C(X^*) = (0.45000325, 1.4886855)$. Having performed Steps 2.1 – 2.4, we have

$$d_{denom} = 16.45815067, \quad X^{**} = (0.63128532, 1.2808757),$$

$$X^{***} = (0.45000325, 1.4886855). \quad \Delta = 0.$$

Therefore, in Case 2, the initial and final solutions coincide.

Our algorithm is heuristic (Hypothesis 1 is not proved analytically). To prove our Hypothesis experimentally, we generated the example problems, solved it with our algorithm and compared the results with the results of the grid search.

For problem generating, we used the algorithm below.

Algorithm 4.1. Example problems generating.

Require: Number of demand points N.

Step 1: $\mathcal{S}_< = \emptyset$; $\mathcal{S}_> = \emptyset$.

Step 2: For $i \in \{\overline{1,N}\}$ do:

Step 2.1: $x_1 = 4 \cdot Random()$; $x_2 = 4 \cdot Random()$; $X = (x_1, x_2)$; If $\exists A \in \{A_1, ..., A_{i-1}\}$: $\|X - A\|_2 < 0.1$ then repeat Step 2.1.

Step 2.2: $A_i = (x_1, x_2)$; $w_i = 9 \cdot Random() + 1$.

Step 2.3: $r = Random()$; If $r < 0.5$ then $\mathcal{S}_< = \mathcal{S}_< \cup \{i\}$ else $\mathcal{S}_> = \mathcal{S}_> \cup \{i\}$.

Step 2.4 Continue Step 2.

Step 2: $X = (4 \cdot Random(), 4 \cdot Random())$; Implement modified Algorithm 3.3 to calculate $X_r = \mathcal{C}(X)$; If $X_r \in \mathcal{R}_f$ is not obtained (i.e. if the feasible region is empty), remove one randomly chosen element from $\mathcal{S}_<$ and repeat Step 2.

Step 3: STOP, return $\{A_i\}$, $\{w_i\}$, $\mathcal{S}_<$, $\mathcal{S}_>$.

Algorithm 3.3 works under the assumption that the feasible region is not empty. If it is empty, its Step 2 turns into an endless loop. To avoid this situation, we add Step 2.3a:

Step 2.3a: If $\min_{X \in \{C_1, C_2, I_1, I_2\}} \mathcal{G}(X) \geqslant G$ then STOP, return no result.

Having generated an example problem, we solve it with Algorithm 3.1.

The objective function of Weber problem is Lipschitzian [16]. To obtain the Lischitzian constant λ, we use the following inequality [12]

$$\begin{aligned} \lambda\|X - X'\| \geqslant |f(X) - f(X')| &= |\sum_{i=1}^{N} w_i\|A_i - X\| - \sum_{i=1}^{N} w_i\|A_i - X'\||\\ &\leqslant \sum_{i=1}^{N} w_i| \|A_i - X\| - \|A_i - X'\||. \end{aligned} \tag{4.1}$$

From the triangle rule,

$$\sum_{i=1}^{N} w_i| \|A_i - X\| - \|A_i - X'\|| \leqslant \sum_{i=1}^{N} w_i\|X' - X\|.$$

Thus,

$$\lambda\|X - X'\| \geqslant \sum_{i=1}^{N} w_i\|X' - X\|.$$

With Lipschitzian constant value

$$\lambda = \sum_{i=1}^{N} w_i,$$

the inequality (4.1) is true.

From Algorithm 4.1, $1 < w_i < 10$. Therefore, $N < \lambda < 10 \cdot N$.

For convex objective function, having performed the grid search (also known as the sampling method, see [8]) on a grid with step s along each coordinate, we can estimate the minimum of the objective function with tolerance $10 \cdot N \cdot s$. Since the coordinates of the demand points belong to the interval $[0, 4]$, estimating n values of each coordinate gives result with tolerance $40 \cdot N/n$. We used the following algorithm.

Algorithm 4.2. Sampling method.

Require: Number of samples of each coordinate n, objective function f, number of demand points N.

Step 1: $f^* = +\infty$; $X'^* = (0, 0)$.

Step 2: For $i \in \{\overline{1, n}\}$ do:

Step 2.1: For $i \in \{\overline{1, n}\}$ do:

Step 2.1.1: $x_1 = 4 \cdot i/n$; $x_2 = 4 \cdot j/n$; $X = (x_1, x_2)$; $f^{**} = f(X)$.

Step 2.1.2: If $f^{**} < f^*$ then $X'^* = X$; $f^* = f^{**}$.

Step 2.1.3: Continue Step 2.1.

Step 2.2: Continue Step 2.

Step 3: STOP, return minimum value f^*, minimum point X'^* and the tolerance $\varepsilon_t = 40 \cdot N/n$.

If X^* is the solution of the problem (1.1) obtained by Algorithm 3.1 and X'^* is the solution obtained by Algorithm 4.2 then

$$|f(X^*) - f(X'^*)| \leqslant 40 \cdot N/n. \tag{4.2}$$

Having generated 30 problems using Algorithm 4.1 with $N = 3, 4, 5, 7, 10, 15, 20$ and solved them with both Algorithm 3.1 and Algorithm 4.2 ($n = 1000$) and proved the inequality (4.2) for each pair of results, we proved Hypothesis 1 and the convergence of Algorithm 3.1 experimentally.

5. Conclusion

The proposed algorithm which realizes the slightly modified Weiszfeld procedure is able to solve the single-facility constrained Weber problems with the connected feasible region bounded by arcs with equal radius. The algorithm can be useful for problems with maximum and minimum distance limits as a simple alternative for the Mixed-Integer Nonlinear Programming procedure. However, the computational complexity of the proposed algorithm is subject to the further research.

REFERENCES

1. M. M. Deza and E. Deza: *Encyclopedya of Distances*, Springer Verlag, Berlin, Heilderberg, 2009.
2. Z. Drezner: *A note on accelerating the weiszfeld procedure*, Location Science, **3(4)** 1995, 275–279.
3. Z. Drezner, K. Klamroth, A. Schobel and G. O. Wesolowsky: *The Weber problem*, in Z. Drezner and H. W. Hamacher (editors), *Facility Location: Applications and Theory*, Springer-Verlag, 1–36
4. Z. Drezner, A. Mehrez and G. O. Wesolowsky: *The facility location problem with limited distances*, Transportation Science **25** 1991, 183–187.
5. Z. Drezner, C. Scott and J.-S. Song: *The central warehouse location problem revisited*, IMA Journal of Managemeng Mathematics **14(4)** 2003, 321–336.
6. R. Z. Farahani and M. Hekmatfar, editors: *Facility Location Concepts, Models, Algorithms and Case Studies*, Springer-Verlag Berlin Heidelberg, 2009.
7. I. F. Fernandes, D. Aloise, D. J. Aloise, P. Hansen and L. Liberti: *On the Weber facility location problem with limited distances and side constraints*, Optimization Letters (22 August 2012), 1–18, published online, doi:10.1007/s11590-012-0538-9.
8. D. E. Finkel and C. T. Kelley: *Convergence Analysis of Sampling Methods for Perturbed Lipschitz Functions*, Pacific Journal of Optimization **5** 2009, 339–350.
9. P. Hansen, D. Peeters and J.-F. Thisse: *Constrained location and the Weber-Rawls problem*, North-Holland Mathematics Studies **59** 1981, 147–166.
10. P. Hansen, D. Peeters and J.-F. Thisse: *An algorithm for a constrained Weber problem*, Management Science **28(11)** 1982, 1285–1295.
11. H. Idrissi, O. Lefebvre and C. Michelot: *A primal-dual algorithm for a constrained Fermat-Weber problem involving mixed norms*, Revue francaise d'automatique, d'informatique et de recherche operationnelle. Recherche Operationnelle **22(4)** 1988, 313–330.
12. D. R. Jones, C. D. Perttunen and B. E. Stuckman: *Lipschitzian optimization without the Lipschitz constant*, Journal of Optimization Theory and Application **79(1)** (1993), 157–181.
13. L. A. Kazakovtsev: *Adaptation of the probability changing method for Weber problem with an arbitrary metric*, Facta Universitatis, (Niš) Ser. Math. Inform. **27(2)** 2012, 289–254.
14. L. A. Kazakovtsev: *Wireless coverage optimization based on data provided by built-in measurement tools*, WASJ **Vol.22**, Special Volume on Techniques and Technologies 2013, 8–15.
15. H. Minkowski: *Gesammelte Abhandlungen*, zweiter Band, AMS Chelsea Publishing, 2005.
16. I. A. Osinuga and O. M. Bamigbola: *On the minimum norm solution to Weber problem*, SAMSA Conference Proceedings, Windhoek 2007, 27–30.
17. O. R. Oniyide and I. A. Osinuga: *On the existence of best sample in simple random sampling*, J. Math. Assoc. Nigeria, **33(2B)** 2006, 290–294.
18. I. A. Osinuga and O. R. Oniyide: *An overview of norm approximation applications*, Far East J. Math. Sciences **43(2)** 2010, 189–202.
19. P. S. Stanimirović and M. Ćirić: *Discrete location problem on arbitrary surface in R3*, Facta Universitatis, (Niš) Ser. Math. Inform. **25** 2010, 47–56.
20. P. S. Staminirović, M. Ćirić, L. A. Kazakovtsev and I. A. Osinuga: *Single-facility Weber location problem based on the Lift metric*, Facta Universitatis, (Niš) Ser. Math. Inform. **27(2)** 2012, 175–190.

21. Y. Vardi and C.-H. Zhang: *A modified Weiszfeld algorithm for the Fermat-Weber location problem*, Mathematical Programming **90(3)** 2001, 559–566.

22. A. Weber: *Ueber den Standort der Industrien, Erster Teil: Reine Theorie des Standortes*, Tuebingen: Mohr, 1909.

23. E. Weiszfeld: *Sur le point sur lequel la somme des distances de n points donnes est minimum*, Tohoku Mathematical Journal **43(1)** 1937, 335–386.

24. G. Wesolowsky: *The Weber problem: History and perspectives*, Location Science **1** 1993, 5–23.

25. G. G. Zabudski: *A minimax planar location problem with forbidden zones: its solution algorithm*, Automation and Remote Control **65 (2)** 2004, 241–247. Translated from Avtomatika i Telemekhanika, **2** 2004, 93–100.

Lev A. Kazakovtsev
Siberian State Aerospace University named after academician M. F. Reshetnev
Department of Information Technologies
prosp.Krasnoyarskiy Rabochiy, 31
660014 Krasnoyarsk, Russian Federation
levk@bk.ru