ABSTRACT. A unified approach to multisource location problems on a sphere is presented. Euclidean, squared Euclidean and the great circle distances are considered. An algorithm is formulated and its convergence properties are investigated.

KEY WORDS AND PHRASES. Facility location, optimum location, nonlinear programming, applications of mathematical programming.

1980 AMS SUBJECT CLASSIFICATION CODES. 90C30, 90C50.

1. INTRODUCTION.

The multifacility location problem can be defined as finding the location of new facilities, in a specified space, with respect to several existing facilities so that the sum of weighted distances between the locations of new facilities, and new and existing facilities is minimized. Mathematically, problem can be stated as:

\[
\text{minimize } f(X_1, X_2, \ldots, X_n) = \sum_{1 \leq j < k \leq n} v_{jk} d(X_j, X_k) + \sum_{i=1}^{n} \sum_{j=1}^{m} w_{ij} d(X_j, P_i) \tag{1.1}
\]

where \( v_{jk} \) is a positive weight associated with new facilities \( j \) and \( k \), \( w_{ij} \) is a positive weight associated with the existing facility \( i \) and new facility \( j \), for all \( i \) and \( j \), \( X_j = (x_{j1}, x_{j2}, \ldots, x_{jn})' \) is the location of the new facility \( j \) to be determined, \( P_i = (p_{i1}, p_{i2}, \ldots, p_{in})' \) is the location of the existing facility \( i \), and \( d(\cdot) \) is the distance between two points by a given norm.

Problem (1.1) has been treated extensively in the literature. The distance functions most commonly encountered are the Euclidean and rectilinear distances. Euclidean distance problem is discussed in [1-5]. Problem (1.1) involving rectilinear distance is studied in [1,2], [6-9].

When modeling and analyzing the problem of finding the best geographical locations for a number of new facilities, an analyst has to make several assumptions. One of these assumptions is concerned with the size of the area covering the destinations (or the demand points). If the area covering the demand points is
sufficiently small, then this part of the earth's surface can be approximated by a plane. And the problem can be analyzed and solved by using the well known techniques of location theory. However, the earth's surface can be described, with considerable exactness, as a sphere. And plane is a good approximation only for relatively small areas. When the destination points are widely separated, the area covering these points can no longer be approximated by a plane and (1.1) is no longer a suitable model. In this case the destination points and the new facilities are restricted to be on the sphere \(||x||^2 = R, x \in \mathbb{R}^3,\) where \(R\) is the radius of the earth's surface. Because of the constraint, solution space becomes nonconvex. And the shortest travel distance between two points is not the Euclidean distance but the geodesic (or great circle) distance. Problems concerning location of international headquarters, distribution/marketing centers, detection station placement, placement of radio transmitters for long range communication may fall into this category.

In recent years, many papers have dealt with the large region location problems. Location problems involving points on a sphere are discussed in [10-27]. A complete review of the state of art in spherical location can be found in Wesolowsky [25]. The only attempts to solve multisource location problems on a sphere, of which we are aware are due to Dhar and Rao [21-22]. They considered the multisource location problem on a sphere involving geodesic distance, and developed a normalized gradient method.

In this paper, we present a unified approach for the solution of multisource location problems on a sphere. Three distance measures are considered; Euclidean, squared Euclidean, and the great circle distance. An algorithm analogous to the Weiszfeld algorithm [28] for the classical Weber problem is formulated. When the distance measure is Euclidean or the squared Euclidean distance, it is shown that the proposed algorithm always converges to a local minimum. Computational results are presented.

2. PROBLEM FORMULATION.

Let \(X,\) and \(Y\) be two points on the surface of a sphere (without loss of generality, we assume a sphere with unit radius, i.e. \(R = 1\)). We consider the following metrics:

1. Squared Euclidean distance : 
\[
d_1(X,Y) = ||X-Y||^2 
\]  
(2.1)

2. Euclidean distance : 
\[
d_2(X,Y) = ||X-Y|| 
\]  
(2.2)

3. Geodesic distance : 
\[
d_3(X,Y) = 2 \arcsin \left( \frac{||X-Y||}{r} \right) 
\]  
(2.3)

Note that the metrics are all of the form \(d(X,Y) = h(||X-Y||)\). Using this general form the multifacility location problem on a sphere may be stated as follows:

\[
\min f(X_1, X_2, \ldots, X_n) = \sum_{1 \leq j < k \leq n} \nu_{jk} h(||X_j - X_k||) + \sum_{j=1}^{n} \sum_{i=1}^{m} \nu_{ji} h(||X_j - P_i||) 
\]

subject to
\[
||X_j||^2 = 1, \quad j=1,2,\ldots,n 
\]
\[
X_j \in \mathbb{R}^3, \quad j=1,2,\ldots,n 
\]  
(2.4)

The objective function is bounded both from below and above and there exists a maximum as well as a minimum (Aykin and Babu [26]). Furthermore the solution space is not convex resulting in a non-convex programming problem.
3. THE ALGORITHM.

In order to derive an algorithm we first derive necessary conditions for a set of points to be minimum. Using Lagrange multipliers $\mu_j$, $j=1,2,\ldots,n$, the following Lagrange function is obtained:

$$L(x_1,x_2,\ldots,x_n) = \sum_{1 \leq j \leq k \leq n} v_{jk}h(||x_j - x_k||) + \sum_{j=1}^{n} \sum_{i=1}^{m} w_{ij}h(||x_j - p_{ij}||) + \sum_{j=1}^{n} \mu_j(||x_j||^2 - 1)$$  \hspace{1cm} (3.1)

which leads to the necessary conditions

$$\nabla_j L = \sum_{k=1}^{n} v_{jk} h'(||x_j - x_k||) \frac{x_k - x_j}{||x_j - x_k||} + \sum_{i=1}^{m} w_{ij} h'(||x_j - p_{ij}||) \frac{p_{ij} - x_j}{||x_j - p_{ij}||}$$

$$+ 2\mu_j x_j = 0, \quad j=1,2,\ldots,n$$  \hspace{1cm} (3.2)

$$\frac{\partial L}{\partial \mu_j} = ||x_j||^2 - 1 = 0, \quad j=1,2,\ldots,n$$  \hspace{1cm} (3.3)

where $\nabla_j L$ is the vector of partial derivatives of $L(x_1,x_2,\ldots,x_n)$ with respect to $x_j$, $j=1,2,\ldots,n$ (coordinates of source $j$). Solution of (3.2) and (3.3) yields

$$\left\{ \begin{array}{c}
\sum_{k=1}^{n} v_{jk} \frac{h'(||x_j - x_k||)}{||x_j - x_k||} + \sum_{i=1}^{m} w_{ij} \frac{h'(||x_j - p_{ij}||)}{||x_j - p_{ij}||} + 2\mu_j \right\} ||x_j||^2 = 0, \quad j=1,2,\ldots,n
\end{array} \right.$$  \hspace{1cm} (3.4)

so that

$$x_j = T_j(x_1,x_2,\ldots,x_n) = \frac{\sum_{k=1}^{n} v_{jk} \frac{h'(||x_j - x_k||)}{||x_j - x_k||} x_k + \sum_{i=1}^{m} w_{ij} \frac{h'(||x_j - p_{ij}||)}{||x_j - p_{ij}||} p_{ij}}{\sum_{k=1}^{n} v_{jk} \frac{h'(||x_j - x_k||)}{||x_j - x_k||} + \sum_{i=1}^{m} w_{ij} \frac{h'(||x_j - p_{ij}||)}{||x_j - p_{ij}||}}$$  \hspace{1cm} (3.5)

By choosing positive sign in (3.5) and by taking $T_j(x_1,x_2,\ldots,x_n)$, $j=1,2,\ldots,n$, as iteration functions an algorithm, analogous to Weiszfeld [28] algorithm, is defined as follows:
Step 1. Designate the starting points by $X_j^t$, $j=1,2,...,n$. Set $t=1$, and $\nu$.

Step 2. Calculate $X_j^{t+1}$, $j=1,2,...,n$ by (3.5) using $X_1^t$, $X_2^t$, ..., $X_j-1^t$, $X_j^t$, $X_{j+1}^t$, ..., $X_n^t$.

$$X_j^t = T_j(X_1^t, X_2^t, ..., X_{j-1}^t, X_j^t, ..., X_n^t)$$

Step 3. If

$$\max \left\{ \frac{||X_j^{t+1} - X_j^t||}{||X_j^t||}, j=1,2,...,n \right\} \leq \nu$$

stop. Otherwise, set $t=t+1$ and go to step 2.

In the algorithm given above, each source location is updated during the iteration in the sense that $X_j^{t+1}$ is a function of $X_1^t$, $X_2^t$, ..., $X_{j-1}^t$, $X_j^t$, $X_j^t$, ..., $X_n^t$ for all $j$. This approach is known as Gauss-Seidel procedure. An alternative, and potentially less efficient procedure, is to compute $X_j^{t+1}$ as a function of $X_1^t$, $X_2^t$, ..., $X_{j-1}^t$, $X_j^t$, $X_j^t$, ..., $X_n^t$.

$$X_j^{t+1} = T_j(X_1^t, X_2^t, ..., X_n^t).$$

Note that if

$$\sum_{k=1}^{n} v_{jk} \frac{h'(||X_j - X_k||)}{||X_j - X_k||} X_k + \sum_{i=1}^{m} v_{ji} \frac{h'(||X_j - P_i||)}{||X_j - P_i||} P_i = 0$$

for some $j$, then the iterative functions given in (3.5) will be undefined. We call such a set of vectors an irregular solution.

The algorithm defined by the iteration functions $T_j$, $j=1,2,...,n$, gives a gradient projection method with precalculated step size. In this respect this algorithm is analogous to the Weiszfeld scheme [28]. Let $\nabla_j f$ be the vector of partial derivatives of $f$ with respect to $x_j, y_j, z_j$ ($x, y, z$ coordinates of $j$th source), then

$$\nabla_j f = \sum_{k=1}^{n} v_{jk} \frac{h'(||X_j - X_k||)}{||X_j - X_k||} (X_j - X_k) + \sum_{i=1}^{m} v_{ji} \frac{h'(||X_j - P_i||)}{||X_j - P_i||} (X_j - P_i)$$

so that

$$\sum_{k=1}^{n} v_{jk} \frac{h'(||X_j - X_k||)}{||X_j - X_k||} X_k + \sum_{i=1}^{m} v_{ji} \frac{h'(||X_j - P_i||)}{||X_j - P_i||} P_i$$

for $j=1,2,...,n$ (3.6)
MULTIFACILITY LOCATION PROBLEMS ON A SPHERE

\[ h(x) = \frac{h'(||x_j - x_k||)}{||x_j - x_k||} \sum_{i=1}^{m} v_{ji} \frac{h'(||x_j - p_i||)}{||x_j - p_i||} ||x_j - P_i|| \]

Therefore

\[ T_j(x_1, x_2, \ldots, x_n) = \frac{x_j - s_j}{||x_j - s_j||} \sum_{j=1}^{n} v_{jkl} \frac{h'(||x_j - x_k||)}{||x_j - x_k||} + \sum_{i=1}^{m} v_{ji} \frac{h'(||x_j - p_i||)}{||x_j - p_i||} \]

where

\[ s_j(x_1, x_2, \ldots, x_n) = \frac{1}{\sum_{k=1}^{n} v_{jkl} \frac{h'(||x_j - x_k||)}{||x_j - x_k||} + \sum_{i=1}^{m} v_{ji} \frac{h'(||x_j - p_i||)}{||x_j - p_i||}} \]

1) Squared Euclidean Distance

In this case \( h(U) = U^2 \) and \( h'(U) = 2U \). Then (3.5) becomes

\[ x_j = T_j(x_1, x_2, \ldots, x_n) = \frac{\sum_{k=1}^{n} v_{jkl} x_k + \sum_{i=1}^{m} v_{ji} p_i}{\sum_{k=1}^{n} v_{jkl} + \sum_{i=1}^{m} v_{ji}} \]

One useful property of the algorithm proposed is that the objective function value decreases at each iteration. This is proved in the following theorem.

**THEOREM 1.** Suppose \( X = \{x_1, x_2, \ldots, x_n\} \) is a regular solution. Let \( T(X) = \{T_1(X), T_2(X), \ldots, T_n(X)\} \) be the mapping given in (3.10) then the mapping \( T(X) \) is a descent mapping; that is \( f(T(X)) \leq f(X) \) and \( f(T(X)) = f(X) \) if \( X = T(X) \).

**PROOF.** Let \( X = \{x^t_1, x^t_2, \ldots, x^t_n\} \) be the set of the current source locations at the end of \( t \)th iteration. Value of the objective function at \( X^t \) is \( f(x^t_1, x^t_2, \ldots, x^t_n) \).

Let \( x^t(X) = T_1(x^t_1, x^t_2, \ldots, x^t_n) \). Now consider another problem, which is a single facility location problem with the squared Euclidean distance, and with the existing facilities \( x^t_2, \ldots, x^t_n, p_1, p_2, \ldots, p_m \) (i.e., sources \( x_2, \ldots, x_n \) are "fixed" at their current locations).

\[ \text{minimize } f_1(x_1) = \sum_{j=2}^{n} v_{1j} ||x_1 - x_j||^2 + \sum_{i=1}^{m} v_{1i} ||x_1 - p_i||^2 + g_1(x_2, \ldots, x_n) \]

subject to

\[ ||x_j||^2 = 1, \]

\[ x_1 \in \mathbb{E}_3 \]
where

$$g_1(x_2^t,\ldots,x_n^t) = \sum_{2 \leq j < k \leq n} w_{jk} |x_j^t - x_k^t|^2 + \sum_{j=2}^{n} \sum_{i=1}^{m} w_{ji} |x_j^t - p_i|^2$$  \hspace{1cm} (3.12)$$

Since the sources $x_j$, $j=2,\ldots,n$ are "fixed" at their current locations $g_1(x_2^t,\ldots,x_n^t)$ is is a constant and therefore does not affect the optimum solution of this problem.

Spherical single facility location problem with the squared Euclidean distance was first formulated and shown to have a unique minimum by Katz and Cooper [14]. They formulated this problem as to

minimize $f(x) = \sum_{i=1}^{m} w_i ||x - p_i||^2$

subject to

$$||x||^2 = 1, \hspace{1cm} x \in E_3$$  \hspace{1cm} (3.13)$$

and showed that the unique optimizing location is

$$x = \frac{\sum_{i=1}^{m} w_i p_i}{\sum_{i=1}^{m} w_i}$$  \hspace{1cm} (3.14)$$

Hence the minimizing location for the problem (3.11) is given by

$$x_1^{t+1} = \frac{\sum_{j=2}^{n} v_{1j} x_j^t + \sum_{i=1}^{m} w_{1i} p_i}{\sum_{j=2}^{n} v_{1j} x_j^t + \sum_{i=1}^{m} w_{1i} p_i}$$  \hspace{1cm} (3.15)$$

Note that (3.15) is equivalent to (3.10) for $j=1$. Since $x_1^{t+1}$ is the minimizing solution to the problem (3.11) if $x_1^{t+1} \neq x_1^t$, then $f_1(x_1^{t+1}) < f_1(x_1^t)$. But $f_1(x_1) = f(x_1^t,x_2^t,\ldots,x_n^t)$. Thus we obtain the following inequality:

$$f(x_1^t,x_2^t,\ldots,x_n^t) < f(x_1^{t+1},x_2^t,\ldots,x_n^t)$$  \hspace{1cm} (3.16)$$

And $f(x_1^t,x_2^t,\ldots,x_n^t) = f(x_1^t,x_2^t,\ldots,x_n^t)$ if and only if $x_1^{t+1} = x_1^t$, or

$$x_1 = t_1(x_1^t,x_2^t,\ldots,x_n^t).$$

Now let $x_2^{t+1} = t_2(x_1^{t+1},x_2^t,\ldots,x_n^t)$. By using the same analogy, we formulate another problem which is again a single facility location problem with the squared Euclidean distance and the existing facilities $x_1^t,x_2^t,\ldots,x_n^t,p_1^t,\ldots,p_m^t$, i.e. now the sources $x_1^t$,
and $X_j$, $j=3,\ldots,n$, are "fixed" at their current locations.

\[
\text{minimize } f_2(x_2) = v_{21} \|x_2-x_1^{t+1}\|^2 + \sum_{j=3}^{n} v_{2j} \|x_2-x_j\|^2 \\
+ \sum_{i=1}^{m} w_{2i} \|x_2-p_i\|^2 + \alpha_2(x_1^{t+1},x_3^t,\ldots,x_n^t)
\]

subject to

\[
\|x_2\|^2 = 1, \quad X_2 \in \mathbb{E}_3
\]

where

\[
\alpha_2(x_1^{t+1},x_3^t,\ldots,x_n^t) = \sum_{3 \leq j < k \leq n} v_{jk} \|x_j^t - x_k^t\|^2 + \sum_{j=3}^{n} v_{1j} \|x_1^{t+1} - x_j^t\|^2 \\
+ \sum_{j=3}^{n} \sum_{i=1}^{m} v_{ji} \|x_j^t - p_i\|^2 \\
+ \sum_{i=1}^{m} w_{1i} \|x_1^{t+1} - p_i\|^2
\]

is a constant. Again this problem has a unique solution.

\[
x_2^{t+1} = \frac{\sum_{j=3}^{n} v_{2j} x_j^t + v_{12} x_1^{t+1} + \sum_{i=1}^{m} w_{2i} p_i}{\|\sum_{j=3}^{n} v_{2j} x_j^t + v_{12} x_1^{t+1} + \sum_{i=1}^{m} w_{2i} p_i\|}
\]

This is equivalent to (3.10) for $j=2$. Since $x_2^{t+1}$ is the minimizing location for the problem (3.17), we have the following inequality:

\[
f_2(x_2^{t+1}) < f_2(x_2^t)
\]

provided that $x_2^{t+1} \neq x_2^t$. But $f_2(x_2^t) = f(x_1^{t+1},x_2^t,x_3^t,\ldots,x_n^t)$. Thus

\[
f(x_1^{t+1},x_2^{t+1},x_3^t,\ldots,x_n^t) < f(x_1^{t+1},x_2^t,x_3^t,\ldots,x_n^t)
\]

By combining the inequalities (3.16) and (3.21), we obtain
\[ f(x_{t+1}, x_{t+1}, x_3, \ldots, x_n) < f(x_t, x_t, x_3, \ldots, x_n) \] (3.22)

provided that \( x_{t+1} \neq x_1 \) and/or \( x_{t+1} \neq x_2 \). By repeating the same steps for \( x_j, j=3, \ldots, n, \)
we get
\[ f(x_{t+1}, x_{t+1}, \ldots, x_n) < f(x_t, x_t, \ldots, x_n) \] (3.23)

whereas
\[ f(x_{t+1}, x_{t+1}, \ldots, x_n) = f(x_t, x_t, \ldots, x_n) \] if and only if
\[ \{x_1^{t+1}, x_2^{t+1}, \ldots, x_n^{t+1}\} = \{x_1^t, x_2^t, \ldots, x_n^t\}, \text{ or } X = T(X). \]

Thus when the metric is the squared Euclidean distance the objective function value decreases at each iteration, and the algorithm either converges to a minimum or stops at an irregular solution.

COROLLARY 1. Suppose \( \{x_1, x_2, \ldots, x_n\} \) is a regular solution. If \( x_j, j=1,2, \ldots, n, \)
is the minimizing location with respect to both the other new facilities and the destination points then the solution is a minimizing solution to the problem.

PROOF. Proof follows from Theorem 1.

Note that the method used above can be easily extended to other location problems. In order to prove that the algorithm designed for the solution of a multifacility location problem is a descent algorithm, it is sufficient to prove the descent mapping property of the same algorithm for the single facility version of the same problem.

2) Euclidean Distance

In this case \( h(U)=U \) and \( h'(U)=1. \) (3.5) then becomes
\[ X_j = T_j(x_1, x_2, \ldots, x_n) = \frac{\sum_{k=1}^{n} \sum_{i=1}^{m} v_{jk}x_k + \sum_{i=1}^{m} v_{jip} i}{\sum_{k=1}^{n} ||x_j-x_k|| + \sum_{i=1}^{m} ||x_j-x_i||} \] (3.24)

The iteration functions given in (3.24) are not defined at all the points in the solution space. In particular, if either new facilities \( j \) and \( k \) have the same location or new facility \( j \) and existing facility \( i \) have the same location then the objective function \( f(x_1, x_2, \ldots, x_n) \) is not differentiable and, therefore, the iteration functions (3.24) are not defined. In order to resolve this difficulty, the Hyperbolic Approximation Procedure of Eyster et al [2] is used as follows:
\[ d(X,Y) = ||X - Y||^2 + \xi \] (3.25)

where \( \xi \) is a small positive constant.

We now prove that the iteration functions (3.24) define a descent mapping in the following theorem.

THOREM 2. Suppose \( X = \{x_1, x_2, \ldots, x_n\} \) is a regular solution, \( x_j \neq x_k \) for \( j \neq k, \) and \( x_j \neq x_i \) for all \( j, i. \) Let \( T(X) = \{T_1(X), T_2(X), \ldots, T_n(X)\} \) be the mapping given in (3.24), then the mapping \( T(X) \) is a descent mapping.
MULTIFACILITY LOCATION PROBLEMS ON A SPHERE

I, X₂, ..., Xₙ be the set of the current source locations, and Xₖ₊₁ = Tₖ(X₁, X₂, ..., Xₖ₊₁), j = 1, 2, ..., n. Now consider the following problem.

\[
\begin{align*}
\text{minimize} & \quad f_j(X_j) = \sum_{k=1}^{j-1} v_{jk} |X_j - X_k| + \sum_{k=j+1}^{n} v_{jk} |X_j - X_k| + \sum_{l=1}^{m} v_{jl} |X_k - P_l| \\
& \quad + g_j(X_1, X_2, ..., X_{j-1}, X_{j+1}, ..., X_n)
\end{align*}
\]

subject to

\[
\sum_{k=1}^{n} |X_k| = 1,
\]

\[
X_j \in E_3
\]

where

\[
g_j(X_1, X_2, ..., X_{j-1}, X_{j+1}, ..., X_n) = \sum_{1 \leq q < j \leq 1} v_{qk} |X_{qk} - X_{j-1}|
\]

is a constant. Note that this problem is a single facility location problem with the existing facilities X₁, X₂, ..., Xₙ, P₁, P₂, ..., Pₘ, for j = 1, 2, ..., n.

Problem (3.13) involving Euclidean distance has been studied by Katz and Cooper [14]. They proposed a descent algorithm with the following iterative function.

\[
X = \frac{\sum_{i=1}^{m} w_i P_i}{\sum_{i=1}^{m} w_i P_i}
\]

This iterative function can be used iteratively to solve (3.26) as follows.

\[
x_{j+1} = \frac{\sum_{k=1}^{j-1} v_{jk} x_{j+1}}{\sum_{k=1}^{j-1} |X_{j+1} - X_k|} + \sum_{k=j+1}^{n} v_{jk} x_k + \sum_{i=1}^{m} \frac{w_{ji} P_i}{|X_{j+1} - P_i|}
\]

Since (3.28) defines a descent mapping, we have
By repeating the same steps for $X_j$, $j=1,2,...,n$, we get

$$f(X_1, X_2, ..., X_n) \leq f(x_1^t, x_2^t, ..., x_n^t)$$

whereas $f(X_1^t, X_2^t, ..., X_n^t) = f(x_1^t, x_2^t, ..., x_n^t)$ if $x_{j+1}^t = \{x_1^t, x_2^t, ..., x_n^t\}$,

i.e. $X = T(X)$.

Thus when the metric is the Euclidean distance, the objective function value decreases at each iteration, and the algorithm either converges to a local minimum, or stops at an irregular solution, or when $X_j = X_k$, $j \neq k$, or $X_j = P_1$ for any $j,1$.

3) Geodesic Distance

In this case $h(U) = 2 \arcsin (U/2)$, $U = \|X-Y\| = \sqrt{2(1-XY)}$, and

$$h'(\|X-Y\|) = \frac{1}{\sqrt{(1-XY)^2}}$$

(3.5) then becomes

$$X_j = T_j(X_1, X_2, ..., X_n) = \frac{\sum_{k=1, k \neq}^{n} w_{jk}^X X_k}{\sum_{k=1, k \neq}^{n} \frac{w_{jk}^X}{1 - (X_j X_k)^2}^{1/2}} + \frac{\sum_{i=1}^{m} w_{ij}^P P_i}{\sum_{i=1}^{m} \frac{w_{ij}^P}{1 - (X_j P_i)^2}^{1/2}}$$

An iterative scheme for determining minimizing source locations $X_j$, $j=1,2,...,n$, is now defined by taking (3.33) as iterative functions and using them in the algorithm outlined before.

Note that the iterative functions (3.33), and the partial derivatives of the objective function $f(X_1, X_2, ..., X_n)$ are not defined when $X_j = X_k$ for $j \neq k$, or $X_j = P_i$ for any $j,1$, or when two new facilities $j$ and $k$, or new facility $j$ and existing facility $i$ have the coordinates of a point and its antipode ($X$ and $Y$ on a unit sphere are antipodal points if $d_3(X,Y) = \pi$). In order to resolve this difficulty, we used the Hyperbolic Approximation Procedure of Eyster et al [2].

We have not succeeded in proving that the algorithm with (3.33) as the iteration functions is a descent algorithm. However, above in Theorems 1, and 2, we proved that the algorithms with (3.10) and (3.24) as the iteration functions are descent algorithms. In an analogous manner, one may prove that the algorithm proposed is a descent algorithm.
4. AN EXAMPLE PROBLEM.

This problem is concerned with the location of three distribution centers. The products are to be distributed to 10 European and Asian cities by air. The names and the location coordinates of the cities, and the weights are shown in Tables 1, and 2.

We solved this problem using the proposed algorithm with $\xi = 10^{-12}$ and $\nu = 0.00001$ starting from $X_1=(0.,0.,1.)'$, $X_2=(0.,0.,1.)'$, and $X_3=(0.,0.,1.)'$. The results are presented in Table 3. Additionally this problem was run 10 times with random starting points and the algorithm always converged to a unique solution. Although the difference between the optimal objective function values, calculated using geodesic distances, is small, the distance between the optimum source locations is significant.

<table>
<thead>
<tr>
<th>i</th>
<th>City</th>
<th>Latitude</th>
<th>Longitude</th>
<th>$v_{j1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Berlin</td>
<td>52.517</td>
<td>13.417</td>
<td>0.52 0.78 0.10</td>
</tr>
<tr>
<td>2</td>
<td>New Delhi</td>
<td>28.617</td>
<td>77.217</td>
<td>0.09 0.23 0.14</td>
</tr>
<tr>
<td>3</td>
<td>London</td>
<td>51.513</td>
<td>0.097</td>
<td>0.83 0.97 0.53</td>
</tr>
<tr>
<td>4</td>
<td>Moskow</td>
<td>55.750</td>
<td>37.567</td>
<td>0.12 0.38 0.37</td>
</tr>
<tr>
<td>5</td>
<td>Paris</td>
<td>48.833</td>
<td>2.333</td>
<td>0.97 0.47 0.47</td>
</tr>
<tr>
<td>6</td>
<td>Peking</td>
<td>39.600</td>
<td>116.400</td>
<td>0.69 0.71 0.26</td>
</tr>
<tr>
<td>7</td>
<td>Seoul</td>
<td>37.567</td>
<td>127.000</td>
<td>0.17 0.71 0.77</td>
</tr>
<tr>
<td>8</td>
<td>Tehran</td>
<td>35.667</td>
<td>51.417</td>
<td>0.39 0.85 0.27</td>
</tr>
<tr>
<td>9</td>
<td>Tokyo</td>
<td>35.683</td>
<td>139.750</td>
<td>0.57 0.69 0.58</td>
</tr>
<tr>
<td>10</td>
<td>Victoria</td>
<td>22.333</td>
<td>114.167</td>
<td>0.32 0.49 0.88</td>
</tr>
</tbody>
</table>

**TABLE 1. Weights and Locations of Existing Facilities**

<table>
<thead>
<tr>
<th>i</th>
<th>City</th>
<th>Latitude</th>
<th>Longitude</th>
<th>$v_{jk}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>j = 1 2 3</td>
</tr>
<tr>
<td>1</td>
<td>Berlin</td>
<td>52.517</td>
<td>13.417</td>
<td>0.0 0.15 0.25</td>
</tr>
<tr>
<td>2</td>
<td>New Delhi</td>
<td>28.617</td>
<td>77.217</td>
<td>0.15 0.0 0.25</td>
</tr>
<tr>
<td>3</td>
<td>London</td>
<td>51.513</td>
<td>0.097</td>
<td>0.25 0.25 0.0</td>
</tr>
</tbody>
</table>

**TABLE 2. Weights, $v_{jk}$**

<table>
<thead>
<tr>
<th>Measure</th>
<th>Stops at (Latitude, Longitude)</th>
<th>Obj.Fn.Val.</th>
<th>No. of Iter.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Squared</td>
<td>(59.042,62.591), (55.580,73.663), (52.207,90.548)</td>
<td>571.434</td>
<td>5</td>
</tr>
<tr>
<td>Euclidean</td>
<td>(53.017,13.911), (54.736,52.394), (40.655,114.890)</td>
<td>569.608</td>
<td>102</td>
</tr>
<tr>
<td>Geodesic</td>
<td>(56.745,37.356), (54.521,59.743), (45.620,104.939)</td>
<td>565.164</td>
<td>61</td>
</tr>
</tbody>
</table>

**TABLE 3. Computational Results**
REFERENCES


