FACTA UNIVERSITATIS (NIŠ)
Ser. Math. Inform. Vol. 27 No 2 (2012) 239–254

ADAPTATION OF THE PROBABILITY CHANGING METHOD FOR WEBER PROBLEM WITH AN ARBITRARY METRIC

Lev A. Kazakovtsev

Abstract. Fermat-Weber problem in its simple form (unconstrained, single facility, Euclidean metric) is well investigated. Lot of algorithms are also developed for more complicated cases. But the generalized multi-facility problem with barriers, restricted zones and an arbitrary metric has no well-known algorithm for its solving. In this paper, we consider the planar multi-facility Weber problem with restricted zones and non-Euclidean distances, propose an algorithm based on the probability changing method and prove its efficiency for approximate solving of the Weber problem by replacing the continuous coordinate values with discrete ones. An example for a problem with discrete coordinate system 200x400 is given. Version of the algorithm for multiprocessor systems is proposed.

1. Introduction

Location problems are a special class of optimization problems [38, 12]. We determine a single-facility Weber problem as a problem of searching for such a point X that the sum of weighted distances from this point X to some existing points $A_1, A_2, \ldots, A_N$ is minimum [6, 31].

$$F(X) = \sum_{i=1}^{N} w_i L(X, A_i) \to min. \tag{1.1}$$

Here, w_i is a weight of the ith point, $L(A, B)$ is the distance between the points A and B. In the most common Euclidean metrics l_2,

$$L(A, B) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}. \tag{1.2}$$

Here, a_1 and a_2 are the coordinates of the point A and b_1 and b_2 are the coordinates of the point B. But other types of distances have been exploited in the facility

Received June 20, 2012.
2010 *Mathematics Subject Classification.* Primary 90B85; Secondary 90C27, 90B80

location problem. A review of exploited metrics is presented in [12]. Distance functions based on altered norms are investigated in [25, 26]. Problems with weighted one-infinity norms are solved in [37]. Asymptotic distances [18] and weighted sums of order p [4], [36] are also implemented in location problems. An original research on Weber problem in discrete coordinates is given in [24].

Also, the rectangle [25] (or Manhattan) metrics l_1 is well investigated. Here,

$$L(A,B) = |a_1 - b_1| + |a_2 - b_2|. \tag{1.3}$$

Manhattan metric can be used for fast approximate solution instead of Euclidean metric.

Weber problem (sometimes called Fermat-Weber problem) is a generalization of a simple problem (Fermat problem) and has series of generalized formulations.

Multi-facility problem (Multi-Weber problem) is a generalization of the single-facility problem [7]:

$$F(X_1, X_2, \ldots, X_M) = \sum_{i=1}^{N} \sum_{j=1}^{M} w_i L(X_j, A_i) \to min. \tag{1.4}$$

In this case, the problem is searching for M additional places for new facilities $X_j, 1 \le j \le M$.

Or, in other case [7], the objective function is defined as (minsum problem):

$$F(X_1, X_2, \ldots, X_M) = \sum_{i=1}^{N} w_i \min_{j, 1 \le j \le M} L(X_j, A_i) \to min. \tag{1.5}$$

Also, the problem can include the restricted areas, barriers etc. In case of restricted zones, the optimization problem, in general, includes constraints:

$$X_j \notin R_Z \tag{1.6}$$

where R_Z is a set of restricted coordinates.

In case of barriers, the distance between 2 points is, in general, non-Euclidean (Fig. 1.1)

Here, the distance between points A and B is the sum of distances d_1 and d_2 (shortest path which does not cross the barrier).

Further generalization of the Fermat-Weber problem is the continuous (regional) Weber problem that deals with finding a median for a continuum of demand points. In particular, we consider versions of the "continuous k-median (Fermat-Weber) problem" where the goal is to select one or more center points that minimize the weighted distance to a set of points in a demand region [13]. If the existing facilities are distributed in some compact area $\Omega \in \mathbb{R}^n$ then the single-facility continuous Weber problem [15] is to find X so that

$$F(X) = \sum_{i=1}^{N} w_i \int_{\Omega} L(X, A_i) d\mu(A_i) \to min. \tag{1.7}$$

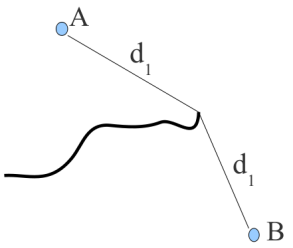

FIG. 1.1: Distance with a barrier

where $\mu(A_i)$ is the expectation of the fact that the ith customer is placed in some point.

If this expectation is equal among some area, i.e., if ith customer is uniformly (equably) distributed in the area then

$$F(X) = \sum_{i=1}^{N} w_i \int\limits_{\Omega_i} L(X, A)\rho_i(A)dA \to min, \tag{1.8}$$

$$\rho_i(A) = \begin{cases} 1, & A \in \Omega_i \\ 0, & A \notin \Omega_i. \end{cases} \tag{1.9}$$

In case of continuous (regional) multi-Weber problem,

$$F(X_1, ..., X_M) = \sum_{i=1}^{N} w_i \int\limits_{\Omega_i} \min_{j,1\le j\le M} L(X_j, A)\rho_i(A)dA \to min. \tag{1.10}$$

In this paper, we propose an algorithm for approximate solving of the problem (1.10) with constraints (1.6) where the distance function $L()$ is an arbitrary monotone function. In the example given this function is the well-known path loss function implemented to calculate the radio-propagation features of the area (media).

2. Related Works, Probability Changing Algorithm

The Weber problem locates medians (facilities) at continuous set of locations in the Euclidean plane (or plane with an arbitrary metrics in generalized case). Hakimi proposed similar problem statement for finding medians on a network or graph [16, 17], and his absolute median problem is similar to Weber's weighted problem. Hakimi defined the absolute median as the point on a graph that minimizes the sum of the weighted distances between that point and the vertices of the graph. Hakimi allowed this point to lie anywhere along the graphs edges, but proved that

an optimal absolute median is always located at a vertex of the graph, thus providing a discrete representation of a continuous problem. In [17] Hakimi generalized the absolute median to find p medians on a graph in order to minimize the sum of the weighted distances. Hakimi again provided a discrete representation of a continuous problem by restricting the search to the vertices. Solutions consisting of p vertices are called p-medians of the graph. Thus, the p-median problem differs from the Weber problem because it is discrete, a consequence of only being allowed to select medians from the candidate set V. It is also defined on a graph or a network, not a plane.

Primary problems in the field of discrete location theory: the p-median problem, the p-center problem, the uncapacitated facility location problem (UFLP) and the quadratic assignment problem (QAP) [29]. These problems decide the location of facilities and allocate demand points to one or multiple facilities (so called location-allocation problems). The p-median problem is well studied in the literature. Lots of the solution methods have been proposed [9]. The p-median problem is one of a larger class of problems known as minisum location-allocation problems. These problems find medians among existing points, which is not the same as finding centers among points, a characteristic of minimax location-allocation problems (the p-center problem is an example, where the goal is to minimize the maximum distance between center(s) and points). But this (p-median) problem is stated as searching among a set of pre-defined feasible points. In case of discrete coordinates, considering all points as feasible, we have p-median problem. But the dimension of such problem is very large in the discretization is fine enough. In general, p-median problem is $\mathcal{NP}$-hard, the polynomial time algorithm is available on trees only [22], [14]. Cabot, Francis and Stary [5], [21] utilized a network flow procedure (an algorithm for p-median problem) to solve the multi-facility location problem with rectilinear distances . In case of the weber problem (except one with Manhattan or similar metric) [35], the sum of the edges weights is not equal to distance in the problem with the discrete coordinate grid.

Drezner and Wesolowsky [11] researched the continuous problem under an arbitrary l_p distance metric.; and, in [37], authors formulated the well-known "block-norm" for the distances involved. In [27] and [28], the authors had shown that the Euclidean and rectilinear cases of the unconstrained Weber promlems are $\mathcal{NP}$-complete.

An unconstrained problem with the mixed coordinate system (discrete and continuous) is considered in [35].

For the generalized Multi-Weber problem with restricted zones and barriers, only some special cases are considered. In [3], authors provided two heuristics for the Multi-Weber Problem with barriers, and reported that their algorithms can attain solutions of reasonably sized multifacility location–allocation problems with barriers, both with regard to computation time and solution quality.

Having transformed our continuous Weber problem into problem with discrete coordinate grid, we have a combinatorial discrete optimization problem.

Most exact solution approaches to the problem of discrete (combinatorial) op-

timization (knapsack problem, the traveling salesman problem etc.) are based on branch-and-bound method (tree search) [1, 2, 20]. Unfortunately, most of such problems are in the complexity class $\mathcal{NP}$-hard and require searching in a tree of the exponential size and even parallelized versions of such algorithms do not allow us to solve some large-scale pseudo-Boolean optimization problems in acceptable time without significant simplification of the initial problem.

The heuristic random search methods do not guarantee any exact solution but they are statistically optimal. I.e. the percent of the problems solved "almost optimal" grows with the increase of the problem dimension [1].

The real large-scale problems have sometimes millions of variables. For example, the problem of assortment planning of the retail trade company [23] may include thousands goods names to be selected which can be shipped from hundreds suppliers and have 3-10 variants of retail price. In general, problems of such kind can be solved only with random search algorithms.

Being initially designed to solve the unconstrained pseudo-Boolean optimization problems and classified sometimes as a variant of the genetic algorithms [10], the probability changing method (MIVER) is a random search method organized by the following common scheme [1, 2].

Algorithm 2.1. Probability changing method

1. $k = 0$, the starting values of the probabilities $P_k = \{p_{k1}, p_{k2}, \ldots, p_{kN}\}$ are assigned where $p_{kj} = P\{x_j = 1\}$. Correct setting of the the starting probabilities is a very significant question for the constrained optimization problems.

2. With probability distributions defined by the vector P_k, we generate a set of the independent random points X_{ki}.

3. The function values in these points are calculated: $F(X_{ki})$.

4. Some function values from the set $F(X_{ki})$ and corresponding points X_{ki} are picked out (for example, point with maximum and minimum values).

5. On the basis of results in item 4, vector P_k is modified.

6. $k = k + 1$, if $k < R$ then go to 2. This stop condition may differ.

7. Otherwise, stop.

To be implemented for the constrained problems, this method has to be modified. The modified version of the variant probability method, offered in [23, 24] allows us to soling large-scale problems with dimensions up to millions of Boolean variables.

3. Problem Statement

Let's consider the problem (1.10) with constraints (1.6) in the case when the coordinate grid is discrete. For most practically important problems, the solution of such approximated (discretized) problem is enough. Moreover, the distance measurement and maps precision are always finite.

The transformation of the continuous coordinates into discrete coordinate grid is shown in Fig. 3.1. The area is divided into N_x columns and N_y rows and the whole area forms a set of cells. In this case, the integral in formula (1.10) of the regional Weber problem is transformed back into a sum (1.5) of a Multi-Weber problem.

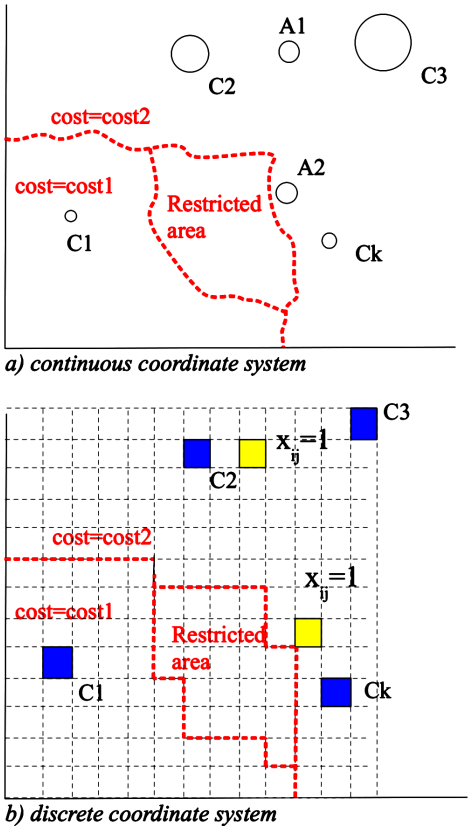

FIG. 3.1: Weber problems with continuous and discrete coordinates

The problem is to select N_F cells where the new facilities will be placed so that

$$F_1(X) = \sum_{k=1}^{N_x} \sum_{l=1}^{N_y} w_{ij} \min_{\substack{1 <= i <= N_x, \\ 1 <= j <= N_y, \\ x_{ij} = 1}} x_{ij} L((i,j),(k,l)) \to min. \tag{3.1}$$

$$X = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1N_y} \\ x_{21} & x_{22} & \dots & x_{2N_y} \\ \dots & \dots & \dots & \dots \\ x_{N_x 1} & x_{N_x 2} & \dots & x_{N_x N_y} \end{pmatrix} \tag{3.2}$$

with constraints

$$x_{ij} = 0 \ \forall (i,j) \in R_z; \tag{3.3}$$

$$\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} x_{ij} <= N_F. \tag{3.4}$$

Here, X is a matrix of Boolean variables, R_z is a set of cells restricted for facility placement, $L()$ is a distance function, in general, arbitrary but monotone. If we have a problem with barriers, this function is calculated as shown in Fig. 1.1. A coefficient w_{ij} is the weight of the cell (i,j), N_F is quantity of facilities to be placed.

Also, the problem may have additional constraints.

As an example of the problem (3.1)–(3.4), we consider a problem of antennas placement (Fig. 3.2).

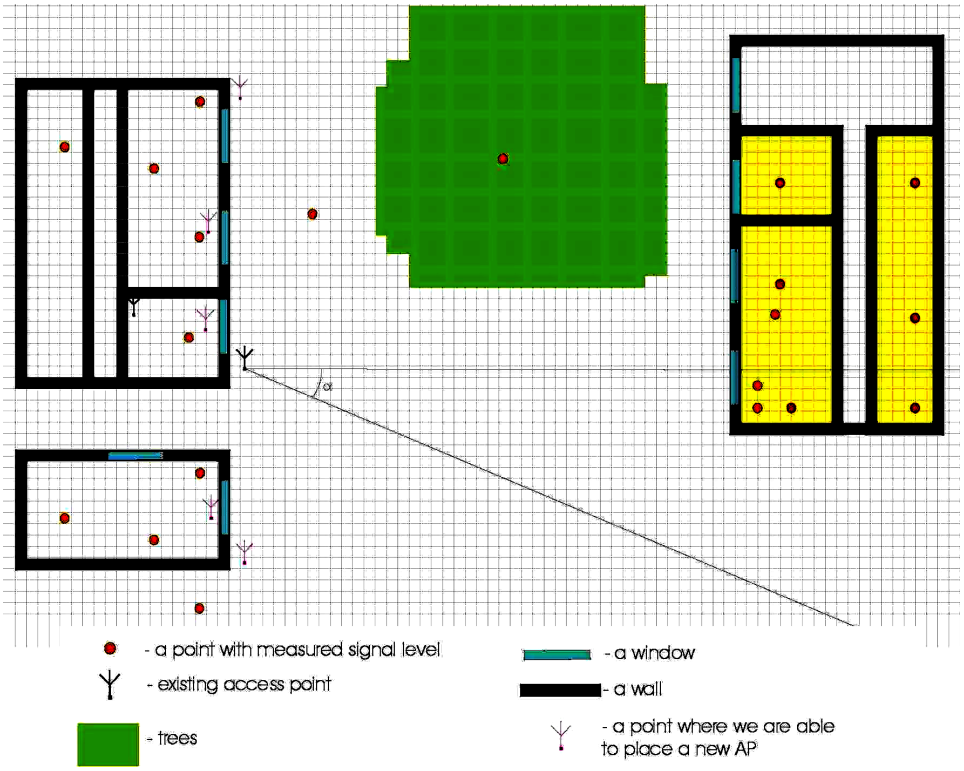

FIG. 3.2: Scheme of the antennas placement problem

Here, we have a map with discrete coordinates, each cell has its weight which is a measurement of its importance to be covered with stable RF signal from any of antennas. The cells can contain different kinds of obstacles (walls, trees etc). As

the distance function, we can use the well-known path loss function [33] calculated as

$$L((i,j)(k,l)) = 20log\|(i,j),(k,l)\| + L_{OBST}((i,j),(k,l)). \tag{3.5}$$

Here, L_{OBST} is the obstacle path loss which is calculated algorithmically as loss at all the obstacles (depending on their material and thickness) along the path between the cells (i,j) and (k,l). The RF absorbing properties of the environment elements are available from the information tables [19] and able to be defined more exactly if the element is situated between the existing transmitter and receiver or in the Fresnel zone. In our distance function, we do not take into consideration the antenna gain since this parameter does not depend on antennas placement.

In continuous coordinates, the objective function is monotone.

So, we have constrained pseudo-Boolean optimization problem (3.1)–(3.5). The total quantity of variables of our problem is $N_x \times N_y$. Thus, even in case of 100×100 coordinate grid, we have a problem with 10000 variables. In case of the large-scale problems, even the calculation of the linear objective function takes significant computational resources. Here, we have a function calculated algorithmically. So, the calculation of the objective function and the constraints is a very large computational problem if it is repeated lot of times.

That is why, the distribution of the computational tasks between the parallel processors or cluster nodes is very important.

In this paper, we do not consider greedy search algorithms (which are deterministic or also randomized [34]) though they are often used to improve the results of the random search methods as the final step of them.

Also, we do not consider the parallel genetic algorithms [32, 30] though some approaches offered for genetic algorithms may be implemented for the random search algorithms parallelization.

4. Serial Version of the Algorithm and Its Parallelization with OpenMP

The scheme of the algorithm for the serial systems is shown in Fig. 4.1, variant 1.

At the step of initialization, all the variables p (components of the probability matrix P) are set to their initial values ($0 < p_{ij} < 1,\ i \in \{\overline{1, N_x}\}, j \in \{\overline{1, N_y}\}$). Since our algorithm realizes the method of constrained optimization, the initial value of the probability variables is sometimes very important. Then, we generate the matrices X of optimized boolean variables. In our case of constrained problem, the large values of vector P components generate the values of X which are out of the feasible solutions area due to the constraints (3.3). Due to the constraints (3.3), the optimal initial values of the matrix P components do not exceed $N_F/(N_x N_y)$ [24].

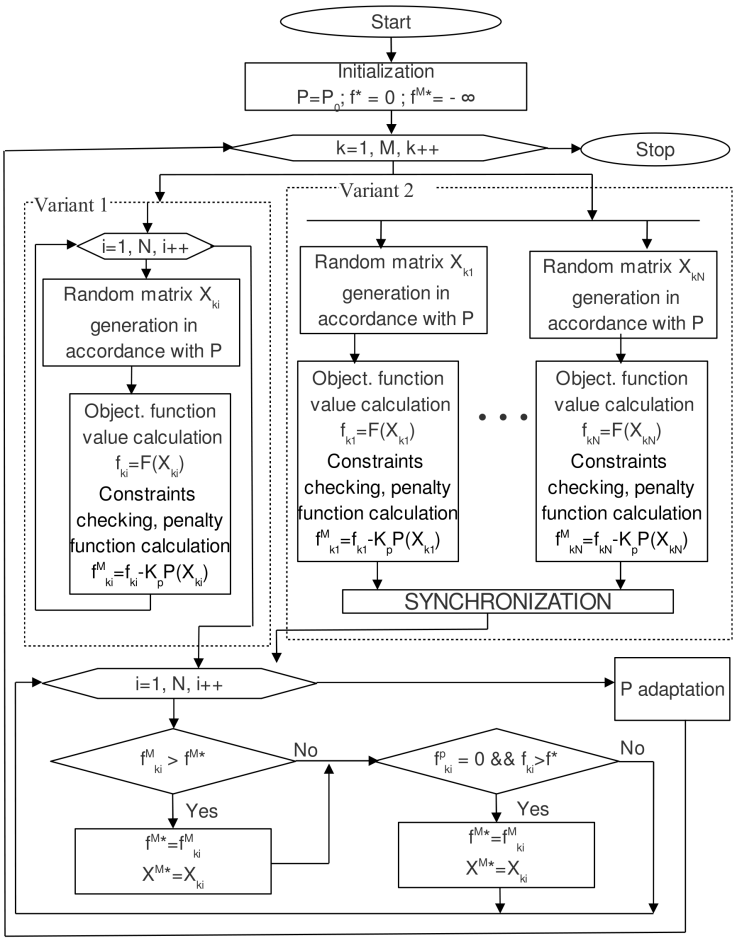

FIG. 4.1: An algorithm flowchart

Our algorithm is based on the Algorithm 2.1 . Here, instead of the probability vector P, we have matrix P. Also, we have matrix of Boolean variables X instead of a vector. It does not change the general scheme of our algorithm but this representation of the variables is more convenient for the further description.

We set the initial values of the matrix P equal to $N_F/(N_x N_y)$ but we have to reduce this value if several starts of our algorithm give us no results in feasible solutions area. This process is not illustrated in the Fig. 4.1 for the simplicity.

Instead of maximum number of steps (M in Fig. 4.1), we can use the maximum run time as the stop condition. In some cases, it is reasonable to use the maximum number of steps which do not improve the result achieved as the stop condition.

In the cycle $(i = 1, N)$, we generate the set of N matrices X_{ki} in accordance

with the probability matrix P. Then, the objective function is calculated for each exemplar X_{ki}.

To take into consideration the constraints (3.3), we modify the step of X exemplars generation.

Algorithm 4.1. Random X matrices generating

1. $X_{set} = \emptyset$; $n = 0$;
2. while $n < N_F$ do
2.2. for each $i \in \{1, N_x\}$: $S_i = \sum_{j=1}^{N_Y} p_{ij}$;
2.3. $S_x = r_x \sum_{i=1}^{N_Y} S_i$;
2.4. select minimum i so that $\sum_{k=1}^{i} S_i >= S_x$;
2.5. $r_y = f_{Random}()$;
2.6. $S_y = r_y S_i$;
2.7. select minimum j that $\sum_{l=1}^{j} p_{ij} >= S_y$;
2.8. if $(i, j) \in R_z$ then goto 2.2;
2.9. else $X_{set} = X_{set} \cap (i, j)$; goto 2;

Here, X_{set} is a set of coordinates (numbers of columns and rows) of the resulting matrix X which are equal to 1, N_F is quantity of the facilities placed, $f_{Random}()$ is a function with random value in range [0,1).

As the result of our the Algorithm 4.1, we have a matrix $X : X \notin R_z$.

The solution of various practical problems with the probability changing method shows the best result if we use the multiplicative adaptation of elements of matrix P with a rollback procedure [2]. In this case, the components of the vector P are never set to the value of 0 or 1 which may cause that all the further generations of the X vector have the same value at the corresponding position.

In Algorithm 2.1, all Boolean variables are considered as independent and the value of an element p_{ij} of the matrix P at the kth step can be calculated as

$$p_{k,i,j} = \frac{p_{(k-1),i,j} d_{k,i,j}}{\sum_{l=1}^{N_x} \sum_{m=1}^{N_y} p_{(k-1),i,j} d_{k,i,j}} \sum_{l=1}^{N_x} \sum_{m=1}^{N_y} p_{(k-1),i}. \tag{4.1}$$

Here, $d_{k,i,j}$ is the adaptation coefficient, $x_{i,j,k}^{max}$ and $x_{i,j,k}^{min}$ are the exemplars of the matrix X giving the maximum and the minimum values of the objective function (3.1). In case of multiplicative adaptation, $d_{k,i,j}$ does not depend on the step number k. In this case, the absolute value of adaptation step depends on the corresponding value of p_{kj}.

In the original variant of the probability method, the variables x_i and corresponding probabilities p_i are considered independent from each other. In case of Weber problem, we consider the variables x_{ij} as dependent from each other. As follows from the common sense (and proved experimentally), if the objective function has maximum value with X^{max} that $x^{max}_{i^*,j^*,k} = 1$ then, with some probability, the results at the next steps will be better than other variants if $x_{i^*,j^*,k+K} = 1$ and for the some surrounding area:

$$x_{i,j,k} = 1, i^* - N_S \le i \le i^* + N_S, j^* - N_S \le j \le j^* + N_S. \tag{4.2}$$

where N_S is the width and height of the surrounding area. The value of the coefficient d must be bigger for the points close to (i^*, j^*) for farther points, it must tend to 0.

We used the formulas below:

$$\begin{aligned}
d_{k,i,j} &= d^*_{k,i,j}/d^{"}w"_{k,i,j};\\
d^*_{k,i,j} &= \begin{cases} 1 + d_0/(1 + L((i^*, j^*), (i, j))), & |i^* - i| \le N_S \wedge |j^* - j| \le N_S \\ 1, & |i^* - i| > N_S \vee |j^* - j| > N_S \end{cases};\\
d^w_{k,i,j} &= \begin{cases} 1 + d_0/(1 + L((i^w, j^w), (i, j))), & |i^w - i| \le N_S \wedge |j^w - j| \le N_S \\ 1, & |i^w - i| > N_S \vee |j^w - j| > N_S \end{cases}.
\end{aligned} \tag{4.3}$$

Here, (i^*, j^*) and (i^w, j^w) are the closest to (i, j) points that $x^{worst}_{i^*,j^*,k} = 1$ and $x^{best}_{i^*,j^*,k} = 1$ correspondingly, X^{worst} is an exemplar of the X matrix with the maximum objective function value among the generated set and X^{best} is an exemplar of the X matrix with the maximum objective function value.

After several steps performed, the values of P matrix elements are close to 0 or 1 results in generation of the similar vector X exemplars which correspond to some local maximum. The rollback procedure is useful to avoid that situation. It sets the values of P matrix to the initial (or other) values. In the simplest case, rollback is performed after several steps which do not improve the best objective function value.

The best results are demonstrated with methods of partial rollback procedure which change some part of P matrix components or change all the components so that their new values depend on previous results. We can use the following rollback formula:

$$p_{kij} = (p_{k-1,i,j} + q_k p_0)/(1 + q_k). \tag{4.4}$$

Here, p_0 is the average value of the probability of the matrix P. The coefficient q_k can be constant or vary depending on the results of previous steps. For example, it may depend on the quantity of the steps which do not improve the maximum result (s_m).

$$q_k = w/s_m. \tag{4.5}$$

The weight coefficient w has to be chosen experimentally and depends on the frequency of the rollback procedure calls.

The adaptation of our algorithm for multiprocessor systems with shared memory can be performed by the parallel generation of the exemplars of the X vector and their estimation. The scheme of that version of our algorithm is shown in Fig. 4.1, variant 2. If our system has N_P processors, the cycle of generation of N exemplars of the vector X can be divided between the processors. Each processor has to generate N/N_p exemplars of the vector X and calculate the value of the objective function, left parts of the constraint conditions and calculate the modified objective function values. Organizing of the parallel thread takes significant computational expenses. In [8], authors estimate that expenses as 1000 operations of real number division.

5. Numerical Results

For testing purposes, we used a planar problem (3.1)–(3.5) with $N_x = 200$, $N_y = 400$. The map of our problem is shown in Fig. 5.1. Dark areas correspond to the cells with the higher weight (important points), white with zero weight (points there RF-coverage is not important). The scheme has 3 obstacles (barriers).

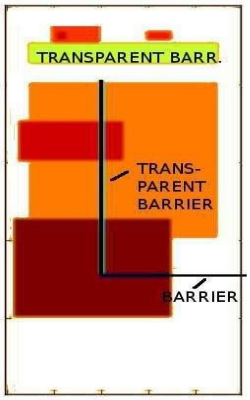

FIG. 5.1: Problem map

The experiments at 4-processor system with linear 100-dimension ($N_x = N_y = 10$) problem show that the parallel version runs 2.8 times faster than the serial one. For large-scale problems of our type with millions of variables, the parallel efficiency is almost ideal (0.93-0.97) for 1000 variables and more.

The average value 0.95 of the parallel efficiency is calculated as the average speed-up coefficient after 10 runs for 5 different objective functions.

The probability matrix changing for $N_F = 10$ is shown in Fig. 5.2.

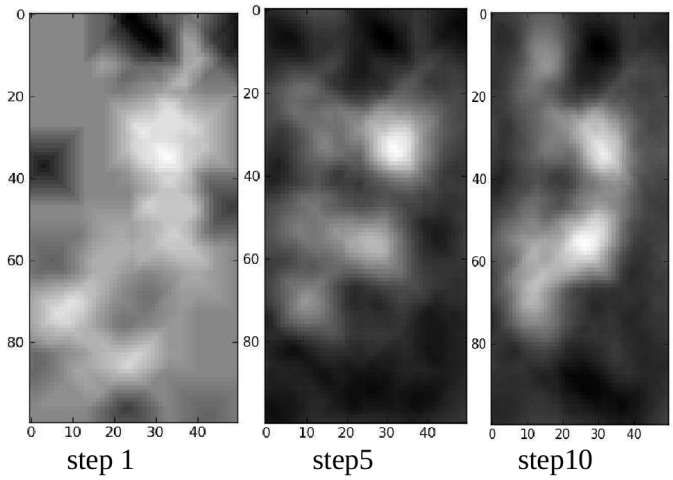

FIG. 5.2: Probability matrix changes

The results can also be easily visualized (Fig. 5.3). Here, dark areas of the map show the areas uncovered with the forecast of the RF signal.

Here, the color of each point (i, j) shows the anticipated signal level multiplied by the weight coefficient $w_{i,j}$.

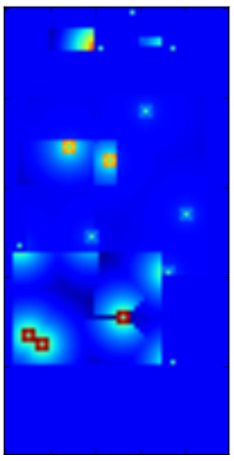

FIG. 5.3: Results visualization

6. Conclusion

Proposed modified algorithm based on the probability changing method can be used for the approximate solution of the planar generalized Weber problem with an arbitrary monotone non-Euclidean metric. Computational facilities of the modern computers (multiprocessor systems, inexpensive clusters) allow solving such problems with the appropriate accuracy. Algorithm parameters tuning and its computational complexity evaluation are subject of the future investigation.

REFERENCES

1. A. N. Antamoshkin: *Optimizatsiya funktsionalov s bulevymi peremennymi (Engl.:Optimization of functionals with Boolean variables)*. Tomsk University Press, Tomsk, 1987 (in Russian).
2. A. Antamoshkin, H. P. Schwefel, A. Toern, G. Yin and A. Zhilinskas: *Systems Analysis, desing and optimization. An introduction*, Ofset, Krasnoyarsk, 1993.
3. M. Bischoff, T. Fleischmann and K. Klamroth: *The multi-facility location-allocation problem with polyhedral barriers*, Computers and Operations Research **36** 2009, 1376–1392.
4. J. Brimberg and R. F. Love: *Properties of ordinary and weighted sums of order p used for distance estimation*, Recherche Operationnelle, **29** 1995, 59–72.
5. A. V. Cabot, R. L. Francis and M. A. Stary: *A network flow solution to a rectilinear distance facility location problem*, American Institute of Industrial Engineers Transactions **2** 1970, 132–141.
6. R. Chen: *Location problems with costs being sums of powers of Euclidean distances*, Comput. & Ops. Res. **11** 1984, 285–294.
7. L. Cooper: *An extension of the generalized Weber problem*, Journal of Regional Science, Vol.8 **2** 1968, 181–197.
8. L. Dagum and R. Menon: *OpenMP: An industry-standard API for shared-memory programming*, IEEE Computational Science and Engineering (IEEE Computer Society Press), Vol. 5 **1** 1998, 46–55.
9. M. S. Daskin: *Network and discrete location: models, algorithms, and applications*, John Wiley & Sons, Inc., New York, 1995.
10. A. S. Degterev, A. V. Kanashkin and A. D. Sumarokov: *Obobshenie genetich-eskikh algoritmov i algoritmov skhemy MIVER (Engl.: Generalization of the tenetic algorithms and algorithms based on MIVER scheme)*, Issledovano v Rossii, Vol.**7** (2004), URL: //http://zhurnal.ape.relarn.ru/articles/2004/153.pdf (accessed: 10.01.2012).
11. Z. Drezner and G. O. Wesolowsky: *A trajectory method for the optimization of the multifacility location problem with lp distances*, Management Science **24** 1978, 1507–1514.
12. Z. Drezner and H. Hawacher: *Facility location: applications and theory*, Springer-Verlag, Berlin, Heidelberg, 2004.
13. S. P. Fekete: *On the continuous Fermat-Weber problem*, Operations Research Vol. 53 **1** 2005,61–76.

14. M. R. Garey and D. S. Johnson: *Computers and intractibility: a guide to the theory of NP-completeness*, W. H. Freeman and Co., San Francisco, 1979.

15. M. Gugat and B. Pfeiffer: *Weber problems with mixed distances and regional demand*, Math Meth Oper Res **66** (2007), 419–449.

16. S. L. Hakimi: *Optimum Locations of switching centers and the absolute centers and medians of a graph*, Operations Research, **12(3)** 1964, 450–459.

17. S. L. Hakimi: *Optimum distribution of switching centers in a communication network and some related graph theoretic problems*, Operations Research, **13(3)** 1965, 462–475.

18. M. J. Hodgson, R. T. Wong and J. Honsaker: *The P-centroid problem on an inclined plane*, Operations Research **35** 1987, 221–233.

19. J. Hu and E. Goodman: *Wireless access point configuration by genetic programming*, in Evolutionary Computation, 2004. CEC2004. Congress on, Vol.**1** 2004, 1178–1184.

20. Yu. Finkelstein et all: *Diskretnaya optimizatsiya (Engl.: Discrete optimization)*, in: Metody optimizatsii v ekonomiko-matematicheskom modelirovanii, Nauka, Moscow, 1991, 350–367 (in Russian).

21. R. L. Francis and A. V. Cabot: *Properties of a multifacility location problem involving Euclidean distances*, Naval Research Logistics Quarterly **19** 1972, 335–353.

22. O. Kariv and S. L. Hakimi: *An algorithmic approach to network location problems. II. The p-medians*, SIAM Journal on Applied Mathematics **37(3)** 1979, 539–560.

23. L. A. Kazakovtsev : *Adaptation of the variant probability method for the problems of assortment planning and others*, Vestnik NII SUVPT (Research Institute of Control Systems, Wave Processes and Technologies, Krasnoyarsk) **6** (2001), 42–50.

24. L. Kazakovtsev: *Wireless access points placement as a discrete optimization problem*, URL http://eprints.ictp.it/494, (accessed 10.01.2012).

25. R. F. Love and J. G. Morris: *Computation procedure for the exact solution of location-allocation problems with rectangular distances*, Naval Research Logistics Quarterly **22** 1975, 441–453.

26. R. F. Love and W. G. Truscott, J. Walker: *Terminal location problem: a case study supporting the Status Quo*, J. Opl Res. Soc. **36** 1985, 131–136.

27. S. Masuyama, T. Ibaraki and T. Hasegawa: *The computational complexity of the m-center problems on the plane*, The Transactions of the Institute of Electronics and Communication Engineers of Japan **64E** 1981, 57–64.

28. N. Megiddo and K. Supowit: *On the complexity of some common geometric location problems*, SIAM Journal on Computing **13** 1984, 182–196.

29. P. B. Mirchandani and R. Francis, editors. *Discrete location theory*, Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons, Inc., New York, 1990.

30. H. Muhlenbein, M. Shomisch and J. Born: *The parallel genetic algorithm as function optimizer*, Proceedings of the fourth Conference of Genetic Algorithms (Morgan Kaufmann, San Mateo), 1991, 271–278.

31. I. A. Osinuga and O. N. Bamigbola: *On the minimum norm solution to Weber problem*, SANSA Conference Proceedings (2007), 27–30.

32. P. M. Pardalos, L. Pitsoulis, T. Mavridou and M. G. C. Resende: *Parallel search for combinatorial optimization: genetic algorithms, simulated annealing, tabu*

Search and GRASP, in: Parallel Algorithms for Irregularly Structured Problems: Lecture Notes in Computer Science, **980** 1995, 317–331.

33. T. S. Rappaport: *Wireless communications: principles and practice, 2nd ed.*, IEEE Press, 2002
34. M. G. C. Resende and C. C. Ribero: *Greedy randomized adaptive search procedures*, International Series in Operations Research & Management Science **57** "Handbook on Metaheuristics" 2003, 219–249
35. P. S. Stanimirovic and M. Ciric: *Single-facility Weber location problem based on the lift metric*, URL http://arxiv.org/abs/1105.0757 (accessed 10.01.2012).
36. H. Uster: *Weighted sum of order p and minisum location models*, Ph.D. Thesis, McMaster Univer- sity, Canada, 1999.
37. J. E. Ward and R. E. Wenll: *Using block norms for location modelling*, Operations Research **33** 1985, 1074–1090.
38. G. Wesolowsky: *The Weber problem: history and perspectives*, Location Science **1** 1993, 5–23.

Lev A. Kazakovtsev
Siberian State Aerospace University
Department of Information Technologies
prosp.Krasnoyarskiy Rabochiy, 31
660014 Krasnoyarsk, Russian Federation
levk@ieee.org