LOCATION OF BANKING AUTOMATIC TELLER MACHINES BASED ON CONVOLUTION

Mansour A. Aldajani and Hesham K. Alfares

[1]

Abstract

In this paper, the problem of determining the optimum number and locations of banking automatic teller machines (ATMs) is considered. The objective is to minimize the total number of ATMs to cover all customer demands within a given geographical area. First, a mathematical model of this optimization problem is formulated. A novel heuristic algorithm with unique features is then developed to efficiently solve this problem. Finally, simulation results show the effectiveness of this algorithm in solving the ATM placement problem.

Index Terms - Facility location, Mathematical models, Optimization algorithms, ATM site selection.

Corresponding Author: Mansour A. Aldajani
Systems Engineering Department
King Fahd University of Petroleum and Minerals
Dhahran 31261
Saudi Arabia
Email: dajani@kfupm.edu.sa
Fax: +966-3-8603315
Phone: +966-3-8602215

[1]This work is supported by King Fahd University of Petroleum and Minerals.

I. INTRODUCTION

Facility location problems are classical optimization problems that have numerous applications, especially in the service industries. Examples of these applications include optimal location of gas stations, health care units, warehouses, police stations, and power plants. Facility location models determine the minimum-cost location of a set of facilities to satisfy a set of demands (customers), subject to a set of constraints.

Automatic teller machines (ATMs) are among the most important service facilities in the banking industry. Since their appearance some 35 years ago, ATMs have literally changed the face of banking. The number and impact on the banking and retail business is growing steadily. The number of ATMs in the United States grew from only 25,000 in 1981 to more than 150,000 in 1999 [1]. While most of these ATMs are located at banks, there is a growing number of ATMs located off-premises. Bank Network News magazine [2] reports that the number of off-premise ATMs in the U.S. jumped from 28,700 in 1994 to 67,000 in 1997. There are many factors that banks take into consideration in order to determine location priorities for ATM sites. According to [1], the concerned bank must first determine if its main objective of placing a new off-premise ATM is visibility or free income. The usual first step is to determine where potential customers live, where they work, and what main roads they use [3]. Customer surveys as well as geographic, demographic, economic, and traffic data are useful for answering these questions. Other considerations include safety, cost, convenience, and visibility. Quite often, malls, supermarkets, gas stations, and other high-traffic shopping areas are prime locations for ATM sites. In this paper, the priorities for different potential ATM locations will be assumed given, based on a-priori analysis of all the applicable factors.

As recently surveyed in [4], the literature on facility location models and algorithms is quite large. However, attention to bank ATM location has been scarce. Assuming ATMs are placed only in bank branches, ATM location could be merged with the branch location problem. In this case, bank branch location approaches described in [5], [6], and [7] become applicable. The work in [8] uses queueing analysis to evaluate the workload and congestion at existing ATM locations, in order to determine in which locations to install additional ATMs. Few authors consider off-site ATM location, but only as an example within a larger given class of service facilities, such as discretionary service facilities, hierarchical commercial facilities, and immobile service facilities with stochastic customer demands. Another study developed models and algorithms for what they called "discretionary service facilities", such as gas stations and automated teller machines [9]. Generally, customers do not regard these facilities as end destinations, but they will use their services if they pass by them on their way on planned trips from one location to another. In [9], two equivalent integer programming models were formulated to locate N facilities in order to maximize the potential customer flow. The study also developed a greedy heuristic and a branch-and-bound algorithm to solve this problem. Alternatively, the study determined the minimum number of facilities required to a intercept the flow of a given fraction of customers. This problem was extended in [10] by allowing the service facilities to be congested.

In this work, we propose a completely new approach of solving the ATM location problem. The new approach differs from the previous ones in three aspects. Unlike previous approaches which demand speciality and complex model building process, the new approach uses very simple user interface to build the model. Second, the solution in the new approach is obtained using a simpler and more efficient mathematical technique. Lastly, the new approach allows any arbitrary service demand pattern and any service degradation model, allowing it to be more applicable to real-life problems.

The remainder of this paper is organized as follows. Definition of the service and demand patterns used in this study are described in the following section. The ATM location problem is formulated in Section III. The solution algorithm is described in Section IV followed by simulation and verification in Section IV-C.

II. Design Considerations

The main advantage of the proposed scheme is that it provides high flexibility for location specialists to choose arbitrary service and demand patterns by selecting proper structures of the matrices A and D. In the following, we describe in more detail the role of these two matrices in the model design process.

A. *Role of the Service Pattern Matrix A*

The service pattern matrix A describes how the quality of service changes as customers move away from the machine. Fig. 1 shows a simple illustration of a 5×5 matrix A. In this example, the service level starts with 100% at the center cell and then decrements inversely-proportional to the Euclidian distance from the machine. Another example of a rectilinear distance relation is shown in Fig. 2. Fig. 3 shows the contour plots of the two matrices. Using a similar approach, many other service patterns can be designed by choosing proper values of the matrix A. In general terms, the algorithm allows any arbitrary pattern of A which makes it suitable for modelling real location problems.

$$A = \begin{bmatrix} 26.1 & 30.9 & 33.3 & 30.9 & 26.1 \\ 30.9 & 41.4 & 50.0 & 41.4 & 30.9 \\ 33.3 & 50.0 & 100.0 & 50.0 & 33.3 \\ 30.9 & 41.4 & 50.0 & 41.4 & 30.9 \\ 26.1 & 30.9 & 33.3 & 30.9 & 26.1 \end{bmatrix}$$

Fig. 1. *An example of the service pattern matrix A (Euclidian distance model).*

$$A = \begin{bmatrix} 0 & 10 & 40 & 10 & 0 \\ 10 & 40 & 70 & 40 & 10 \\ 40 & 70 & 100 & 70 & 40 \\ 10 & 40 & 70 & 40 & 10 \\ 0 & 10 & 40 & 10 & 0 \end{bmatrix}$$

Fig. 2. *Another example of the service pattern matrix A (Rectilinear distance model).*

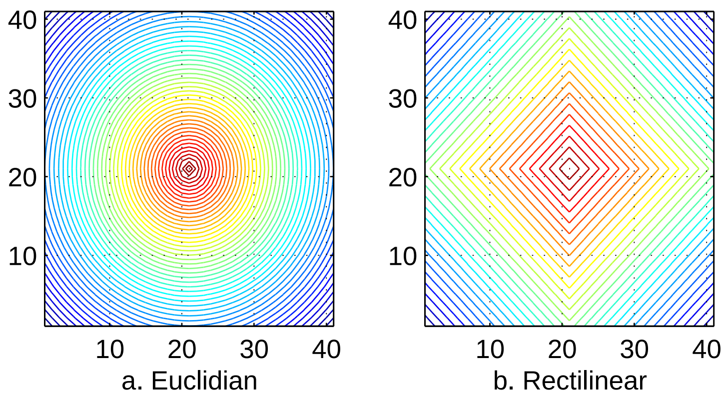

Fig. 3. *Contour plots of the two service patterns in Fig 1 and 2*

B. Role of the Demand Matrix D

The demand matrix D plays a major role in the placement of ATMs using the proposed algorithm. It provides flexibility in choosing any type of desired demand pattern. For the sake of illustration, Fig. 4 shows an example of a color-coded map that represents the coverage demand pattern in different parts of an actual geographical region at the center of Riyadh, Saudi Arabia. Each color represents a level of demand. In this example, the regions with green color have the highest demand. The blue and white colors represent high and normal demand regions respectively. The red color represents no-demand regions where the algorithm should avoid assigning ATMs.

The algorithm then interprets this colored map and builds the demand matrix D. The interpretation of the values of D is as follows. The high demand regions are reflected in D

by positive values with magnitude that is proportional to the demand level. The positive values in D would result in small contributions in C_n (see equation (10)) and therefore will be chosen first for machine locations. On the other hand, negative values in D indicate that these regions should be avoided. In this case,the convolution values in C_n will be large and therefore the algorithm will avoid assigning machines at these regions. Finally, regions with normal coverage priority will be reflected by zero values in the matrix D. As a result, the matrix D will be mostly full of zeros since normal coverage is usually the default case. The sparsity of the matrix D helps in substantially reducing the computations in the proposed scheme as will be described in the next section.

Using this color-coding technique, designers can set any arbitrary number of relative levels of demand. Although the example in Fig. 4 shows only four color codes, this number can be increased as desired according to the relative demand levels on hand.

III. Problem Formulation

In this paper, the problem of finding the minimum number of ATMs and their locations given arbitrary demand patterns is considered. In the following, the variables used in modelling the placement problem are defined.

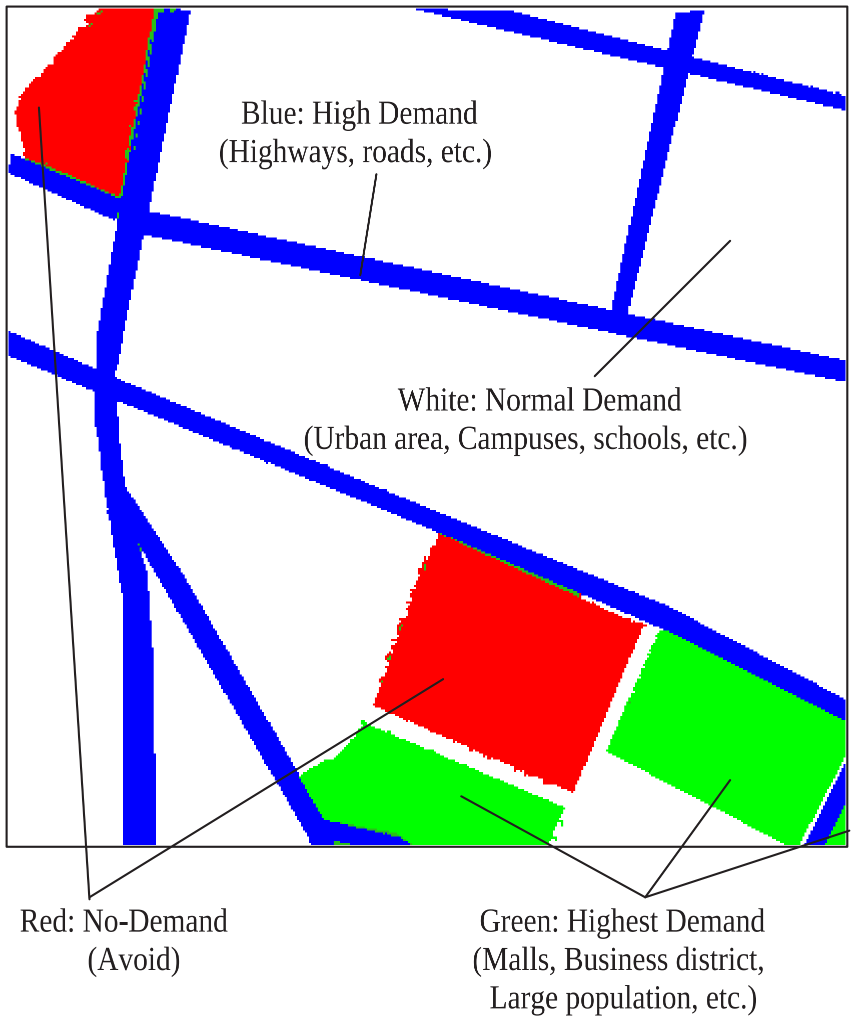

Fig. 4. *Example of designing the demand levels on a real map using color codes*

N	Total number of machines.
$s_n(x, y)$	Service supply from the n^{th} machine to location (x, y).
$d(x, y)$	Service demand at location (x, y).
$e(x, y)$	Difference between supply and demand at location (x, y).
α	Service margin; a constant that specifies the difference between supply and demand.
S_n	$(I \times J)$ supply matrix containing the discretized values of $s_n(x, y)$.
D	$(I \times J)$ demand matrix containing the discretized values of $d(x, y)$.
E	$(I \times J)$ difference matrix containing the discretized values of $e(x, y)$.
A	$(I_A \times J_A)$ fixed matrix that represents the degradation pattern of the service quality away from each machine
b	Frame penalty value
L_n	Location matrix indicating the location of the n^{th} machine
(u_n, v_n)	Coordinates of the n^{th} machine.
E_n	Difference matrix after assigning machine n.
$e_{min}(n)$	Smallest element inside E_n.

The objective of the placement problem is to minimize the total number of machines N such that the service supply exceeds the demand by a fixed amount α all over a confined 2-D space Γ. In mathematical terms, we can write

$$min \quad N \tag{1}$$

such that

$$e(x,y) \triangleq \max_{n=[1,N]} \{s_n(x,y)\} - d(x,y) \geq \alpha \qquad \forall \ x,y \in \Gamma \tag{2}$$

The quantity $s_n(x,y)$ represents the service supply from the n^{th} ATM to the user at coordinates (x,y). This quantity is dependent mainly on the service pattern of the machine, i.e., how the service level (SL) varies around an ATM machine. In most practical cases, the service level by a certain machine decreases as we move away from that machine. We assume here that the SL at any point is associated with only one machine that delivers the maximum service. The SL received from other machines at this specific point will be simply ignored. The term $d(x,y)$ represents the service demand level at point (x,y). Priority coverage, forbidden regions, streets and highways all can be incorporated within this term.

The choice of the service margin α is dependent on the problem at hand. Large α means that the supply will exceed demand by a large amount. Obviously, this will be at the cost of increasing the number of ATM's.

Discretization of the Model

To solve the placement model discussed above, the variables are first discretized into a finite number of uniform grid points of size (I,J). The number of divisions in the grid depends on the required resolution and available computational power. The variables $e(x,y)$, $s_n(x,y)$, and $d(x,y)$ are discretized in 2-D Euclidian space to form the matrices E, S_n, and D respectively. Therefore, the optimization problem can be written in matrix format as

$$\min \quad N \tag{3}$$

subject to

$$E_N(i,j) = \max_{n=[1,N]} \{S_n(i,j)\} - D(i,j) \geq \alpha \quad \forall \quad i,j \tag{4}$$

where E_N is the difference matrix of size $(I \times J)$ after assigning N machines, S_n is the supply matrix of the n^{th} machine, and D is the demand matrix.

The matrix S_n can be obtained from the convolution of two matrices as follows

$$S_n = A \otimes L_n \tag{5}$$

where the symbol $\otimes$ indicates the 2-dimensional convolution given by the expression

$$S_n(i,j) = \sum_{r=-\frac{I_A}{2}}^{\frac{I_A}{2}} \sum_{s=-\frac{J_A}{2}}^{\frac{J_A}{2}} A(r,s) L_n(i+r, j+s). \tag{6}$$

The matrix A of size $(I_A \times J_A)$ is a fixed service pattern matrix of the machines. The matrix L_n indicates the location of the machine n. If we denote this location by the coordinates (u_n, v_n) then L_n has all its elements equal to zero except at (u_n, v_n) where it is equal to "1". In other words,

$$L_n(i,j) = \begin{cases} 1 & \text{at} \quad (u_n, v_n) \\ 0 & \text{elsewhere.} \end{cases} \tag{7}$$

For the sake of illustration, suppose that

$$A = \begin{bmatrix} 30 & 50 & 30 \\ 50 & 100 & 50 \\ 30 & 50 & 30 \end{bmatrix} \text{ and } L_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \text{ Then } S_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 30 & 50 & 30 \\ 0 & 50 & 100 & 50 \\ 0 & 30 & 50 & 30 \end{bmatrix}.$$

The convolution values outside the range of the matrix L_n are simply truncated. Notice that the objective of the convolution here is to surround the unique non-zero element in L_n

with the service pattern matrix A. Therefore, the convolution operation in this case can be performed very efficiently by simply centering the elements of the A matrix at (u_n, v_n).

Notice also that minimizing the number of machines N is equivalent to minimizing the summation norm of the location matrices L_n for all machines. In view of this fact, the optimization problem can finally be written as

$$\min \quad \|\sum_{n=1}^{N} L_n\| \tag{8}$$

subject to

$$E_N = \max_{n=[1,N]} \{A \otimes L_n\} - D \geq \alpha\Theta. \tag{9}$$

where Θ is a matrix full of ones. The representation of the placement problem in this matrix format helps in borrowing useful tools from matrix theory to find a near-optimal solution for this problem as will be discussed in the next section.

IV. Solution of the Placement Problem

The optimization problem given by (8-9) is solved in this study using a new and simple heuristic approach. This approach turns out to offer high flexibility in choosing arbitrary service and demand patterns. It also allows a simple human user interface modelling of the problem and provides the solution in relatively short time. The solution approach is described in the following.

First, the fixed service pattern matrix A and the demand matrix D are given by the designer. Then, the algorithm will compute the *service level contribution* of every point on the grid to its neighboring points in case the given point is chosen as a machine location. This, off course, takes into account the given demand pattern. Then the point that results in the highest neighborhood coverage is chosen as the new machine location.

After placing each machine, the matrix E is updated and the process is repeated to choose the ne

the service ma

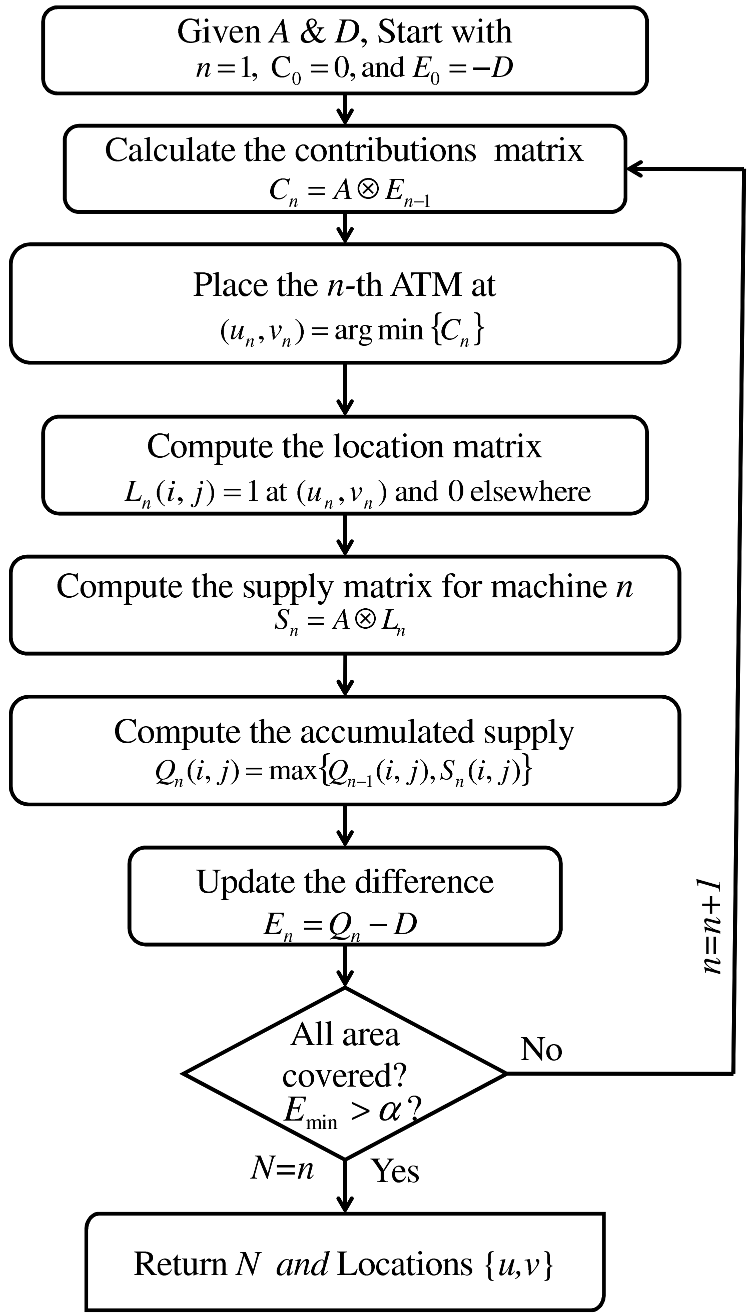

Fig. 5. *The proposed solution algorithm.*

A flow chart of the proposed algorithm is shown in Fig. 5. To determine the contribution of each point on the grid to the service distribution within the grid in case it is chosen as a machine location, the service pattern A is convolved with the existing difference matrix

E_{n-1} from previously assigned machines, i.e.,

$$C_n = A \otimes E_{n-1}, \quad E_0 = -D. \tag{10}$$

The matrix C_n describes the contribution provided by the ATM when located at each point in the grid to the neighboring points given the previous difference matrix E_{n-1}. The role of the convolution here is as follows. For each point on the previous difference matrix E_{n-1}, the matrix A is centered at that point and dot-multiplied with the intersecting sector of E_{n-1}. The multiplication values are then summed up and the answer is stored at the corresponding point in C_n. This convolution process is repeated for all other points in E_{n-1}. Then, the coordinates that correspond to the minimum value of the matrix C_n are then chosen as the location of the n^{th} machine, i.e.,

$$(u_n, v_n) = \underset{(i,j)}{\operatorname{argmin}}\ C_n. \tag{11}$$

When a set of points give the same minima, the middle among these points is arbitrarily chosen to break the tie.

To understand the motivation behind this choice, suppose first that the space has equal demand all over the area. If $n-1$ machines are already placed, then E_{n-1} will have large positive values of SLs around these machines. When A is convolved with E_{n-1}, the convolution values will be smallest at the location that is farthest away from the previous $n-1$ machines. Consequently, (11) will chose this location for the next machine. This guarantees that the new machine will be placed at locations with poorest service.

Now suppose that a certain area has higher demand than others. In this case, negative values can simply be assigned in the corresponding regions in D. Since $E_0 = -D$, the convolution at these locations will be smallest and therefore they will be chosen first by (11) as machine location. In this way, the matrix D can be designed to fulfill any arbitrary demand patterns.

Once a new machine location is computed, the location matrix L_n is constructed from (7). The difference matrix is then updated as follows

$$E_n = Q_n - D, \qquad n = 1, 2, ..., N \tag{12}$$

where Q_n is the accumulated supply of service due to the machines: 1,....,n. The elements inside Q_n are obtained recursively from the expression

$$Q_n(i,j) = \max\ \{Q_{n-1}(i,j), S_n(i,j)\}, \quad i = 1, ..., I, \quad j = 1, ..., J, \quad Q_0 = \mathbf{0} \tag{13}$$

where $\mathbf{0}$ is the $(I \times J)$ zero matrix.

In summary, given the service pattern and demand matrices A and D, the location of the machines is determined by iterating equations (10-12) starting from $E_0 = -D$. The algorithm terminates when the constraint (2) is satisfied, or equivalently, the minimum difference: $e_{min}(n) \triangleq \min\ \{E_n\}$ exceeds the margin α. The algorithm then returns the total number of machines N, their locations, and e_{min}.

A. Penalizing Boundaries of the Demand Grid

Based on the discussions above, the proposed algorithm tends to assign machines at the boundaries of the covered area so that they will be farthest apart from each other. This will be at the cost of increasing the required number of machines. This problem can be resolved by augmenting one frame of penalty b around the demand matrix D as shown in Fig. 6. The main objective of this frame value is to push the machines inside the demand area. The optimal value of b is usually a positive number that depends on the size and content of the matrices A and D. The frame value can fine-tune the solution by improving the total coverage for the same number of machines. In this work, an outer loop is performed that implements a simple line-search to find the optimum frame value. In Section VI, we show a numerical example on deciding this value.

$$D = \begin{bmatrix} b & b & b & b & b \\ b & 10 & 5 & -10 & b \\ b & 5 & 5 & -10 & b \\ b & 0 & 0 & -10 & b \\ b & b & b & b & b \end{bmatrix}$$

Fig. 6. *Illustrative example of a demand grid surrounded by the penalty constant b*

B. Percentage Coverage of the machines

The algorithm also computes the percentage coverage for each of the machines as it assigns them one-by-one. After placing each machine, the accumulative percentage coverage (APC) is computed as the number of grid points in E_n that have SL greater than the margin α divided by the total number of grid points in E_n. The percentage coverage (PC) is then evaluated by simply computing the change of APC values from one assigned machine to the next. The algorithm returns both PC and APC with the solution as we shall show in the simulations section. The PC is essential information in locating the ATMs. One example of utilizing this information is to eliminate those machines with negligible percentage coverage, resulting in an overall cost reduction.

Another important issue in this problem is to determine the percent of total demand satisfied which can be defined as follows

$$\Upsilon = \frac{\sum_i \sum_j |B(i,j)|}{\sum_i \sum_j |D(i,j)|} \tag{14}$$

where

$$B(i,j) = \begin{cases} D(i,j) & \text{if} \quad E_N(i,j) > \alpha \\ Q_N(i,j) & \text{otherwise} \end{cases} \tag{15}$$

determines the demand covered by the ATMs. Unlike APC which considered number of points covered, Υ indicates how much demand is covered.

C. Solution Verification

The ATM placement problem is usually NP hard and its solution cannot be found analytically. Therefore, numerical methods are used for verifying the proposed algorithm. First the algorithm is implemented on simple models where solutions are known and the results are then compared [11]. Second, solution is verified by performing an exhaustive search on all possible locations. The search challenges the algorithm by trying to find one of the following

1. a lower number of ATMs that meets the service level requirements.
2. a different location of the same number of ATMs that provides better service coverage (higher e_{min}).

For example, suppose that the proposed algorithm gives a minimum number of ATMs equal to 5 together with their near-optimal locations. First, the exhaustive search will try all possible location combinations on the grid to locate 4 ATMs such that the service requirement is satisfied, i.e. $e_{min} > \alpha$. Next, the exhaustive search will also try to locate 5 ATMs in different places than those given by the algorithm to get a higher value for e_{min}. If both tries fail, then the algorithm can be claimed to achieve an optimal solution. For the exhaustive search, a reasonable grid size is used to make it computationally achievable. Solutions found by the convolution algorithm matched those found by exhaustive search for a set of 6 small problems.

Furthermore, the heuristic solutions have been compared to the optimum solutions produced by integer programming (IP) and exhaustive search. Comparisons with IP were limited to a set of 4 small test problems because optimum IP solutions are hard to attain for larger problems. In all 4 cases, the heuristic solutions matched the optimum IP solutions.

V. Computation complexity of the proposed scheme

In this section, the computation complexity of the proposed scheme is analyzed. From the discussions above, the proposed scheme has an outer loop as well as an inner loop. The outer loop searches for the optimal scalar frame penalty value while the inner loop searches for the optimal number of machines and their locations by implementing the algorithm of Fig. 5.

For the outer loop, a simple line search was found sufficient to locate the optimal scalar frame penalty. The search is limited to the integer values in the range $[0, 500]$. Still, more efficient search algorithms could be adopted to find this value.

In the inner loop represented by Fig. 5, the only computationally expensive operation is the convolution $A \otimes E_{n-1}$. Row convolution costs m^2 multiplications where m is the number of grid points in the search space ($m = I \times J$). However, this number can be substantially reduced by utilizing available efficient schemes for computing the convolution. An example of these schemes is the convolution theorem which reduces the number of multiplications to $m \times \log m$ if m is a power-of-two. In addition, there are two observations that can further reduce the complexity of the convolution operation as follows.

1. The matrix E_n usually starts with a structure that consists mostly of zero elements (corresponding to normal demand in D). This matrix is then gradually filled up with non-zero values as new machines are assigned. Therefore, the sparsity of the matrix E_n can be exploited while computing the convolution to reduce the number of complex operations. For example, there is no need to compute the convolution in regions of E_n with zero values.
2. The search space for optimal locations decreases as new machines are assigned. Therefore, the number of complex operations in the convolution can be substantially reduced by ignoring those locations already meeting the coverage requirement.

VI. Computational Experiments

In this section, we demonstrate the performance of the proposed ATM placement algorithm through simple illustrative examples. Matlab was used to implement the algorithm on a 2.1 GHz personal computer with 256 MB of memory. The Matlab program provides a friendly User Interface (UI). This interface is used to input a color-coded map in a common image format (JPEG) to automatically generate the corresponding demand matrix D. It is also used to input the service pattern matrix A from the user with arbitrary size and values. It subsequently computes the number of machines and their locations and then shows them on the color-coded map. The program also returns the coordinates of the machines, e_{min}, and the percentage coverage (PC) of each assigned machine.

In our experiments, the size of the matrices D and A is fixed to 41×41 for both (corresponds to 1681 possible locations). Furthermore, the service margin is arbitrarily fixed in all instances to the normalized value $\alpha = 1$. We consider first the placement problem where the demand is the same for all points on the grid. This corresponds to $D = \mathbf{0}$. A rectilinear distance model is used to represent the degradation of service around the machines. Such model is common to represent street traveling distances in urban setting. The results are shown in Fig. 7. As expected, the machines were placed uniformly across the demand area starting from the center of the region. The number at the center of each segment indicates the order by which the machine was assigned. In this example, the number of machines needed to cover the entire area is 5 machines. The PC and APC of the assigned machines are shown in Fig. 8. Machine number 1 covered 66% of the whole area, while the other four machines covered 8.5% each.

In another experiment, we changed the demand matrix to include a region of high priority coverage. This region could be a bank, a shopping center, or a highly populated area. The region is simply drawn in a specific distinct color which is interpreted by the algorithm as high demand region. The algorithm then finds the solution and the results are shown in Fig. 9. In this case, the first machine assigned was moved to cover the high demand region first. The next machines were then assigned to cover the remaining

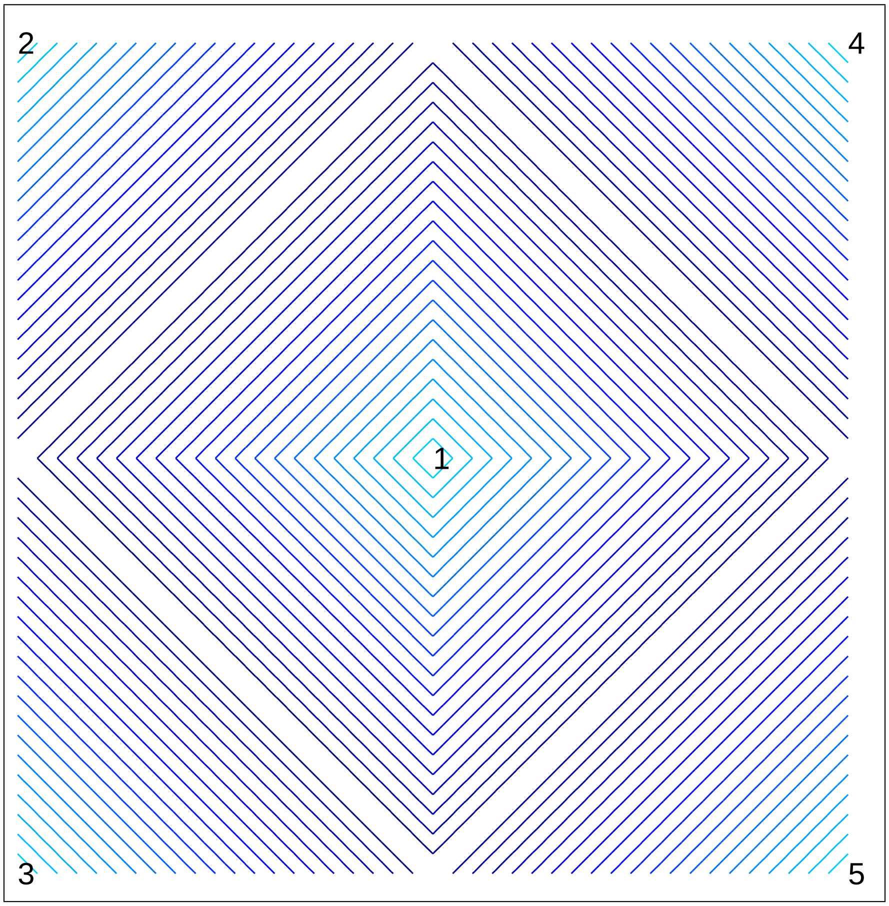

Fig. 7. *Result of machine assignment over an area with uniform demand.*

uncovered areas.

Another case considered a mesh of roads from a real city map. The roads are redrawn with a unique color. The algorithm reads the map and interprets this new color and assigns it an appropriate value in the matrix D. The solution is shown in Fig. 10. Notice that the first machine was assigned next to the road in a way as to enclose as much road distances as possible. The next machines were then assigned the same way.

Finally, an actual map for the down-town area of Khubar City in Saudi Arabia is considered. The map is shown in Fig. 11. The green regions are the shopping areas, the pink and blue lines are the main roads and the red area is the avoid-region. In this case, the grid resolution is 75×75 pixels. The algorithm is implemented for this configuration

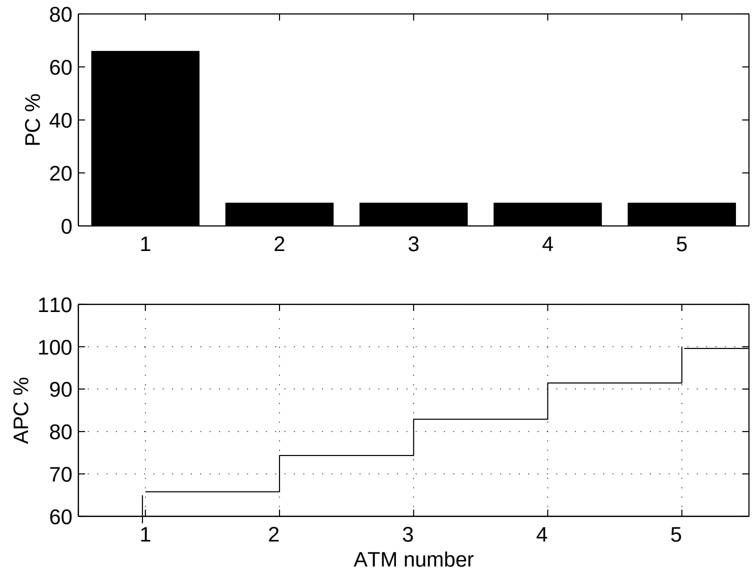

Fig. 8. *Percentage coverage (PC) and accumulative percentage coverage (APC) for the machines in Fig. 7.*

and the results are shown in Fig. 12. The percentage coverage is also shown in Fig. 13. The results are returned within 20 seconds. Looking closely at the results, we notice the following.

1. The 18 ATMs placed by the algorithm covered about 95% of the total demand space.

2. According to (14), the percentage demand coverage is 99.6%. This means that the remaining uncovered 5% of the demand space is not much significant.

3. The shopping areas are covered first by the algorithm as expected.

4. The remaining ATMs are located at the intersections of or along the main roads.

5. No ATM was placed at the avoid-region.

To choose the optimal frame value b, we considered again the map shown in Fig. 9. The number of machines is fixed to the optimum value (five machines). Then, the coverage of the machines is computed for different values of b. Fig. 14 shows the resulting total percentage coverage as a function of b. As we increase b the machines are pushed towards the center of the region increasing the contribution of the machines at the edges. After a

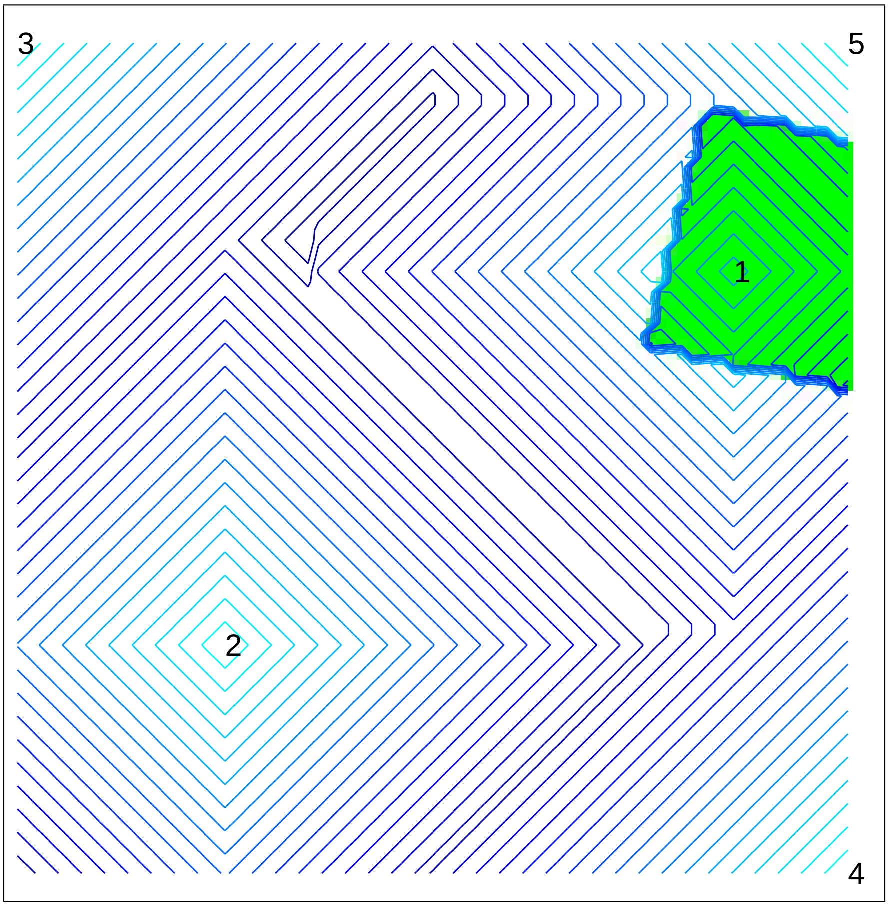

Fig. 9. *Result of the placement problem with high demand region.*

while, the machines are too much pushed that the regions at the edges are not covered. This causes the coverage to drop rapidly. This pattern is typical in all scenarios tested. In this example, the optimal value of b is 280.

In another test, we investigate numerically the computational complexity of the proposed approach. As mentioned earlier, the main advantage of using the convolution as a core process in the proposed algorithm is that there exists many ways of efficiently computing it. For example, the convolution theorem reduces the number of complex operations from m^2 to $m \log(m)$, with m being the grid size of the map ($m = I \times J$). The processing time for the proposed algorithm is proportional to the convolution complexity. In our simulations, we made use of the convolution theorem to reduce the processing time of the

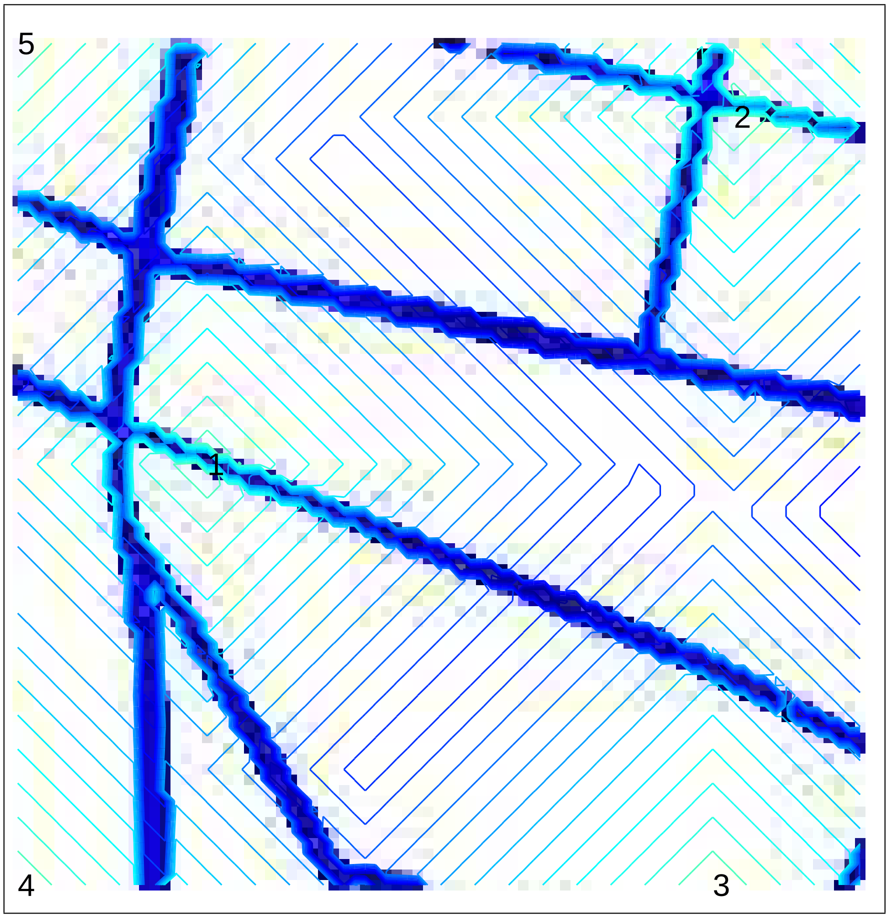

Fig. 10. *Result of the placement problem for a mesh of roads.*

algorithm. In Fig. 15, we show the processing time needed to assign a single station[2] for various values of the grid size m. The theoretical fit is also shown for comparison purpose. Clearly the processing time is reduced to the order of $m\log(m)$ resulting in a substantial saving in computations.

VII. Conclusion

In this work, we proposed a new approach for the placement of automatic teller machines (ATMs). The approach computes the minimum number of machines as well as their locations that satisfy the service level coverage requirements. It does so by implementing a new heuristic solution that is based on the 2-dimensional convolution. The proposed

[2]That is the same time required to go through one round of Fig. 5.

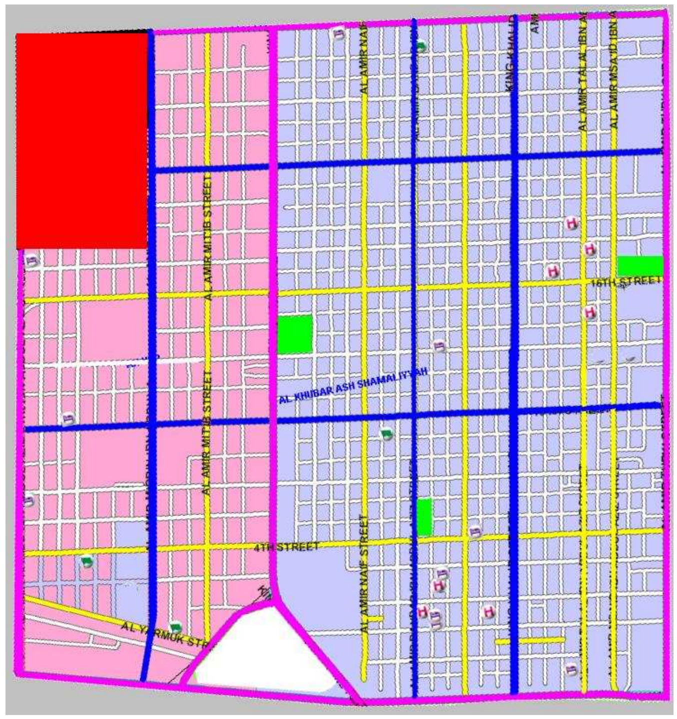

Fig. 11. *Down-town area of Khubar, Saudi Arabia. Green: Shopping areas, Blue and Pink: Main roads, Red: Avoid-region.*

approach provides a flexible means for choosing arbitrary service models and demand patterns, making it suitable for real applications. Experiments with the new algorithm show its efficiency and flexibility in solving ATM placement problems near-optimally.

References

[1] C. Wilson, Where would like that ATM?, America's Community Banker, vol. 8, no. 8, 1999, 18-22.

[2] Anonymous, The fine art of off-premise site selection, Bank Network News, Vol. 16, No. 10, 1997, 1-5.

[3] P. Adams, Choosing a choice ATM site, Credit Union Management, vol. 14, no. 3,

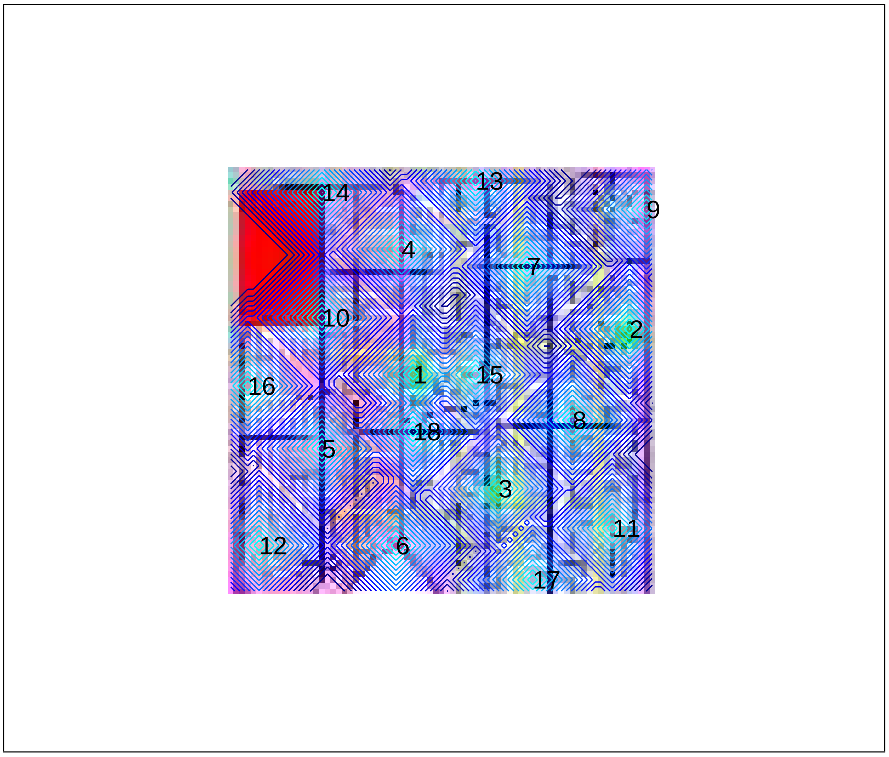

Fig. 12. *Resulting locations of the 18 ATMs and their service patterns.*

1991, 15-17.

[4] T.S. Hale, C.R. Moberg, Location science research. Annals of Operations Research, vol. 123, no. 1-4, 2003, 21-35.

[5] G. Cornuejols, M.L. Fisher, G.L. Nemhauser, Location of bank accounts to optimize float: an analytical study of exact and approximate algorithms, Management Science, vol. 23, 1977, 789-810.

[6] A.C.m. Hopmans, A spatial interaction model for branch bank accounts, European Journal of Operational Research, vol. 27, no. 2, 1986, 242-250.

[7] P.V. Boufounou, Evaluating bank branch location and performance: A case study, European Journal of Operational Research, vol. 87, no. 2, 1995, 389-402.

[8] P. Kolesar, Stalking the endangered CAT: A queueing analysis of congestion at automatic teller machines, Interfaces, vol. 14, no. 6, 1984, 16-26.

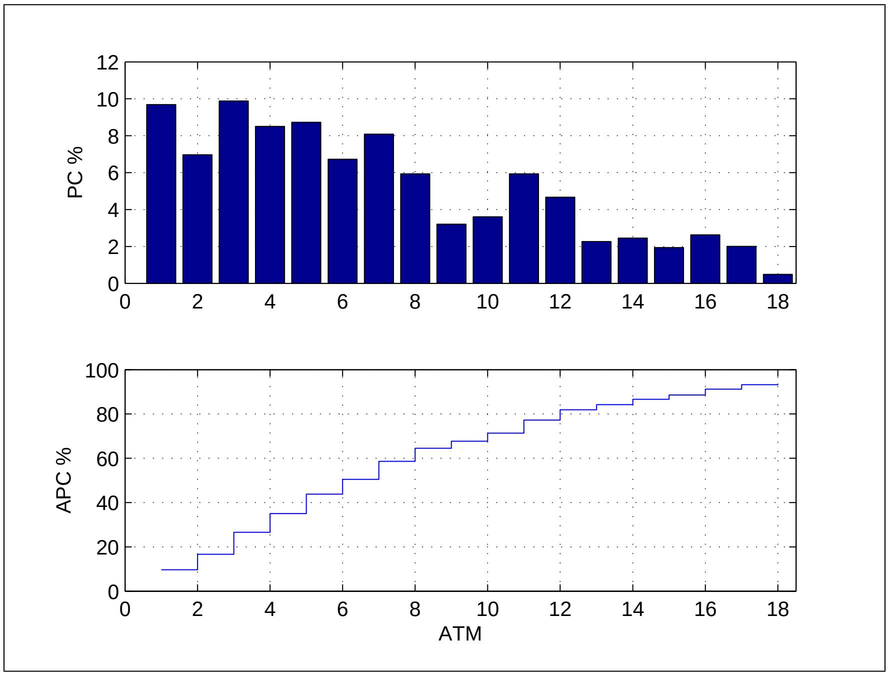

Fig. 13. *Percentage coverage of the 18 ATMs of Fig. 12.*

[9] O. Berman, The maximizing market-size discretionary service facility problem with congestion, Socio-Economic Planning Sciences, vol. 29, no. 1, 1995, 39-46.

[10] O. Berman, R. Larson, N. Fouska, Optimal location of discretionary service facilities, Transportation Science, vol. 26, no. 3, 1992, 201-211.

[11] B. Park, J. Yook, and H. Park, The determination of base-station placement and transmit power in an inhomogeneous traffic distribution for radio network planning, Vehicular Technology Conference (02), vol. 4, Sep. 2002, 2051 - 2055.

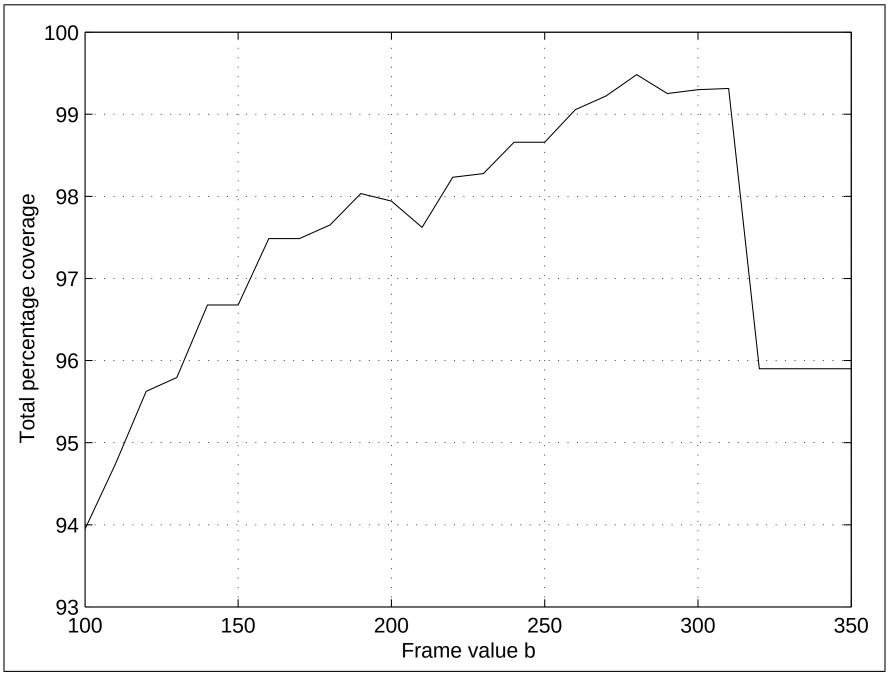

Fig. 14. *Effect of the frame value on the total percentage coverage.*

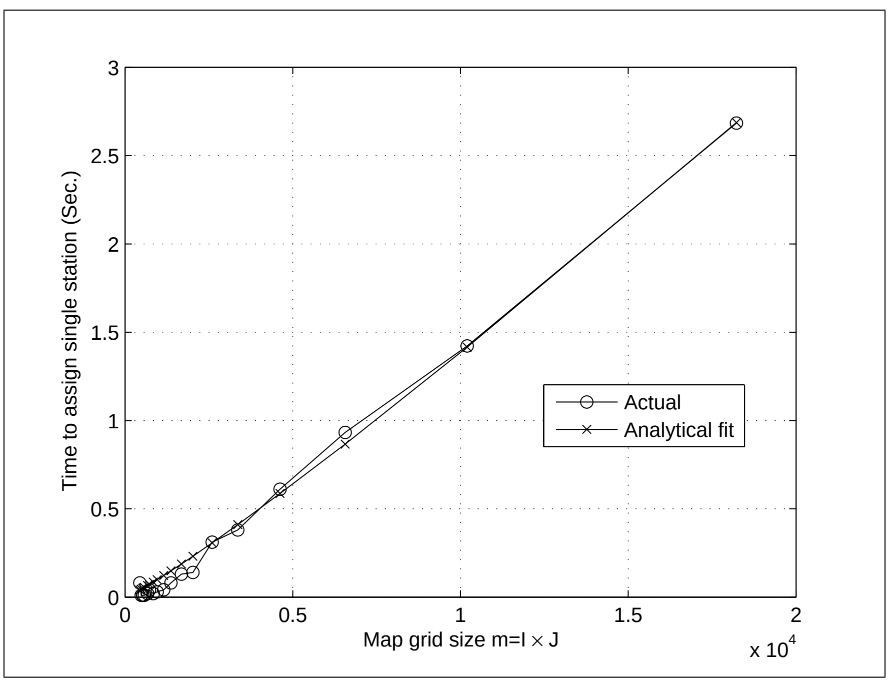

Fig. 15. *Time needed to assign single station as a function of the grid size* $m = I \times J$.