

A Market-Based Study of Optimal ATM'S Deployment Strategy

Alaa Alhaffa and Wael Abdulal

Abstract— ATMs are critical to the success of any financial institution. Consumers continue to list the location of ATMs as one of their most important criteria in choosing a financial institution, for that banks are willing investment more ATMs for the purposes of providing greater convenience and attracting more customers. But there must be some equilibrium number of ATMs in the market otherwise rivals will enter the market and take all non-served customers. In the competitive case, the bank with most ATMs which are optimally deployed by using strong strategies would win the competition and get all the customers. Based on Bank clients' base, this study has placed great emphasis on the ATM's Deployment Strategies in order to provide greater convenience to the customers, consequently, banks can attract more customers and increase its market share and profitability. Technically, three algorithms are designed and compared namely; Heuristic Approach, Rank-Based Genetic Algorithm using Convolution and Simulated Annealing using Convolution. Dual objective is set to achieve highest Percentage Coverage (PC) and less ATMs Number required for covering intended area of study. Three experiments are carried out to measure the performance of each Algorithm. The experimental results show that Rank Based Genetic Algorithm shows a significant improvement in PC over Heuristic Approach, recording minimum improvement of 2.2% and maximum improvement of 20.13%. And it shows that Simulated Annealing outperforms both Heuristic Approach by up to 26.32% and Genetic Algorithm using convolution by up to 2.288% in terms of Percentage Coverage value. Regarding the saving in number of ATMs, Simulated Annealing Algorithm saves up to 33 ATMs over Heuristic Approach and up to 6 ATMs over Genetic Algorithm using Convolution.

Index Terms—Heuristic Approach using Convolution (HAC), Rank Based Genetic Algorithm using convolution (RGAC), Simulated Annealing using Convolution (SAC), Automated Teller Machines (ATMs).

I. INTRODUCTION

In order to survive, both banks and ATM deployers need to anticipate new customer needs, respond much more rapidly to competitive changes and create new sources of customer value and service differentiation.

The ATM optimal Deployment Strategies offer the opportunity to provide greater convenience and to attract more customers by covering the money market with sufficient ATM facilities. These strategies also provide greater cost efficiency by finding the optimal number of

ATMs to be installed and greater profitability by increasing the ATM user base in order to earn much more transactions and services fees [1] as well as through the inflow of deposits from the depositors who consider ATM availability as a main factor in choosing their banks. ATMs have become a competitive weapon to commercial banks whose objective is to capture the maximum potential customers. One important fact to be noted is that commercial banks compete not only on the dimension of price but also on the dimension of location [2].

The problem of ATM deployment is seen to be NP-complete problem (it is analogous to the file server placement problem) [3]. In order to solve this problem, three algorithms are designed and compared namely; Heuristic Approach using Convolution (HAC) [4], Rank Based Genetic Algorithm using Convolution (RGAC) [5] and Simulated Annealing using Convolution (SAC) [6].

The first technique, (HAC) performs very efficiently in solving the ATM deployment problem as long as the size of the market is small, but as market size becomes larger HAC appears to be deficient in finding the best solution and it usually falls in local minima.

Coming to the second proposed technique, (RGAC) increases the search efficiency by improving the evolutionary process while meeting a feasible solution. Moreover, RGAC has proved to be a robust approach for solving the ATMs deployment problem and is able to provide high quality solutions in a reasonable time. Lastly, RGAC is scalable and effective in banking environment even when the size of the market is so large.

The novel technique (SAC) outperforms both RGAC and HAC. The simulation results of SAC show a significant improvement in both Percentage Coverage (PC) and saving in the number of ATMs as compared to previous Heuristic Algorithms HAC and RGAC. The reason behind the high performance of SAC is that SAC uses both global and local search techniques. It is more powerful in global search at high temperature and it becomes more powerful in local search as the temperature is reduced while RGAC is only powerful in global search and weak in local search. In addition to the above, SAC allows "uphill moves" to solutions of higher cost in order to avoid being trapped in poor local optima according to the so-called Metropolis criterion [7].

The rest of the paper is structured as follows: Section II indicates some important related works. Detailed description of the problem encoding and specific operators is explained in Section III. Section IV gives explanation of HAC algorithm. Section V describes the designing of RGAC. Section VI shows the formulation of SAC. Lastly, section VII describes the computer simulation results. Concluding remarks are contained in Section VIII.

Manuscript received March 18, 2011

Alaa Alhaffa, Dept. Economics, Osmania University, Hyderabad 500-007, India. (E-mail: alaa.haffa@yahoo.com)

Wael Abdulal, Dept. CSE, EC, Osmania University, Hyderabad 500-007, India. (E-mail: wael.abdulal@ymail.com)

II. RELATED WORKS

A. Literature on ATM Adopters

1) Customers:

A number of researchers have investigated the demographic characteristics of ATM adopters. [8] studied a Kuwaiti population, [9] studied a Canadian population, and [10] studied a Southeast Asian population and all got consistent results of adopter characteristics of ATM, in which ATM users tend to be young, married and have above average incomes and at least some high school education.

2) Banks:

Study [11] indicates the factors that encourage the banks to increase the investment in ATM network, like:

- The geographical market area of the adopting bank overlaps to a greater extent with the market areas of competing banks.
- The odds of ATM adoption also increase as the size of the bank increases.
- The odds of ATM adoption are higher in markets with faster population growth.
- The more binding the restrictions will be on branch deployment, and the more likely it is that a bank will circumvent the restrictions on branch deployment and deploy ATMs at new locations.

Adding to these factors, a study [12] identified several reasons why the banks invest in the ATM network. The most important reasons are the following:

- Reducing of the cost per transaction.
- Marketing advantages gained through expanding bank's presence on the market.
- Increasing the client's satisfaction.
- Profit gain, based on the charges for initiation of transactions to the non-bank ATM users
- Reducing time needed for the turnover of funds.
- Expanding geographical presence of the bank in the regions where the bank is not usually present.

B. Literature on Service quality

There are a number of studies that refer to the importance of clients' perceptions of quality. One important study [13] suggests that the criteria used by consumers mould their expectations and the perceptions of delivered service quality fit into 10 dimensions: tangibility, reliability, responsiveness, communication, credibility, security, competence, courtesy, understanding/knowing the customer and access. The ATM optimal deployment strategy satisfy such criteria like tangibility, communication, competence and access providing high convenience to the customer.

Another study [14] deals with service quality in the banking industry in general and in particular the application of the SERVQUAL instrument in commercial banks. Study [15] shows from the consumers' perspective, e-banking provides many benefits to individuals, such as immediate access to accounts and balances, ability to conduct remote banking transactions and investments, and completion electronic applications.

One important work [16] examined the problem of locating a single facility on a network when demand for service at the facility is a decreasing function of the distance between the customer and the facility. The objective is to maximize the demand served by the facility.

C. Literature on ATM Geographic Location

A study [17] suggested GIS based solution and states that Banks' deployment planning for branch/ATM needs modeling location-relevant data and providing fast and cost-effective site analysis to confidently and reliably select a new bank branch/ATM location such as Concentration of commercial areas, traffic patterns, workplaces or homes of customers whose demographics and purchase behavior match a bank's target customer profile.

D. Literature on used Algorithms for solving ATM deployment problem and those for solving similar problems

The case study [4] proposed a simple Heuristic Approach using Convolution (HAC). The previous work [5] suggested a modified Genetic Algorithm (RGAC) to solve the same problem; RGAC performs more effectively in the large scale deployments as compared with HAC. The simulation results show that RGAC improves the PC over the previous algorithm [4] using the same number of ATMs, also they exhibit that RGAC reduces the number of ATM machines used in HAC solution causing better presence of the bank in the market and saving in the cost of deploying extra useless machines. The previous work [6] presents the last and highest improvement made in terms of PC value and the saving in the number of ATMs by using SAC algorithm. The study [18] used all costs accompanied with the provision of ATM service as a parameter to choose the ATMs type and location. Coming to the applications of SA, this study is a pioneer work regarding using SAC algorithm to solve ATM deployment problem, there are similar applications of SA in wireless Sensor-Actor Network [19], and in Wireless Sensor Network Deployment [20].

III. ASSUMPTIONS

In order to simplify the complicated structure of the problem and gain control over the study's variables and experiments, some assumptions are required to be set as follow:

- ATMs are homogeneous, in line with this there exists only one matrix A. Matrix A (represents the degradation of ATMs utility as one moves away from its location) is predetermined and held constant for all machines.
- Competitive ATM market; where there is existence of many banks in the market. In case the ATM market is monopolized this leads to a smaller number of ATMs i.e., the availability of ATM services diminish if the monopoly decides to reduce the number of ATMs [21]. Also the decreasing competition may have harmful consequences from the consumers' point of view.
- There are no independent ATM deployers (IADs). The entry of independent deployers limits banks' use of ATM deployment as a way to enlarge their deposit market shares. The effects of IADs entry are:
 - 1) IADs' ATMs are accessible to all cardholders at the same price and consequently, banks become less differentiated by their networks. They have less incentive to deploy machines and their profits increase.

- 2) Consumer surplus also decreases as the first independent deployers enter the market: the IAD entry makes banks deploy less ATMs and it becomes increasingly difficult for cardholders to find a free machine. However, as more IADs enter, consumer surplus may increase if consumers sufficiently value the enlargement of the total ATM network [22].
- One price for the usage of ATM prevails in the market, i.e., there is no differentiation between banks regarding ATM fees.

IV. PROBLEM FORMULATION

The ATM placement problem is modeled and defined mathematically. The variables used in modeling of the intended problem are shown in the table I. The optimization problem is organized in such a way as to realize market clearance. In other words, the difference between CU and SU should be minimized. This difference can be expressed mathematically in equation 1:

$$E = SU - CU \geq \varnothing \tag{1}$$

Where E is the difference matrix of size (I×J) after assigning total number of machines, SU is the Service Utility matrix of all ATMs, \varnothing is the zeros matrix. The generation of previous matrices CU, and SU are explained as follows:

TABLE I. PROBLEM FORMULATION

Variable	Variable Description
N	-Total number of machines.
SU	-Service Utility matrix which represents the service supply side.
CU	-Client Utility matrix which represents the demand side.
E	- Difference or market clearing matrix.
A	- Matrix that represents degradation of service utility as a client moves away from each machine.
L _n	- Location Matrix indicates the location of the nth machine.
(U _{m,v,n})	- Coordinates of the nth machine.

A. Client Utility Matrix CU:

Any exercise to optimize the deployment of ATMs must start with a thorough understanding of the customer base and identification of the priority of the customers [10]. The generation of CU is made by following these procedures:

- The first step is to categorize people based on where they live, where they work and where they may need money in order to make payment for shopping and other transactions. The science of grouping of the people in a geographical area according to socioeconomic criteria is known as Geo-demography. The Commercial Geo-demography has been used to target ATM services to the Bank’s clients based on their lifestyle and location. In this study the geo-demographic approach is used by conducting a survey on potential Customer as well as geographical, demographic, economic, and traffic data. 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 implemented based on a priori analysis of all the applicable factors. Using SPSS program [23], the related data are entered. The variables used are Customers Age, income, Education and Marital Status which constitute the demographic and economic factors. The traffic data are represented by a variable such as the location importance which encompasses factors like number of residents, number of public institutes, number of private institutes and the state of street whether it is main street, by-street or crossroad. The procedure now is to compute the mean value of these variables for each customer then we segment the customers according to their areas and compute the cumulative mean value for customers belonging to each respective segment. Each cumulative mean value represents one element in G(x×y) matrix. The elements of G(x×y) range from 0 to 10. when The element g(x×y) is high means that there are more number of potential customers in that area, in contrast, when g(x×y) is small means less number of potential customer are there.

- Generate sub-matrices (CU_r), the matrix of CU_r is presented in equation 2 and Figure 1:

$$CU_r = G(x \times y) \times U(m \times n) \times 10 \tag{2}$$

$$CU_r = \begin{bmatrix} 23 & 47 & 23 \\ 47 & 70 & 47 \\ 23 & 47 & 23 \end{bmatrix}$$

Figure 1. CU_r Matrix.

Where: $r = 1, 2, \dots, \frac{(I \times J)}{(m \times n)}$

U(m×n) is the degradation of Client Utility, by assuming m =3, n=3, U(m×n) is given in Figure 2:

$$U(m \times n) = \begin{bmatrix} 1/m & 2/m & 1/m \\ 2/m & 1 & 2/m \\ 1/m & 2/m & 1/m \end{bmatrix}$$

Figure 2. Degradation of Client Utility matrix.

- CU matrix can be obtained by replacing each element in G(x×y) by its corresponding matrix CU_r as in Figure 3. The reason behind calculating CU_r is that, CU_r will be strongest at the center of the areas, and it will degrade as one moves away from it.

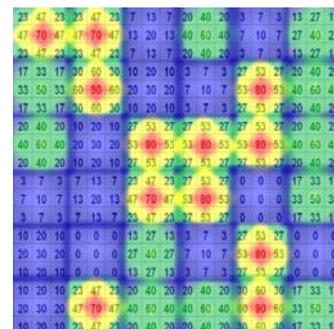


Figure 3. Client Utility Matrix CU.

B. Service Utility Matrix SU:

Once the deployment of ATMs is a one off project, hence it is done once. It is essential to distribute the limited number of ATMs in such a way as to maximize the utility of services. In order to find SU, this study assumes that ATMs are homogeneous, in line with this there exists only one matrix A. Matrix A (represents the degradation of ATMs utility as one moves away from its location) is predetermined and held constant for all machines.

The rectilinear distance model is adopted as shown in figure 4.

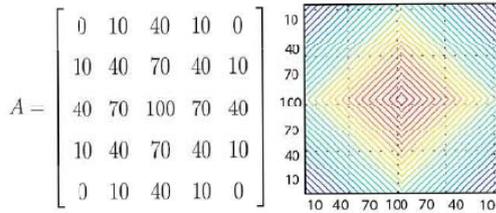


Figure 4. Service Matrix A (The rectilinear distance model)

The matrix L_n indicates the location of the nth machine. If this location is denoted by the coordinates (u_n, v_n) then all elements of L_n are equal to zero except for coordinates (u_n, v_n) where they are equal to one as in the equation 3 and figure 5.

$$L_n = \begin{cases} 1 & \text{at } (u_n, v_n); \\ 0 & \text{elsewhere.} \end{cases} \quad (3)$$

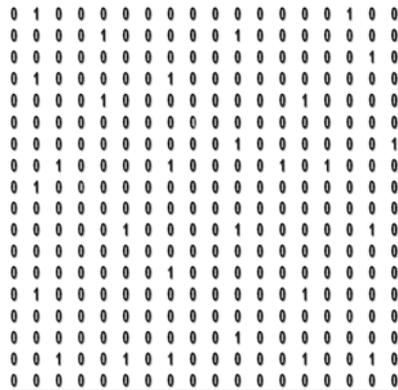


Figure 5. The Location Matrix L_n after deployment N machines.

The matrix SU can be obtained from the convolving of two matrices A and L as in equation 4 and figure 6. 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) . For the sake of illustration, figure 6, explain the convolution process.

$$SU = A * L \quad (4)$$

Where: the symbol * indicates the convolution product.

$$A = \begin{bmatrix} 30 & 50 & 30 \\ 50 & 100 & 50 \\ 30 & 50 & 30 \end{bmatrix} \quad \text{and} \quad LI = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Then

$$SUI = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 30 & 50 & 30 \\ 0 & 50 & 100 & 50 \\ 0 & 30 & 50 & 30 \end{bmatrix}$$

Figure 6. A Simple example on Convolution.

C. Percentage Coverage (PC):

In order to satisfy the client, his Utility should be satisfied by covering his demand, and the Service Utility should be maximized through effective deployment of ATMs, this will save the cost of providing additional ATM. PC is computed as the percentage of ψ (ψ is equal to one in all points in E that have SU greater than CU) divided by the number of elements in E. PC is given as in equation (5):

$$PC = \frac{\sum_{i=1}^I \sum_{j=1}^J (\psi \times 100)}{I \times J} \quad (5)$$

where ψ is given in equation (6):

$$\psi = \begin{cases} 1 & \text{if } E(i, j) \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

In addition to PC, another important measure (the total Client Utility satisfied γ) is calculated. The formula of γ is given in equation (7):

$$PC = \frac{\sum_{i=1}^I \sum_{j=1}^J |B(i, j)|}{\sum_{i=1}^I \sum_{j=1}^J |CU(i, j)|} \quad (7)$$

Where:

$$B(i, j) = \begin{cases} CU(i, j) & \text{if } E(i, j) > 0; \\ Q(i, j) & \text{otherwise.} \end{cases} \quad (8)$$

The algorithm returns both γ and PC values with the solution as will be shown in the simulations section. PC and γ values are essential in measuring the goodness of deployment of ATMs. The value of g ranges between [0, 1] and it approaches one only when all elements in E are zeros or positive values, denoting the saturation level of Client Utility. In order to deploy less number of ATMs without affecting negatively on PC and γ , this study uses trial method such that, the trial starts with number of ATMs which can be determined from the HAC output and then run RGAC or SAC to compute the best PC and γ . If the value of PC is equal to hundred (100), then next step the number of ATMs is reduced by one, the trial continues reducing N as long as PC is within the acceptable limit (i.e. more than the lower limit 99). Otherwise when the value of PC is less than the acceptable limit, then trial increases number of ATMs till PC reaches the acceptable limit. The previous conditions are presented in equation 9:

$$\text{Deploy } (M) \text{ ATMs if } \begin{cases} PC_{(M=N-K)} = 100 \\ \text{Or} \\ |PC_{(M=N+K)} < 99 \end{cases} \quad (9)$$

Where: $k = 1, 2, \dots$

V. HEURISTIC APPROACH ALGORITHM FOR SOLVING THE BANKING ATM'S LOCATION PROBLEM (HAC)

This approach turns out to offer high flexibility in choosing arbitrary service and demand patterns [4]. It also allows a simple human user interface modeling of the problem and provides the solution in relatively short time. The solution approach is described as follows; first, the matrix A is the same in all algorithms and the demand matrix D used in HAC is similar to CU matrix used in RGAC and SAC 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, of 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 next machine. The algorithm terminates when all the elements of E exceed the service margin Θ or when the overall percentage coverage is satisfactory.

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.,

$$C_n = A * E_{n-1}, \quad E_0 = -D \quad (10)$$

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 in 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 nth machine, i.e., equation 11.

$$(u_n, v_n) = \operatorname{argmin}_{(i,j)} C_n \quad (11)$$

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 service levels 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 choose 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 (similar to CU in later discussion) can be designed to fulfill any arbitrary demand patterns. Once a new machine location is computed, the location matrix L_n is constructed from (3). The difference matrix is then updated as in equation 12.

$$E_n = Q_n - D \quad (12)$$

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 13.

$$Q_n(I, j) = \max \{Q_{n-1}(I, j), S_{u_n, v_n}(I, j)\}, \quad Q_0 = 0. \quad (13)$$

Where 0 is the (I, J) zero matrix.

A flow chart of the proposed algorithm is shown in Figure. 7.

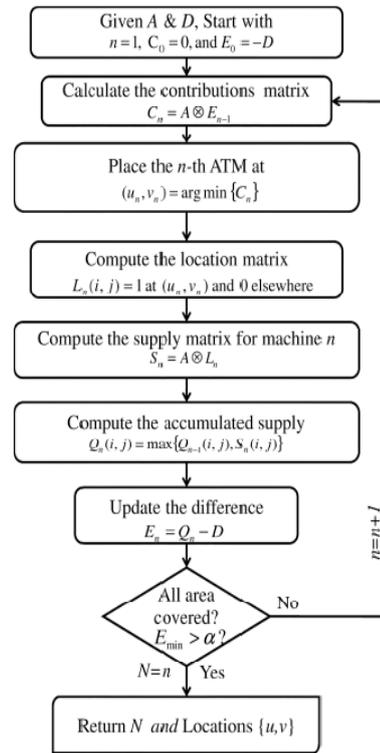


Figure 7. The HAC Algorithm.

But even though HAC is an effective algorithm as discussed above, it has some shortcomings which affect on its performance in solving the ATM's deployment problem. The major drawbacks accounted for by HAC are as follow:

- HAC binds ATMs sequentially.
- HAC uses arbitrary numbers for calculating the demand matrix D.
- HAC assigns machines at the boundaries of the targeted area.

VI. RANK BASED GENETIC ALGORITHM FOR SOLVING THE BANKING ATM'S LOCATION PROBLEM (RGAC)

GA is used to solve optimization problems by imitating the genetic process of biological organisms [24]. A potential solution to a specific problem may be represented as a chromosome containing a series of genes. A set of chromosomes makes up the population. By using Selection, Crossover and Mutation Operators, GA is able to evolve the population to generate an optimal solution. The parameters of RGAC are listed in table II.

TABLE II
PARAMETERS USED IN THIS STUDY

Parameters of RGAC algorithm	Value
Crossover Rate	1
Mutation Rate	1
Maximum Generations	700
Population Size	$(I \times J \times 2)$
Stopping Criterion	If 100 iterations of RGAC are not showed improvement in PC value Or PC value equals to hundred (100)

A. Chromosome Representation

The efficiency of GA depends largely on the representation of a chromosome which is composed of a series of genes. In this paper, each gene represents an ATM location which is equal to one or zero based on binding of the ATM to its location as in equation 3. As a result, L represents the chromosome. Population Initialization is generated randomly.

B. Fitness Equation

A fitness equation must be devised to determine the quality of a given chromosome and always returns a single numerical value. In determining the fitness equation, it is necessary to maximize the Percentage Coverage PC of CU. RGAC takes the PC value as a fitness equation for a given chromosome, which presented in equation 5.

C. Evolutionary Process

Evolutionary process is accomplished by applying Rank based Roulette Wheel Selection (RRWS) [25], [26]. Crossover and mutation operate from one generation to the next. Selection Operator determines how many and which individuals will be kept in the next generation. Crossover Operator controls how to exchange genes between individuals, while the Mutation Operator allows for random gene alteration of an individual. Besides the standard genetic operators discussed above, the Elitism Phase is used to preserve the best candidates. These stages are discussed in details as below. Firstly, in order to carry out the RRWS, the Relative Probability (shown in equation 14) and cumulative proportion of each chromosome are calculated.

$$P_i = \frac{\text{Rank}_{(\text{fitness})}}{\sum \text{Rank}_{(\text{fitness})}} \quad (14)$$

After that, one-Point Crossover and Mutation Operators, the algorithms (1, 2) are applied to the chromosomes from the selection phase.

Mutation Operator runs through the genes in each of the chromosomes and mutates each gene according to a Mutation Rate P_m . Finally, Elitism combines the parent population with the modified population (the candidates generated by Crossover and Mutation Operators), and takes the best chromosomes to the next generation. The purpose of this phase is to preserve the best chromosomes from being lost. After this phase, the algorithm continues to the next iteration. RGAC is presented in the algorithm 3.

Algorithm 1 One point Crossover

```

1: for i=1 to popSize/2 do
2:   Select two chromosomes p1, p2 randomly
3:   Select Crossover point Point randomly
4:   Save coordinates of ATM locations of two chromosomes p1, p2 in row1, col1, row2, col2.
5:   if random[0,1] < probCrossover then
6:     for k=Point+1:ChromosomeLength do
7:       Swap the coordinates of ATM locations of two chromosomes p1, p2
8:     end for
9:     Keep the newly produced chromosomes as candidates
10:  end if
11: end for

```

Algorithm 2 Mutation

```

1: for i=1 to popSize do
2:   Select chromosome p randomly
3:   Save coordinates of ATM locations of chromosome p in row, col
4:   if random[0,1] < probMutation then
5:     Select one ATM location of chromosome p randomly and make it equals to zero
6:     Generate one ATM location of chromosome p randomly and make it equals to one
7:     add the newly produced chromosomes as candidates
8:   end if
9: end for

```

Algorithm 3 RGAC

```

1: Generate Initial Population P of size NI randomly.
2: Evaluate each chromosome using equations (1, 5, and 7) }
3: for g = 1 to MaximumGenerations do
4:   Generate offspring Population from P
5:   {Rank based Roulette Wheel Selection
6:   Crossover and Mutation (algorithms 1, and 2 )
7:   Evaluate each chromosome resulting from Crossover and Mutation stages using equations (1, 5, and 7) }
8: (Elitist) Select the members of the combined population based on maximum fitness, to make the population P of the next generation.
9: Evaluate each chromosome using equations (1, 5, and 7) }
10: if Stopping criterion has reached then
11:   return PC, γ values and best ATM Location Matrix L found;
12:   break;
13: end if
14: end for

```

D. Performance Analysis

RGAC needs to execute some hundreds of iterations to come up with an optimal solution. However, the shortcoming of HAC is convergence to a local optimum. According to the simulation results, it is proved that RGAC is effective in speeding up convergence while meeting a feasible result. Also RGAC outperforms HAC, in the PC and g values to obtain the final schedule.

VII. SIMULATED ANNEALING ALGORITHM FOR SOLVING THE BANKING ATM'S LOCATION PROBLEM USING CONVOLUTION (SAC)

The idea of Simulated Annealing methodology is motivated by the physical process of annealing in metallurgy. In an annealing process, a solid is heated to a high temperature and gradually cooled in order for it to crystallize. At high temperatures the atoms move randomly and at high kinetic energy, but as they are slowly cooled, they tend to align themselves in order to reach the minimum energy state. Initial state of a thermodynamic system was chosen at energy E and temperature T , the initial configuration is performed and the change of energy ΔE is computed. The current state of the thermodynamic system is analogous to the current solution to the combinatorial problem, the energy equation for the thermodynamic system is analogous to the objective function and ground state is analogous to the global minima [27].

A. Generic choices:

These choices have been made for implementation of SAC.

- The initial value of the temperature parameter T is chosen to be 1000.
- A temperature function, $T(t)$, is used to determine how the temperature will be lowered at each iteration over the course of the algorithm, It has a major impact on convergence rate and solution quality. On one hand, if the temperature is decreased quickly, then the algorithm converges fast, but final solutions will tend to get worse. On the other hand, slow cooling will make the algorithm slow but give better results. For this, slow cooling option has been chosen in this study in order to obtain good solutions, the used rule is the geometric one as in equation (15):

$$T(n+1) = \alpha * T(n) \quad (15)$$

Where: $\alpha = 0.99$.

- The number of iterations, $N(t)$, to be performed at each temperature is taken to be 100.
- Stopping criterion: the SAC algorithm will be terminated after $(I \times J)^m$ iterations depending on problem's nature.

B. The acceptance criterion:

The algorithm works iteratively keeping a single tentative solution at any time. In every iteration, a new solution is generated from the previous one, and either replaces it or not depending on an acceptance criterion. The acceptance criterion works as follows: both the old and the new solutions have an associated quality value, determined by a fitness function (PC value), if the new solution is better than the old one, then it will replace it. If it is worse, it replaces it with probability P . This probability depends on the difference between their quality values and a control parameter T named temperature. This acceptance criterion provides a way of escaping from local minima [28]. The SAC algorithm is described in algorithm 4.

C. Fitness Equation:

As in section VI, B. Both RGAC and SAC take the PC value as a fitness equation for a given chromosome as presented in equation 5.

Algorithm 4 Simulated Annealing using Convolution

```

1: Choose an initial chromosome  $s \in S$  randomly
2:  $i = 0$ ,  $s^* = s$ ,  $T_{initial} = 1000$ ,  $best = PC(s)$ 
3:  $t_i = T_{initial}$ 
4: repeat
5:   Generate randomly a chromosome  $s^+ \in Neighbor(s)$ 
6:   if  $PC(s^+) > best$  then
7:      $s^* = s^+$ 
8:      $best = PC(s^*)$ 
9:   else
10:    if  $random[0,1] < \exp(\frac{PC(s^+) - PC(s)}{t_i})$  then
11:       $s = s^+$ 
12:    end if
13:  end if
14:   $t_{i+1} = t_i \times 0.99$ ;
15: until Stopping criterion is reached

```

VIII. SIMULATION RESULTS

This section shows the experiments that have been carried out in order to evaluate the proposed algorithms (HAC, RGAC and SAC) which are simulated using MATLAB with Intel P4 2.2GHz CPU, 2GB memory and Linux operating system. They are focused on testing how the different proposed algorithms perform, and making comparisons between them.

Experiment-I has been carried out using the same number of ATMs ($N1$) in all algorithms to evaluate and compare PC values, i.e., satisfying the first objective function. Experiment-II has been made by using the highest number of ATMs ($N2$) in HAC, and by comparing $N2$ with its corresponding $N1$ used in SAC to measure saving in ATMs number. The purpose behind conducting Experiment-III is to investigate the ability of SAC in saving ATM machines over HAC and RGAC. To gain insight into the quality of the solutions, Table III provides detailed comparisons applied to different groups of CU matrices: (12×12) , (18×18) , (24×24) , and (30×30) , respectively. The reason behind testing grouping is to explore the performance of algorithms when the size of the market is extended. Table III and figures (8, 9, and 10) show the following:

- Experiment-I: $N1$ is used as the number of ATMs in all algorithms. Table III and figure 8 depict that: Firstly, (RGAC vs. HAC), RGAC shows a significant improvement in PC over HAC, recording minimum improvement of 2.2%, CU(18×18) sample 4, and maximum improvement of 20.13%, CU(12×12) sample 3. Secondly, (SAC vs. HAC), SAC shows a much better improvement in PC over HAC, recording minimum improvement of 2.778%, CU(12×12) sample 4, and maximum improvement of 20.83%, CU(12×12) sample 3. Thirdly, (SAC vs. RGAC), SAC performs more efficiently than RGAC in term of PC, recording no improvement, CU(12×12) sample 1&2&4&5 and CU(18×18) sample 1, and maximum improvement of 2.222%, CU(30×30) Sample 1. Same analytical framework can be made for γ .

- Experiment-II: the strategy here is to add more ATMs as long as the PC value for HAC is increasing and stop adding extra ATM when PC gets stuck at a constant value then we denote (N2) to the highest number of ATMs used by HAC, thereafter a comparison has been made between N1 and N2 to measure the Saving (N2-N1). One required condition to be satisfied states that the reduction in the number of ATMs is considered saving when the PC values for SAC at (N1) are higher than them for HAC at (N2). Table III and figure 9 show that the Saving (N2-N1) ranges between 2 ATMs CU (24×24) sample 5, up to 33 ATMs CU(24×24) sample 1.
- Experiment-III compares between two Global Search

Techniques namely RGAC and SAC and judges which one is more suitable for ATMs Deployment problem. The strategy used is to reduce N1 by one and monitor PC value for SAC, as long as SAC's PC value at $N3=N1-1$ is higher than its value for RGAC at N1, the procedure here is to apply more reduction on N3, till $N3=N1-k$ used in SAC, gives zero or near zero difference in PC value between two algorithms then, we accept the reduction (k). If the reduction by one machine gives negative difference in PC values between SAC and RGAC, we can say there is no saving being made. Table III and figure 10 show that there is no acceptable saving in the number of ATMs regarding CU(12×12) all samples, CU(18×18) all samples and CU(24×24) sample 1, because the difference $PC5-PC2$ is negative, figure 5, the area to the right of 0.0 on SAC(pc)-RGAC(pc) axis. The remaining samples, CU (24×24) sample 2&3&4&5, the reduction by one ATM is acceptable. Coming to CU (30×30) more reductions in the number of ATMs can be made, it ranges from 2 in CU (30×30) sample 3 up to 6 in CU (30×30) sample 1.

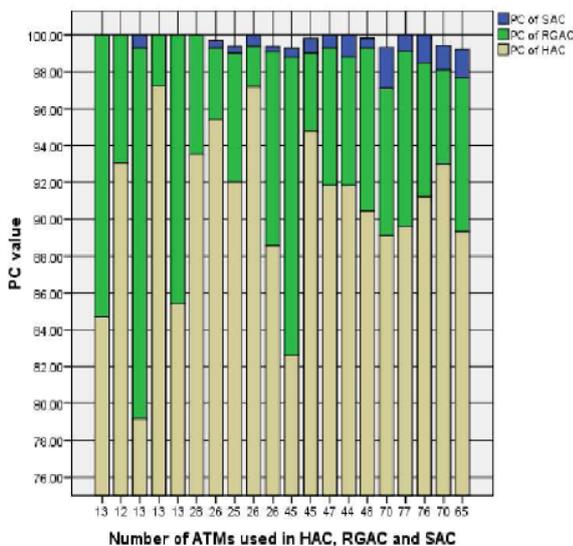


Figure 8. Experiment-I: The improvement in PC value HAC vs. RGAC vs.SAC using same Number of ATMs.

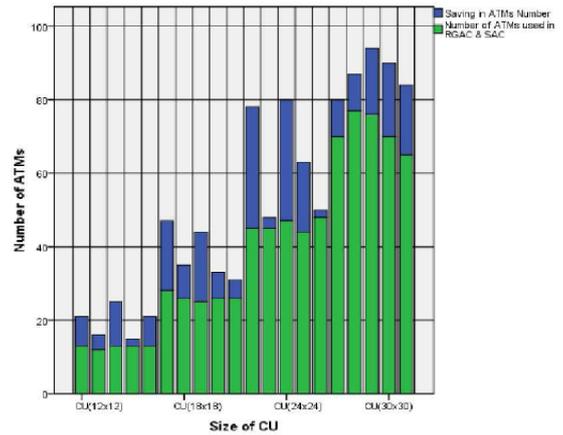


Figure 9. Experiment-II: The Reduction in ATMs Number (RGAC& SAC) vs. HAC.

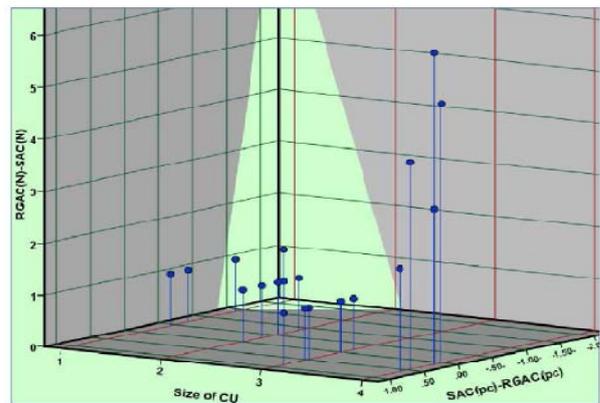


Figure 10. Experiment-III: The Reduction in ATMs Number (SAC vs. RGAC).

IX. CONCLUSION AND FUTURE WORK

In this paper, ATM deployment problem has been solved using different techniques HAC, RGAC and SAC. A detailed comparison between the previous techniques was made by executing three experiments and four different market sizes represented by CU matrices. HAC is an effective Algorithm in Small ATM market but it appears to be deficient as the market size becomes larger, and it usually falls in local minima in finding the optimal solution.

RGAC shows a significant improvement in PC over HAC, recording minimum improvement of 2.2% and maximum improvement of 20.13%.

SAC algorithm is found to be scalable, highly effective and efficient as the size of the market becomes larger. The experimental results have demonstrated the feasibility of SAC to the ATM deployment problem by maintaining PC values over 99 in all samples and significant reducing the required ATMs up to 6 compared with RGAC.

One possible direction of the future work would be the consideration of both demand side represented by potential market for ATM service, and supply side represented by the cost of providing this service in order to find an optimization model that takes into account this consideration.

REFERENCES:

[1] Chin S. Ou , David C. Yen , Chia-Sheng Hung . "Determinants of information technology investments: The case of ATM in an emerging

economy” Advances in Accounting, incorporating Advances in C-S. OU, HIN YUAN HUNG, D. C. YEN, AND F-C. LIU, “CAN AUTOMATIC TELLER MACHINE INVESTMENT IMPROVE BANK COST EFFICIENCY” 2006.

[3] S. HABIB, “SIMULATED ANALYSIS OF SERVER PLACEMENT ON NETWORK TOPOLOGY DESIGNS,” OCTOBER 2005.

[4] M. A. ALDAJANI AND H. K. ALFARES, “LOCATION OF BANKING AUTOMATIC TELLER MACHINES BASED ON CONVOLUTION” . COMPUT. IND. ENG., VOL. 57, NO. 4, PP. 1194-1201, 2009.

[5] A. ALhaffa, W. Abdual, O. A. JADAAN, AND A. JABAS, “RANK BASED GENETIC ALGORITHM FOR SOLVING THE BANKING ATMS LOCATION PROBLEM USING CONVOLUTION,” MARCH 2011, ACCEPTED.

[6] A. ALhaffa, W. Abdual, “Simulated Annealing Algorithm for Solving the Banking ATM’s Location Problem Using Convolution” IEEE Transl. S. Thatcher, vol. 2, Feb.2011, pp. 331-336 [International Conference on Machine Learning and computing, Singapore, 2011].

[7] Philips GmbH, “Constructing efficient simulated annealing algorithms” Discrete Applied Mathematics 77. Aachen, Germany July, 1996. [Elsevier, 1997].

[8] El-Haddan, A., & Almahmeed, M. (1992). “ATM banking behavior in Kuwait: A consumer survey”. International Journal of Bank Marketing, 10(3), 250-232.

[9] Marshall, J., & Heslop, L. (1988). “Technology acceptance in Canadian retail banking”. International Journal of Bank Marketing, 6(4), 31-41.

[10] Swinyard, W. R., & Ghee, L. (1987). “Adoption patterns of new banking technology in Southeast Asia”. International Journal of Bank Marketing, 5(4), 35-48.

[11] William B. Trautman. “A Framework for Regulating Automated Teller Machine Technology”. Journal of Policy Analysis and Management, Vol. 12, No. 2 (Spring, 1993), pp. 344-358.

[12] Ranković, A., Marko; and Vasković, R., Vojkan. “The Economic Models for the ATM Network Implementation”. Belgrade, Serbia. Dec, 2008.

[13] Parasuraman A, Berry L L and Zeithaml V A (1985), “A Conceptual Model of SQ and Its Implications for Future Research”, Journal of Marketing, Vol. 49, Fall, pp. 41-50.

International Accounting 25 (2009) 278-283.[Elsevier, 2009].

[14] Blanchard R F and Galloway R L (1994), “Quality in Retail Banking”, International Journal of Service Industry Management, Vol. 5, No. 4, pp. 5-23.

[15] Donner, S., & Dudley, C. (1997). “Balancing customer contact and high-tech delivery”. [Electronic version]. American Bankers Association. ABA Banking Journal, 89(1), 18-20.

[16] BERMAN, O. & PARKAN, C. (1984). “Sequential facility location with distance-dependent demand”. J.Oper. Manage., 3, 261-268.

[17] Mohammad Jafrollah, Srinivas Uppuluri, Dr. Nagesh Rajopadhaye, & V. Srinatha Reddy. “An Integrated approach for Banking GIS, Business GIS, Map India, 2003”.

[18] A. Qadrei and S. Habib, “Allocation of heterogeneous banks’ automated teller machines,” pp. 16-21, 2009.

[19] S. Alrashed, P. N. Marimuthu, and S. J. Habib, “Optimal deployment of actors using simulated annealing within wsan,” April 2010.

[20] G. Molina and E. Alba, “Wireless sensor network deployment using a memetic simulated annealing,” July 2008.

[21] Heli Snellman, “Automated Teller Machine network market structure and cash usage, scientific monographs”. Helsinki 2006.

[22] Jocelyn Donze and Isabelle Dubecy, “ATM Direct Charging Reform: the Effect of Independent Deployers on Welfare”, June 9, 2010.

[23] SPSS, “http://www.spss.com.”.

[24] D. E. Goldberg, “Genetic Algorithms in Search Optimization and Machine Learning”. New York, NY: Addison-Wesley, 1989.

[25] W. Abdual, O. A. Jadaan, A. Jabas, and S. Ramachandram, “Genetic algorithm for grid scheduling using best rank power,” in Nature & Biologically Inspired Computing, NaBIC 2009. IEEE, 2009, pp. 181-186.

[26] A. J. S. R. Wael Abdual, Omar Al Jadaan, “Rank based genetic scheduler for grid computing systems,” in The International Conference on Computational Intelligence and Communication Networks (CICN 2010). IEEE, 2010.

[27] S. Fidanova, “Simulated annealing for grid scheduling problem,” 2006.

[28] R. Eglese, “Simulated annealing: A tool for operational research,” Euro-pean Journal of operational Research 46, Holland, 1990.

TABLE III SIMULATION RESULTS

Algorithm	Experiment I												Experiment II				Experiment III							
	HAC			RGAC			SAC			HAC/RGAC			HAC/SAC		RGAC/SAC		HAC		N Save		SAC			RGAC/SAC
Parameter	N1	PC	γ	PC	γ	PC	γ	ΔPC	Δγ	ΔPC	Δγ	ΔPC	Δγ	N2	PC	γ	N2 - N1	N3	PC	γ	ΔPC	N3 - N1		
SampleNo	CU(12 × 12)																							
1	13	84.7	95	100	100	100	100	15.3	5	15.3	5	0	0	21	94.44	98.18	8	12	97.92	99.2	-2.08	1		
2	12	93.06	97.64	100	100	100	100	6.94	2.36	6.94	2.36	0	0	16	97.22	99.56	4	11	98.61	100	-1.39	1		
3	13	79.17	89.06	99.3	99.8	100	100	20.13	10.74	20.83	10.94	0.7	0.2	25	90.28	95.22	12	12	97.92	97.31	-1.38	1		
4	13	97.22	98.06	100	100	100	100	2.78	1.94	2.78	1.94	0	0	15	100	100	2	12	99.31	100	-0.69	1		
5	13	85.42	90.92	100	100	100	100	14.58	9.08	14.58	9.08	0	0	21	93.06	95.53	8	12	99.57	100	-0.43	1		
SampleNo	CU(18 × 18)																							
1	28	93.52	97.41	100	100	100	100	6.48	2.59	6.48	2.59	0	0	47	99.07	99.62	19	27	99.38	100	-0.62	1		
2	26	95.4	99	99.3	100	99.7	100	3.9	1	4.3	1	0.4	0	35	98.77	99.7	9	25	98.77	98.6	-0.53	1		
3	25	92	87	99	100	99.4	99	7	13	7.4	12	0.4	-1	44	96.91	97.51	19	24	98.15	98.85	-0.85	1		
4	26	97.2	98.1	99.4	100	100	100	2.2	1.9	2.8	1.9	0.6	0	33	100	100	7	25	99.38	99.47	-0.02	1		
5	26	88.58	96.05	99.07	99.36	99.38	99.84	10.49	3.31	10.8	3.79	0.31	0.48	31	90.12	96.73	5	25	98.77	99.04	-0.31	1		
SampleNo	CU(24 × 24)																							
1	45	82.6	92.2	98.78	99.4	99.3	99.5	16.18	7.2	16.7	7.3	0.52	0.1	78	92.88	97.17	33	44	98.61	99.06	-0.17	1		
2	45	94.79	98.1	99	99.5	99.8	100	4.21	1.4	5.01	1.9	0.8	0.5	48	96.18	98.66	3	44	99.48	100	0.48	1		
3	47	91.84	96.3	99.3	99.5	100	100	7.46	3.2	8.16	3.7	0.7	0.5	80	98.09	99.03	33	46	99.31	99	0.01	1		
4	44	91.84	95.66	98.8	99.5	100	100	6.96	3.84	8.16	4.34	1.2	0.5	63	96.35	98.2	19	43	99.65	100	0.85	1		
5	48	90.45	96.23	99.3	98.87	99.83	99.57	8.85	2.64	9.38	3.34	0.53	0.7	50	91.49	96.64	2	47	99.83	100	0.53	1		
SampleNo	CU(30 × 30)																							
1	70	89.11	94.48	97.11	98.18	99.33	99.49	8	3.7	10.22	5.01	2.22	1.31	80	94.67	97.52	10	64	97.22	97.51	0.11	6		
2	77	89.6	95	99.1	99.6	100	100	9.5	4.6	10.4	5	0.9	0.4	87	92	96.34	10	73	99.56	100	0.46	4		
3	76	91.22	96.38	98.5	99	100	100	7.28	2.62	8.78	3.62	1.5	1	94	94.56	97.93	18	74	99.11	99.32	0.61	2		
4	70	93	97.35	98.11	98.8	99.44	99.46	5.11	1.45	6.44	2.11	1.33	0.66	90	96.89	98.95	20	65	98.11	98.8	0	5		
5	65	89.33	94.46	97.67	98.74	99.22	99.53	8.33	4.28	9.89	5.07	1.56	6	84	94.89	97.85	19	62	97.78	98.35	0.11	3		