

# Location Area Planning and Cell-to-Switch Assignment in Cellular Networks

İlker Demirkol, Cem Ersoy, M. Ufuk Çağlayan, Hakan Deliç

İ. Demirkol, C. Ersoy and M. U. Çağlayan are with NETLAB, Department of Computer Engineering, Boğaziçi University, Bebek 34342 Istanbul, Turkey. E-mail: {ilker, ersoy, caglayan}@boun.edu.tr.

H. Deliç is with BUSIM Laboratory, Department of Electrical and Electronics Engineering, Boğaziçi University, Bebek 34342 Istanbul, Turkey. E-mail: delic@boun.edu.tr.

This work was supported in part by the State Planning Organization of Turkey under the grant number 98K120890, and Turkcell Inc. This work was presented in part at IEEE INFOCOM 2001, Anchorage, Alaska, USA, April 2001.

### Abstract

Location area (LA) planning plays an important role in cellular networks because of the trade-off caused by paging and registration signalling. The upper boundary for the size of an LA is the service area of a Mobile services Switching Center (MSC). In that extreme case, the cost of paging is at its maximum but no registration is needed. On the other hand, if each cell is an LA, the paging cost is minimal but the cost of registration is the largest. Between these extremes lie one or more partitions of the MSC service area that minimize the total cost of paging and registration. In this paper, we seek to determine the location areas in an optimum fashion. Cell to switch assignments are also determined to achieve the minimization of the network cost. For that purpose, we use the available network information to formulate a realistic optimization problem, and propose an algorithm based on simulated annealing (SA) for its solution. Then, we investigate the quality of the SA-based technique by comparing it to greedy search, random generation methods and a heuristic algorithm.

### Index Terms

Location Area, Location Management, Location Tracking, Location Update, Paging Area, Simulated Annealing.

## I. INTRODUCTION

Mobility tracking is crucial for the wireless networks to achieve the quality of service (QoS) objectives [1]. In cellular communications, upon the arrival of a mobile-terminated call, the system tries to find the mobile terminal by searching for it among a set of base stations (BSs) over the current region knowledge of the mobile. This search is called paging, and the set of base stations in which a mobile is paged is called the Location Area (LA). At each LA boundary crossing, mobile terminals register their new locations through signaling in order to update the location management databases. Therefore, the size of the LA is important for reducing the cost of paging and registration signalling [2]. Although there may be events other than location updates that cause registration, we will use the term “registration” instead of “location update” throughout this paper.

The cost of paging or registration has three distinct sources: (i) the *select* or *update* queries, which create processing and storage load on the database elements of the cellular network; (ii) the load created on the wired connection part of the cellular network; (iii) the load on the air interface. Among these, the least scalable one (that is, the most affected resource as the population and traffic in a cell grow) is the radio bandwidth resource. Moreover, because the

radio bandwidth is shared by control functions such as paging, registration and call traffic, it is regarded as the scarcest resource in a wireless network. Hence, we may just consider the paging and registration load on the radio bandwidth.

A particular LA management approach is classified according to its use of zone, time, movement, distance or profile information in its location update and paging procedures. In zone-based location updating, employed by the Global System for Mobile Communications (GSM) standard, LA boundaries are fixed and the same boundaries are used for all mobile registrations. Optimal location area planning in zone-based schemes is not widely studied. Saraydar *et al.* investigate zone-based schemes by aiming to optimize the LA borders, where they either suppose a one-dimensional service area [2], or Poisson-distributed paging load [3]. The model and simulations in our work rely on actual data obtained from a national cellular service provider so that no assumptions are necessary regarding mobility or call arrivals. In [4], zone-based scheme with selective paging is investigated, and a new location-tracking algorithm, namely *tree-location area algorithm* (TrLA), is proposed. Ozugur [5] defines a multiobjective zone-based LA and routing area (RA) problem for General Packet Radio Service (GPRS) and Universal Mobile Telecommunications System (UMTS) standards, and applies a genetic algorithm formulation. In [6], LA partitioning is considered as a graph optimization problem.

The location update can be performed according to the time elapsed since the last update [7]-[9]. The implementation of the so-called time-based location update schemes are simple. However, redundant signalling may be incurred when the system has low mobility because of numerous stationary mobiles. In movement-based LA management [10]-[13], the location update decision depends on the number of cell boundary crossings measured since the previous location update. Distance-based schemes state that location update will be carried out when a mobile terminal moves a threshold number of cells away from the cell in which the last location update was performed [14]-[17]. A hybrid of zone-based and distance-based scheme is studied in [18]. The distance-based location update scheme produces the minimum signalling cost compared to movement- and time-based techniques [19]. However, its implementation is hard since the distance of the mobile has to be computed dynamically as it moves from cell to cell. In the profile-based scheme, the location update is performed according to some specific criterion such as the mobility behavior of the mobile or the distribution of the mobile terminated call arrivals

[20]-[23].

In this paper, we propose a solution for zone-based location update and paging schemes, since they are used in the widely deployed GSM systems. Most LA planning problems in the literature have a common objective: minimizing the total paging and registration costs, where the objective function requires the addition of the two distinct cost types. To that end, certain assumptions are made for the relative values of these costs; e.g., assuming one unit cost for each paging event is equal to 10 unit cost for each registration event. Such assumptions have the deficiency of being the same throughout the whole network, whereas the load (i.e., cost) of paging and registration varies from cell to cell. Moreover, these costs do not have comparable units, for different control channels are used for paging and registration. Hence, a linear combination of these costs is not very meaningful since the operators cannot specify the relative weight of each cost. In literature, the cell-to-switch assignments are often assumed to be known, and only cell-to-LA assignments are considered. In this paper, we generalize the LA planning problem by including the cell-to-switch assignment decisions. Hence, we define and formulate a problem in which both cell-to-switch and cell-to-LA assignments are jointly investigated to minimize the cellular network cost. Subsequently, a simulated annealing (SA) based solution is developed for the generalized cellular network structuring problem. Although various algorithms are extensively applied to LA planning in literature, SA-based solutions have not been studied for cell-to-LA or cell-to-switch assignment.

In order to prohibit linear combining of incomparable costs, the paging cost is included in the problem formulation as a constraint. The registration cost is left alone in the objective function, which still results in a difficult combinatorial optimization problem. However, compared to minimizing the paging within acceptable registration capacities, this goal is advantageous since paging capacity is easier to quantify as a constraint as explained in Section II. We propose a simulated annealing algorithm and a heuristic approach for the solution of the problem in Section III, and present the results of the computational experiments in Section IV. Section V concludes the paper.

## II. THE LOCATION AREA PLANNING PROBLEM

Without loss of generality, we consider a model that complies with the GSM hierarchy, which consists of Base Stations (BSs), Base Station Controllers (BSCs) and Mobile Switching Centers

(MSCs) [24]. While formulating the LA planning problem, we only use the available network information and try to include all realistic constraints and goals. The overall problem in LA planning and dimensioning stems from the trade-off between the paging cost and the registration cost. The paging cost is a result of the arriving calls to mobiles (i.e., mobile-terminated calls). The called party has to be searched and located in order to establish the connection. In order to have a feasible cellular network, the paging capacities of BS  $i$  and BSC  $j$  (that have the unit *paging commands per second* and denoted by  $P_i^{BS}$  and  $P_j^{BSC}$ , respectively) should not be exceeded. For evaluation of the paging loads, the paging rate for each cell  $i$ ,  $\lambda_i$ , which just includes the paging generated because of the mobiles residing in that cell, is used.

The maximum call traffic load capacities of BSC  $j$  and MSC  $k$  ( $C_j^{BSC}$  and  $C_k^{MSC}$ , respectively) impose constraints that are related to the processing capabilities of the equipment. These constraints are checked through the busy-hour call traffic of each cell, which is  $c_i$  for cell  $i$  and which has the unit *Erlang-hours*.

In order to have a feasible cellular network, the limited call processing capability of MSCs and BSCs may create a limit on the peak call arrival rate (this call arrival rate includes not only the established connections but also the failed attempts), known as the Busy Hour Call Attempt (BHCA) capacity. The BHCA capacities for BSC  $j$  and MSC  $k$  are represented as  $D_j^{BSC}$  and  $D_k^{MSC}$ , respectively. The BHCA load on BSCs and MSCs are calculated by using the call attempt rate of the cells which has the unit *call attempts per unit time*. Let  $d_i$  denote the peak call attempt rate of cell  $i$  per unit time.

Each BS has a finite number of transmitters (TRXs), which defines the number of channels in that cell. The TRX capacity constraint is vendor-specific for BSCs and MSCs. For any BSC  $j$  and MSC  $k$ , the sum of the number of TRXs for each cell connected to that BSC or MSC must comply with a limit, denoted as  $R_j^{BSC}$  and  $R_k^{MSC}$ , respectively. To check for this constraint, we need to know the number  $r_i$  of TRXs in cell  $i$ , for all  $i$ .

The registration cost has several components including usage of radio resources, database updates. Normally, the registration cost is directly proportional to both idle-state cell boundary crossings and active-state handovers between cells at LA boundaries. The calculation of these costs requires the collection of the flow rates at all cell boundaries while mobile terminals are idle (i.e., with hand-set on but not in conversation). Since this is infeasible, cellular network operators (including the four in Turkey) cannot collect idle-state statistics. On the other hand, operators

collect handover data for the mobiles passing cell boundaries during conversation. Since our formulation relies strictly on available data, the aggregate mobile flow behavior is approximated by assuming that the idle mobile flow rate between any given two cells is proportional to the active mobile flow rate (i.e., the handover rate) between these cells. In other words, the handover traffic from the  $i$ th cell to  $s$ th cell,  $h_{is}$ , is used instead of the mobile flow data which has the unit *handovers per unit time*.

To propose a BS-BSC topology, we must know which BS can be connected to which BSC and discard those BS-BSC pairs that are not feasible (due to physical limitations, geographical constraints, etc.). This information is available in the form of a proximity matrix,  $\mathbf{A}$ , among BSs and BSCs in which  $a_{ij} = 1$  if  $BS_i$  can be connected to  $BSC_j$ , and  $a_{ij} = 0$  otherwise. Similarly, for the BSC-MSB topology, we will use a proximity matrix,  $\mathbf{B}$ , among BSCs and MSBs where  $b_{jk} = 1$  if  $BSC_j$  may be connected to  $MSB_k$ , and  $b_{jk} = 0$  otherwise. Because, part of the network structure may already be established with connections that are not allowed to be changed. Hence, using the proximity matrix, existing physical infrastructure can be taken into account in the resulting network solutions.

The following design variables will also be used in the formulation:

- 1) LA-BS topology is represented by the matrix  $[\mathbf{L}]_{in} = \ell_{in}$ . If the  $i$ th BS resides in the  $n$ th LA, then  $\ell_{in} = 1$ , and  $\ell_{in} = 0$  otherwise.
- 2) The  $\mathbf{L}$  matrix is used to establish the matrix  $[\mathbf{D}]_{is} = d_{is}$ , where  $d_{is} = 1$  if the  $i$ th and  $s$ th BSs reside in different LAs, and  $d_{is} = 0$  otherwise.
- 3) The BSC-MSB topology is represented by the matrix  $[\mathbf{Y}]_{jk} = y_{jk}$ . If the  $j$ th BSC is connected to the  $k$ th MSB, then  $y_{jk} = 1$ , and  $y_{jk} = 0$  otherwise.
- 4) The BS-BSC topology is represented by the matrix  $[\mathbf{X}]_{ij} = x_{ij}$  in which, if the  $i$ th BS is connected to the  $j$ th BSC, then  $x_{ij} = 1$ , and  $x_{ij} = 0$  otherwise.

The paging cost on a cell is determined by the total call arrivals to cells that belong to the same LA. We use a vector,  $\lambda^*$ , where the  $i$ th entry holds that total value for  $i^{th}$  cell, which can be calculated as

$$\lambda_i^* = \lambda_i + \sum_s \lambda_s (1 - d_{is}) \quad (1)$$

As a result, our “minimize the total registration signalling” goal is formulated as follows.

$$\text{Minimize } \sum_i \sum_s d_{is} h_{is}, \quad (2)$$

subject to the following constraints.

- 1) Each BS should be assigned to exactly one BSC. Thus, for the  $i$ th cell, the BS-BSC topology matrix should have only one entry in the  $i$ th row that is 1 and others must be 0.

$$\sum_j x_{ij} = 1, \forall i. \quad (3)$$

- 2) Each BSC should be assigned to exactly one MSC. For the  $j$ th BSC, the BSC-MSC topology matrix should have only one entry in the  $j$ th row that is 1 and all others must be 0.

$$\sum_k y_{jk} = 1, \forall j. \quad (4)$$

- 3) Each BS should be assigned to exactly one LA. For the  $i$ th cell, the BS-LA topology matrix should have only one entry in the  $i$ th row that is 1 and others must be 0.

$$\sum_n \ell_{in} = 1, \forall i. \quad (5)$$

- 4) Each LA must reside within exactly one MSC. For each cell pair, if they belong to the same LA, then they also must belong to the same MSC. Thus,

$$\text{if } \sum_n \ell_{in} \ell_{sn} = 1, \text{ then } \sum_k \left( \sum_j x_{ij} y_{jk} \sum_r x_{sr} y_{rk} \right) = 1, \forall (i, s). \quad (6)$$

- 5) The paging capacity of any BS must not be exceeded.

$$\lambda_i^* < P_i^{BS}, \forall i. \quad (7)$$

- 6) The paging capacity of any BSC must not be exceeded. Assuming that a BSC pages the Mobile Station (MS) separately in each cell of the same LA, this constraint becomes

$$\sum_i x_{ij} \lambda_i^* < P_j^{BSC}, \forall j. \quad (8)$$

- 7) The call traffic capacity of any BSC must not be exceeded.

$$\sum_i x_{ij} c_i < C_j^{BSC}, \forall j. \quad (9)$$

- 8) The call traffic capacity of any MSC must not be exceeded.

$$\sum_j \sum_i x_{ij} y_{jk} c_i < C_k^{MSC}, \forall k. \quad (10)$$

9) The BHCA capacity of any BSC must not be exceeded.

$$\sum_i x_{ij} d_i < D_j^{BSC}, \forall j. \quad (11)$$

10) The BHCA capacity of any MSC must not be exceeded.

$$\sum_j \sum_i x_{ij} y_{jk} d_i < D_k^{MSC}, \forall k. \quad (12)$$

11) The TRX capacity of any BSC must not be exceeded.

$$\sum_i x_{ij} r_i < R_j^{BSC}, \forall j. \quad (13)$$

12) The TRX capacity of any MSC must not be exceeded.

$$\sum_j \sum_i x_{ij} y_{jk} r_i < R_k^{MSC}, \forall k. \quad (14)$$

13) The proximity constraints must be satisfied.

$$x_{ij} \leq a_{ij}, \forall (i, j), \quad (15)$$

$$y_{jk} \leq b_{jk}, \forall (j, k). \quad (16)$$

The generalized LA planning formulation in equations (1)-(16) poses a difficult optimization problem in which the LA-BS (matrix  $\mathbf{L}$ ), BSC-MSC (matrix  $\mathbf{Y}$ ) and the BS-BSC topologies (the matrix  $\mathbf{X}$ ) have to be determined. There may be many special cases of the general LA planning problem. Typically, some of the constraints may be omitted. For example, there may not be any call capacity constraint for the MSCs, in which case, Constraint (10) is omitted. Another special case arises when there may be exactly one BSC for each MSC. Then,  $\mathbf{Y}$  is already known.

### III. SOLUTION TECHNIQUES

Our aim is to develop algorithms that take the available constraints, capacity and load information described in Section II as inputs, and find an optimal or near optimal network topology which includes the assignment of cells (BSs) to switches (BSCs and MSCs) and location areas (LAs).

### A. Remarks on the Complexity of the Problem

Since the solution space consists of all possible network topologies, the BS-to-BSC, BSC-to-MS and BS-to-LA assignments determine the complexity of the solution. If there are  $n$  BSs,  $m$  BSCs,  $p$  MSs, the number of possible BS-BSC assignments is calculated as follows: For each BS, there are  $m$  possible BSCs to be connected, hence  $n^m$  possible connections. Likewise, the number of possible BSC-to-MS assignments is  $m^p$  and because the upper limit of the number of LAs is the number of BSs ( $n$ ), the number of possible BS-to-LA assignments is  $n^n$ . Consequently, the size of the solution space is  $n^m m^p n^n$ . Although the solution space includes both feasible and infeasible solutions, all solutions must be assessed whether they are feasible or not. Because of the solution space size, exhaustive methods result in exponential time complexity.

In order to give a sense for the difficulty of the formulation, NP-hardness of two subproblems can be shown. The assignment of cells to switches without considering the location areas was studied in [25], where the problem was formulated as integer programming and shown to be NP-hard.

Next, consider the subproblem of assigning BSs to LAs. LAs correspond to bins due to their common paging capacity limit. We try to pack the BSs to LAs in such a way that the number of LAs is minimum. In general, the smaller the number of LAs, the smaller the registration cost. In the limit, if there is only one LA, there is no need for registration, and the corresponding cost becomes zero. Therefore, having the registration cost or the number of LAs in the objective function results in similar final solutions. Since minimizing the number of LAs can be formulated as a bin packing problem which is known to be NP-hard [26], we conjecture that the optimization described by (1)-(16) is also NP-hard.

### B. Simulated Annealing

Due to the difficulty of the problem described in Section II, it is not possible to guarantee the optimal solution in reasonable running times. Therefore, techniques that give near optimal solutions within acceptable run times are needed. One method that finds a suboptimal solution without searching the entire solution space is simulated annealing (SA) [27], which approximates the solution of very large combinatorial optimization problems [28]. Detailed information about the SA algorithm can be found in [29].

The SA-based algorithm described here finds a network topology that consists of the LA-BS assignments (the matrix  $L$ ), the BSC-MSK topology (the matrix  $Y$ ) and the BS-BSC topology (the matrix  $X$ ).

Figure 1 shows the flowchart of the SA algorithm. For a successful SA implementation, two key items that must be defined carefully are the moves that create neighbor solutions and the cooling schedule. A feasible solution is a topology where all network nodes (BSs, BSCs, MSCs) are connected and the LA borders are specified while satisfying the constraints. Therefore, a (feasible) neighbor solution may be generated by any of the three types of moves:

- 1) Changing a BS-to-BSC assignment: If constraints are not violated, a new BSC is assigned to a randomly chosen BS. Then, a random feasible LA is assigned to that BS. If no feasible LA is found, a new LA is created.
- 2) Changing a BSC-to-MSK assignment: First, a BSC and a candidate MSK for new assignment is chosen randomly. If the constraints are not violated, the LAs residing merely in that BSC (i.e., none of their BSs are connected to another BSC) stay with their BS connections (i.e., nothing is changed for these LAs), but LAs having some of their BSs connected to that BSC go through rearrangement. The assignments of BSs to those LAs are released, and therefore those LAs have no more BSs assigned to them from that BSC. For those “released” BSs, an adequate number of new LAs are created (if the capacity of a new LA is full, then another LA is created).
- 3) Changing a BS-to-LA assignment: Without affecting the BS-BSC connection, a new LA assignment is done by searching the available LAs residing within the same BSC.

The cooling schedule consists of three components: setting the initial temperature; the decision of when and how to decrease the temperature; and the stopping rule of the algorithm. The efficiency of a cooling schedule is determined by the quality of the solution and the rate with which the process converges [30]. The performance of SA depends heavily on the cooling schedule and neighborhood structure, which should be carefully investigated and optimized to have the best results.

### C. Greedy Search

As a natural competitor, Greedy Search (GS) is one of the techniques that we compare the SA-based algorithm to. It starts with randomly selecting one of the feasible neighbors. If the chosen neighbor improves the registration cost of the current solution, then it is accepted as the current solution. The search continues until an “iteration limit” is reached or a “specified number of iterations” result in no improvement. The flowchart of the GS algorithm is obtained by setting the initial temperature to 0 in the SA flowchart (Figure 1). When the system temperature is zero, the probability of accepting a neighbor that yields worse registration cost becomes zero. Therefore, only the neighbors that improve the registration cost of the current solution are accepted.

The registration cost produced by GS should be an upper bound on the SA runs, since the greedy search can only find the local minima of the solution space. The quality of GS results rely on the initially formed solution, and a number of simulation runs are performed to test the GS with different initial solutions.

### D. Heuristics

Since heuristics try to find the best solution without tracing an extensive part of the solution space and instead attempt to build the best solution directly, their time complexity is generally much better than other techniques. However, heuristic solutions tend to offer inferior quality.

The cellular network registration cost is determined by the sum of the number of handovers among cells belonging to different LAs, since those handovers are assumed to be proportional to the number of location updates that occur by crossing the LA borders. The total network cost can be minimized by designing the LA borders such that the cell pairs belonging to different LAs have the lowest handover rates. We call the heuristic based on the preceding approach as the Heuristic for Optimization of Location Area Planning (HOLAP). Hence, HOLAP aims to gather cell pairs with high handovers within the same LA so that the registration cost is not raised by the high handover rates. Figure 2 represents the flow diagram of HOLAP, which consists of two main blocks (loops). The algorithm starts with sorting the cell pairs according to the handover rates between them.

Starting with the pair that induces the highest handover rate, HOLAP tries to assign a common LA to both cells of that pair. The process begins with searching feasible BSCs for the same pair

by assuring that the BSC capacities are adequate to handle these cell-to-BSC assignments. If any cell of the pair under process was previously assigned to an LA, that pair is skipped. Otherwise, if the number of available LAs is greater than one, then the decision of the LA assignment is made based on the decision priorities shown in the flow diagram. After the processing of all cell pairs in a sequential manner, there will be cells that are not assigned to any LA. For the remaining unassigned cells, the algorithm described in the second loop applies, and again the decision priorities are considered. HOLAP is compared to SA and GS in Section IV-C.2.

#### IV. COMPUTATIONAL EXPERIMENTS

The cluster system named ASMA (Advanced System for Multicomputer Applications) is employed in the experiments. ASMA resides in the Computer Engineering Department of Boğaziçi University, and has 32 personal computers (PCs) with Linux operating system [32]. In order to find the SA parameter set that gives the best results, different parameter values are tried on multiple data sets. Experiments are repeated approximately 30 times for each data and parameter set. The data sets used in the experiments are collected from an operational GSM network at different times of a week. By choosing a pilot area from the GSM network, all relevant information about the network elements (BS, BSC, and MSCs) is extracted. The capacity limits are vendor-specific and those stipulated by the GSM operator's equipment are fed to the algorithm as execution parameters. In particular, the paging capacity limits for a cell and a BSC are 33.98 and 8,500 paging commands per second, respectively. Call traffic capacity limits for a BSC and an MSC are 1,300 and 2,600 Erlangs, respectively. The BHCA limits for BSC and MSC are 200,000 call attempts per unit time. Finally, the TRX capacity limits of a BSC and an MSC are 600 and 1,200 TRXs, respectively.

In the second and third set of experiments, the greedy search (with random initial topology), random generation and HOLAP are applied to the same data sets for comparison with the SA technique. Evaluation of SA against other evolutionary algorithms for different applications can be found in [33], [34].

##### A. Simulated Annealing-Based Solution

Figure 3 shows the registration cost of the solutions found at different stages of a typical SA run for a moderate sized example network, which has about 600 BSs, 6 BSCs and 6 MSCs.

While the SA parameters are optimized with this network, verification via other networks is also carried out. In the initial stages, the SA algorithm accepts nearly all solutions, since the temperature is very high. It acts as a random search first, and the registration cost of the accepted solutions fluctuates in a wide range as seen in Figure 3. As the temperature decreases, the probability of accepting worse solutions also decreases so that in the later stages of the run, the search becomes greedy and only better solutions are accepted.

The parameter set of SA includes the number of neighbors assessed for the initial temperature setting (IN), the  $\alpha$  value (alpha) and the number of accepted neighbor movements (AD) to decrement the temperature, the number of iterations to stop the execution (IL) and the number of temperature decrements without registration cost improvement to stop the execution (WI). The optimized parameter set found for the reference case with 600 BSs, 6 BSCs and 6 MSCs is {IN:10,000, alpha:0.9999, AD:20, IL:5,000,000, WI:5,000}, as described in [31]. The execution time varies with the size of the problem and the SA parameter values, but the longest run has taken about 1.5 hours the above network.

### B. Statistical Quality Measures

Besides GS and HOLAP, a large set of randomly generated solutions is also used to investigate the quality of SA results. The two random generation (RG) methods, labelled as RG1 and RG2 are based on the procedure where first BSs are connected randomly to feasible BSCs (considering the constraints), and then BSCs are connected to feasible MSCs. In RG1, for each MSC starting with one LA, BSs are assigned to an LA. If LA capacity (paging capacity of BSs) reaches its limit, then a new LA is created and the remaining BSs are assigned to that LA. Here the goal is to create the minimum number of LAs for each MSC (after randomly establishing the BS-BSC-MSC topology). In contrast, in RG2, a random number of LAs are created for each MSC. The randomization is based on the uniform distribution with upper limit to be the number of BSs connected to BSCs of the MSC. Then, all BSs are assigned to a randomly feasible LA. If for a BS, no feasible LA exists, then a new LA is created and the BS is assigned to it. With RG2, the aim is to make all of the network topology decisions random, and therefore to sample the solution space more uniformly.

A sample histogram of the registration cost values found by RG1 and RG2 can be seen in Figures 4 and 5, respectively. The histograms reflect approximately 1,000,000 random solutions

grouped within 50 bins (classes) according to their network registration costs. The number of solutions that fall into each bin are counted. Relative frequency shows the number of occurrences for each bin, and gives an idea about the distribution of the registration costs of generated solutions. Figure 4 depicts that the registration cost values of RG1 have a normal distribution. On the other hand, because for each MSC a random number of LAs are created in RG2, the total number of LAs created in the system deviates a lot from the minimum possible number. For example, if the number of MSCs in a system is 6, then minimum number of LAs that can be created is 6. For each MSC, LAs are created randomly, and the probability of having the number of LAs close to 6 is very small. This explains the accumulation of the RG2 results to the right side of Figure 5.

### C. Comparison of Solution Techniques

The comparison between SA and the other techniques is based on the resulting network registration cost. Both information collected from three pilot areas of a GSM network and randomly generated network data are employed in experiments. For the latter, the loads on all network elements are assumed to be exponentially distributed with mean set equal to the average value measured on the GSM network. Proximity matrices are such that all BS-to-BSC and BSC-to-MSC connections are allowed. A third set of experiments are performed with random data representing different types of cellular networks. For instance, to represent a network with higher mobility subscribers, the mean value of the handover rates is scaled up. Sample random data generated, the network architecture of the best solution found by SA and GS algorithms, and source code for all techniques are hosted on the Boğaziçi University Computer Engineering Network Research Laboratory webpage [35].

Experimental results produced by the SA algorithm reflect the averages of about 30 independent runs. The minimum registration costs found by the GS, RG1 and RG2 runs are selected for comparison. The number of GS algorithm runs on each data set is between 30 and 100. RG1 and RG2 methods are applied 1,000,000 times to each data set.

1) *Experiments on Real Network Data:* The experiments that are based on the GSM network data yield Table I. All data are collected at the peak hour when channel occupation is at its maximum within the day. The first set of data is measured on a Friday in a pilot area of the GSM network that belongs to a metropolitan city. Data sets 2 to 5 are extracted from a smaller

pilot area of the same city on a Monday, Tuesday, Saturday and Sunday, respectively. The sixth data set is comes from the third pilot area that belongs to a less populated city on a Friday. Table I shows that the SA-based solution technique outperforms the others. Furthermore, the minimum registration costs found by the RG1 runs are uniformly lower than those attained by RG2. Thus, setting the number of LAs randomly and then making the BS-BSC-MSC assignments does not give a better result than making BS-BSC-MSC assignments randomly and then setting the number of LAs to the minimum possible value. Another outcome of Table I is that SA gives approximately 50% better results than GS, which turns out to be the best competitor to SA. The reason behind the superiority of SA is that the SA algorithm is able to escape from local minima, while GS always gets stuck in a local minimum.

Although just the averages of the SA results are presented in Table I, their variance on different runs is not very large. For example, the 30 separate SA runs performed for Data Set 1 range from 405,571 to 475,072. The maximum and the minimum registration costs found on distinct SA runs diverge at most 15% from the average. Therefore, we may state that SA has a tight range, and any execution of the SA method gives results near the values presented in the tables.

In order to have a feeling about the distribution of the registration cost values in the solution space, and to see how far from the average of the solution space the SA results are, RG methods are used, as well. The resulting networks of RG methods consist of real (measured) loads, albeit an arbitrary topology. For the first group of experiments, the mean value of SA results are compared to the mean and standard deviation ( $\sigma$ ) of the RG1 runs in Table II. The SA results are located very far from the average. For Data Set 1, the distance between the SA and the mean RG1 results is  $13\sigma$ .

RG2 designates the actual solution space; however, it only includes the feasible solutions. Finally for the first group of experiments, Table III shows the mean value of SA results compared with the mean and standard deviation of RG2 runs. Since the network topologies created by RG2 method are collected in a smaller region, RG2 has smaller standard deviation compared to RG1. Hence, in the units of  $\sigma$ , the distance of SA result to the RG2 mean is much more than the RG1. As an example, again for the case of the second data set, the distance from SA result to the mean of RG2 results is  $48\sigma$ . Another observation from Table III is that the first data set produces much higher values compared to other data sets. The reason is that the first pilot area is a hotspot in

the city.

2) *Experiments on the Second Set of Example Problems:* *RandSetA1* to *RandSetA10* of Table IV list the resulting network registration costs for a network with 386 BSs, 6 BSCs and 6 MSCs. Although the loads are randomly generated for all elements based on the average values calculated using real GSM network data, the results obtained from all techniques are similar. Table IV depicts that SA again finds the best solution. However, the GS results are closer to those of SA compared to the runs performed with the real data. For the first data set, the SA registration cost is 11% better than GS as opposed to 43% for Data Set 1 in Table I. Loads generated according to a specific distribution are similar unlike real GSM network measurements. Hence, neighboring solutions have similar registration costs, and it is harder for SA to find significantly better solutions.

*RandLargeSet1* and *RandLargeSet2* of Table IV indicate that SA produces superior solutions for a network consisting of 800 BSs, 12 BSCs and 12 MSCs. For the experiments conducted on this larger network topology, data sets are obtained again through random data generation. It is interesting that the results are very close to those listed for Data Sets 2-4 of Table I. This similarity is an evidence for the reliability of random data generation.

The comparison between SA and HOLAP is in Table V for 386 BSs, 6 BSCs and 6 MSCs. The loads of the network elements are generated randomly based on the average values obtained from the GSM network. As it turns out, HOLAP is not a competitor to SA, and not even to GS. The reason behind the worse performance of HOLAP can be that the first those pairs with higher handover rates are considered and the pairs with low handover rates are discarded. The main goal is to allocate the pairs with high handover rates to the same LA. The remaining pairs have to be assigned to different LAs due to system constraints, and it is these ones that determine the network registration cost. That is, cell pairs with average or low handovers are also critical for the network registration costs and should not be ignored.

3) *Effect of Network Properties:* For a network with high (low) mobility, the average value of the handover rates, which is obtained from the GSM network, is multiplied (divided) by two. A network with high (low) paging is generated by multiplying (dividing) the average paging rate by three. A random network with low (high) call traffic load is obtained by dividing (multiplying) the average value of the call traffic load by two. However, for the high call traffic case, because the number of BSCs used in the experiments, which is six, is not capable of handling the number

of BSs, which is 386, we decrease the number of BSs to 250. To obtain a network with high or low TRX load, the same procedure is applied to the average values as in networks with high or low call traffic. Again because of capacity constraints, for high TRX load, the number of BSs is reduced to 250.

In Table VI, data sets from *HighMob1* to *HighMob4* have high mobility, and data sets from *LowMob1* to *LowMob4* belong to networks with low mobility. In the rest of the tables, the data sets represent a random network whose data are formed by using the same average values. This is one reason why the registration costs in the table are close. Table VI indicates that the impact of high mobility is an increase in registration cost due to increased handover rate. When compared to Table IV, this becomes more obvious. Although SA gives better results compared to other techniques, for some experiments, GS does approach SA (e.g., data set HighMob2).

A network with high paging implies that the number of mobile-terminated calls occurring in the network is high. As seen in Table VI, for the high paging situation, SA gives better outcomes compared to other methods. In addition, we notice that there are improvements in SA network registration costs over Table IV. However, the paging capacity is one of the constraints, and high paging reduces the size of the feasible solution space. In contrast, the results of GS, RG1, and RG2 are very close to those in Table IV. Since methods other than SA use random search and simple logic, they cannot improve the results when constraints are tightened. It is also observed that low paging and high paging cases create similar results.

Table VII displays the performance against traffic load. Comparison of data sets *LowTraff1-LowTraff4* of Table VII with data sets *LowPag1-LowPag4* of Table VI reveals that although SA, RG1 and RG2 generate networks with similar registration costs to those in Table VI, GS yields higher registration costs. Moreover, networks with low call traffic have worse registration costs than those with high load, because the number of feasible solution space is enlarged by relaxing a constraint, and it is harder to find better results in a larger space.

## V. CONCLUSIONS

The most important benefit of optimized LA planning is preventing needless radio resource usage that can instead be allocated for the communication of the customers. Furthermore, the network elements can be utilized more efficiently, and network construction costs can be reduced. In order to design a feasible cellular network, constraints on the call handling capacities

of network elements and costs related to the paging and registration activities should be considered. In this paper, we presented a generalized formulation of the LA planning and cell-to-switch assignment problem that uses the available network information in a realistic manner. We also proposed a solution technique based on the SA algorithm, which has not been studied in this field, as well as, a heuristic algorithm to the NP-hard optimization problem. The load values supplied to the SA-based algorithm can be generated using any model, or they can be fetched from an existing network.

We described the implementational details of the algorithm and compared it to other solution techniques. For the comparison between SA and the other techniques, three types of experiments were conducted. Real GSM network data (loads of the functional cells, etc.) were used in the first type of experiments, with the SA-based algorithm delivering the best performance. The second best method was GS, with 50% higher registration cost than SA. For the second group of experiments, to assure that the success of SA is not dependent on to a certain data set, randomly generated network data are deployed. Again SA performed the best; however, the difference between SA and GS results narrowed down to approximately 15%. The heuristic algorithm HOLAP turned out to be poorer than SA and GS. The last group of experiments investigated the impact of load characteristics, with superior results for SA.

Better heuristic algorithms need to be developed in the future. The SA algorithm can also be run on a wider test set to check the optimality of currently proposed SA parameter values.

## REFERENCES

- [1] B. L. Mark and Z. R. Zaidi "Robust mobility tracking for cellular networks", *IEEE International Conference on Communications, Vol. 1*, New York, USA, April-May 2002, pp. 445-449.
- [2] C. U. Saraydar, O. Kelly and C. Rose, "One-dimensional location area design", *IEEE Transactions on Vehicular Technology*, Vol. 49, No. 5, pp. 1626-1633, September 2000.
- [3] C. U. Saraydar and C. Rose, "Location area design using population and traffic data", *Proc. of CISS*, Princeton, NJ, March 1998.
- [4] P. G. Escalle, V. C. Giner and J. M. Oltra, "Reducing location update and paging costs in a PCS network", *IEEE Transactions on Wireless Communications*, Vol. 1, No. 1, pp. 200-209, Jan. 2002.
- [5] T. Ozugur, "Multiobjective hierarchical location and routing area optimization in GPRS and UMTS networks", *IEEE International Conference on Communications, Vol.1*, New York, USA, April-May 2002, pp. 579-584.
- [6] S. Z. Ali, "Design of location areas for cellular mobile radio networks", *Proc. of the 55th IEEE Vehicular Technology Conference, Vol. 3*, 2002, pp. 1106-1110.
- [7] L. D.-Jun and C. D.-Ho, "On optimum timer value of area and timer-based location registration scheme", *IEEE Communications Letters*, Vol. 5, No. 4, pp. 148-150, April 2001.

- [8] S. H. Ryu, K. H. Lee, Y. Y. Oh, J. Y. Lee and S. B. Lee, "Adaptive time-based location update scheme using fuzzy logic", *IEEE International Conference on Consumer Electronics*, Los Angeles, CA, June 1999, pp. 322-323.
- [9] Y. Fang, I. Chlamtac and F. H.-Ring, "Analytical Results for Optimal Choice of Location Update Interval for Mobility Database Failure Restoration in PCS Networks", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 11, No. 6, pp. 615-624, June 2000.
- [10] Z. Mao and C. Douligeris, "Two location tracking strategies for PCS systems", *Proceedings of Eight International Conference on Computer Communications and Networks*, Boston, USA, October 1999, pp. 318-323.
- [11] Y. Fang, "Tradeoff analysis for location update and paging in wireless networks", *Proceedings of IEEE GLOBECOM*, San Antonio, Texas, USA, November 2001, pp. 1754-1758.
- [12] J.-T. Tsai and H.-H. Hsiao, "Performance of Movement-based Location Update and One-step Paging in Wireless Networks with Sparsely Underlaid Microcells", *Proceedings of IEEE GLOBECOM*, San Antonio, Texas, USA, November 2001, pp. 642-647.
- [13] L. Jie, H. Kameda and K. Li, "Optimal dynamic mobility management for PCS networks", *IEEE/ACM Transactions on Networking*, Vol. 8, No. 3, pp. 319-327, June 2000.
- [14] W. Wang and I. F. Akyildiz, "Intersystem location update and paging schemes for multitier wireless networks", *Proc. ACM MobiCom'00*, Boston, Massachusetts, USA, August 2000.
- [15] V. Wong and V. Leung, "An adaptive distance-based location update algorithm for next-generation PCS networks", *IEEE Journal on Selected Areas in Communications*, Vol. 19, No. 10, pp. 1942-1952, October 2001.
- [16] M. Verkama, "A simple implementation of distance-based location updates", *Proc. of the 6th IEEE International Conference on Universal Personal Communications*, vol. 1, San Diego, CA, October 1997, pp. 163-167.
- [17] V. W.-S. Wong and V. C. M. Leung, "An adaptive distance-based location update algorithm for PCS networks", *IEEE International Conference on Communications*, Vol. 7, Helsinki, Finland, June 2001, pp. 2000-2005.
- [18] V. Casares-Giner and J. Mataix-Oltra, "Global versus distance-based local mobility tracking strategies: a unified approach", *IEEE Transactions on Vehicular Technology*, Vol. 51, No. 3, pp. 472-485, May 2002.
- [19] A. Bar-Noy, I. Kessler and M. Sidi, "Mobile users: to update or not to update?", *Wireless Networks*, Vol. 1, No. 2, pp. 175-185, July 1995.
- [20] I. F. Akyildiz and W. Wang, "A dynamic location management scheme for next-generation multitier PCS systems", *IEEE Transactions on Wireless Communications*, Vol. 1, No. 1, pp. 178-189, Jan. 2002.
- [21] C. Woo-Jin and S. Tekinay, "An adaptive location registration scheme with dynamic mobility classification", *IEEE International Conference on Communications*, Vol. 1, New York, USA, April-May 2002, pp. 440-444.
- [22] K. T. Chen, S. L. Su, and R. F. Chang, "Design and analysis of dynamic mobility tracking in wireless personal communication networks", *IEEE Transactions on Vehicular Technology*, Vol. 51, No. 3, pp. 486-497, May 2002.
- [23] L. Shou-Chih and A. L. P. Chen, "Adaptive region-based location management for PCS systems", *IEEE Transactions on Vehicular Technology*, Vol. 51, No. 4, pp. 667-676, July 2002.
- [24] G. Heine, *GSM Networks: Protocols, Terminology, and Implementation*, Artech House Publishers, Boston, 1998.
- [25] A. Merchant and B. Sengupta, "Assignment of cells to switches in PCS networks", *IEEE/ACM Transactions on Networking*, Vol. 3, No. 5, pp. 521-526, October 1995.
- [26] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, San Francisco, California: W. H. Freeman, 1979.
- [27] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller, "Equation of state calculations by fast computing machines", *Journal of Chemical Physics*, Vol. 21, No. 6, pp. 1087-1092, June 1953.

- [28] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, "Optimization by simulated annealing", *Science*, Vol. 220, pp. 671-680, May 1983.
- [29] E. H. Aarts, J. Korst and P. J. van Laarhoven, *Simulated Annealing*, In E. Aarts and Lenstra J. K. (editors), *Local Search in Combinatorial Optimization*, pp. 91-120, John Wiley & Sons Ltd., Chichester, 1997.
- [30] E. Rich and R. Knight, *Artificial Intelligence*, McGraw-Hill, New York, 1991.
- [31] İ. Demirkol, C. Ersoy, M. U. Çağlayan and H. Deliç, "Location area planning in cellular networks using simulated annealing", *Proceedings of IEEE INFOCOM*, Anchorage, Alaska, April 2001, pp. 13-20.
- [32] Boğaziçi University Computer Engineering Department, *BU ASMA Project Home Page*, 1999 <http://www.asma.cmpe.boun.edu.tr>.
- [33] S. Elmohamed, P. Coddington and G. Fox, "A comparison of annealing techniques for academic course scheduling", *Proceedings of the 2nd International Conference on the Practice and Theory of Automated Timetabling*, Toronto, Canada, August 1997, pp. 146-166.
- [34] W. E. Hart, "A theoretical comparison of evolutionary algorithms and simulated annealing", *Proc. of 5th Annual Conf. on Evolutionary Programming*, San Diego, CA, February-March 1996, pp. 147-154.
- [35] Boğaziçi University Computer Engineering Department, *Computer Networks Research Lab*, 2002 <http://www.netlab.boun.edu.tr/thesis/ilker/index.html>.

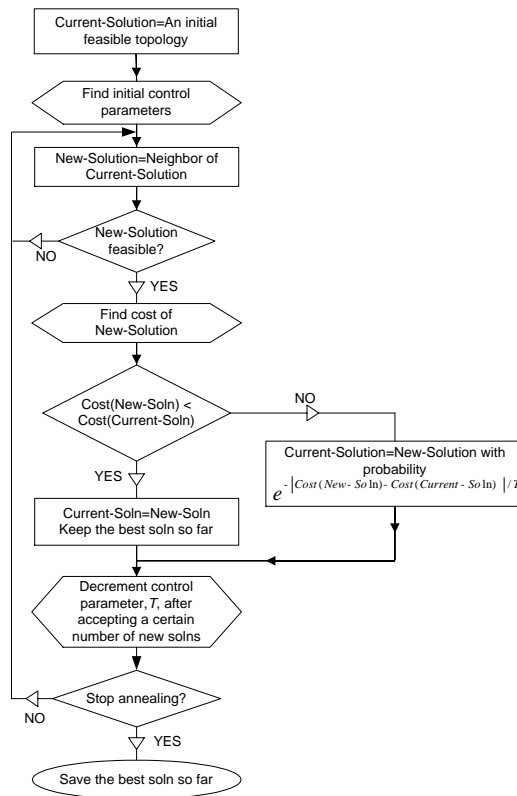


Fig. 1. SA Flowchart

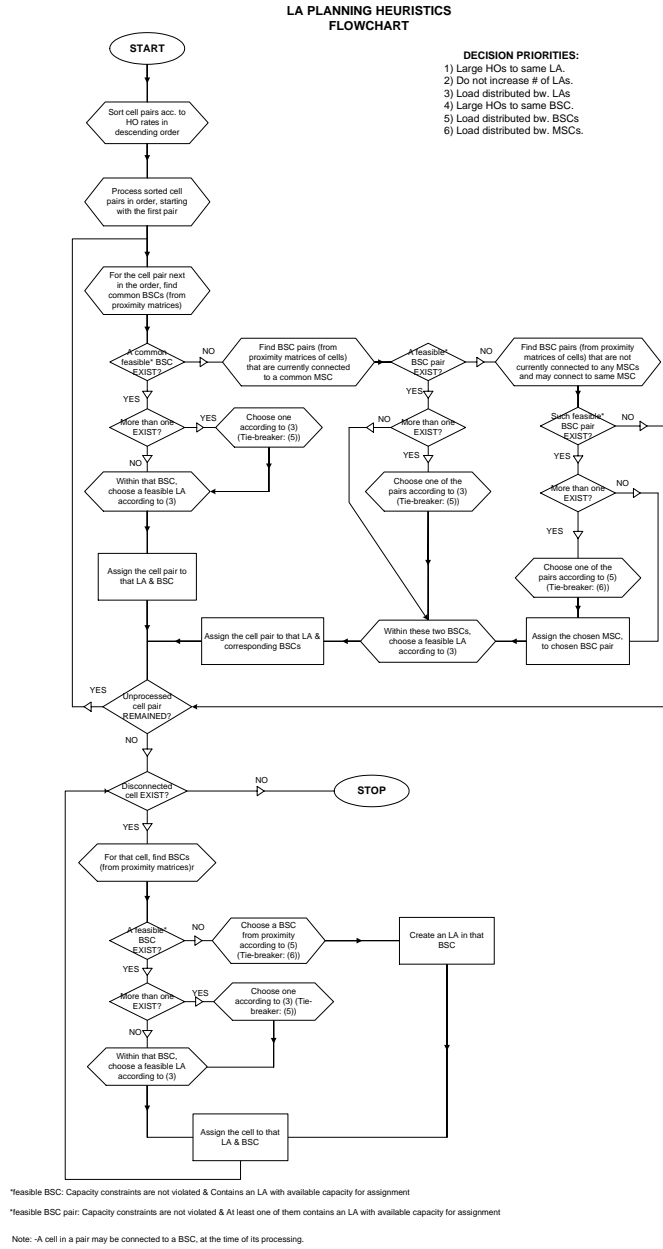


Fig. 2. Flowchart of HOLAP.

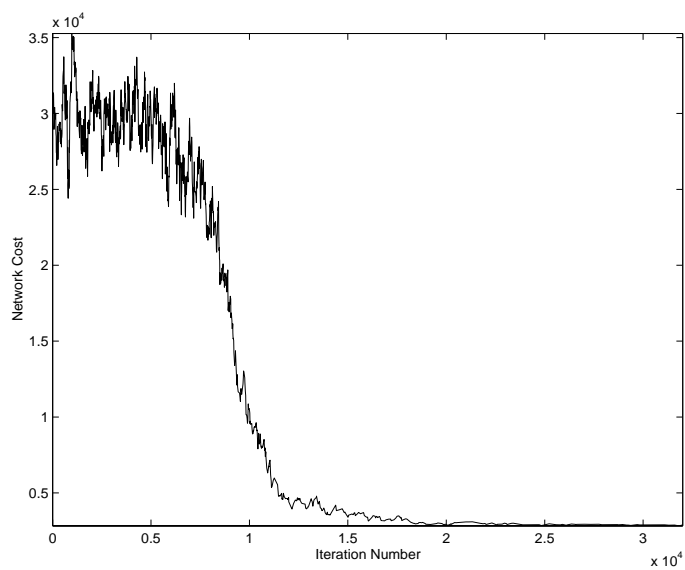


Fig. 3. A typical run of the SA algorithm on a sample network. The registration cost of the network is computed in total number of hand-offs incurred between the LAs per second.

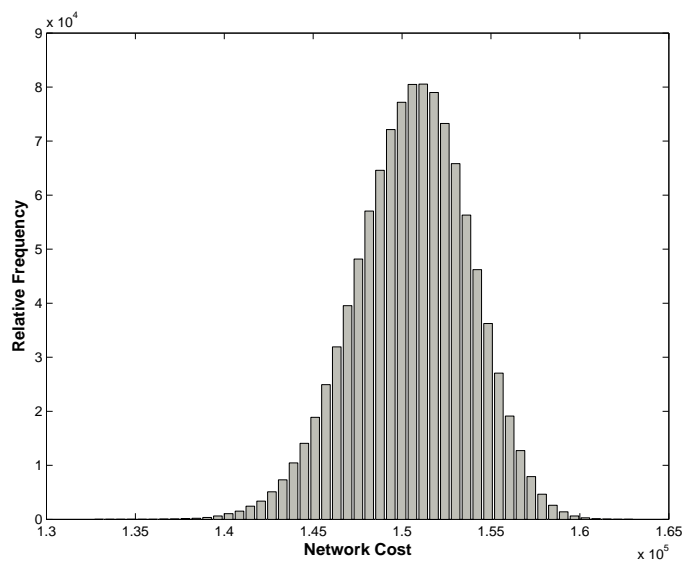


Fig. 4. RG1 histogram. The registration cost of the network is computed in total number of hand-offs incurred between the LAs per second.

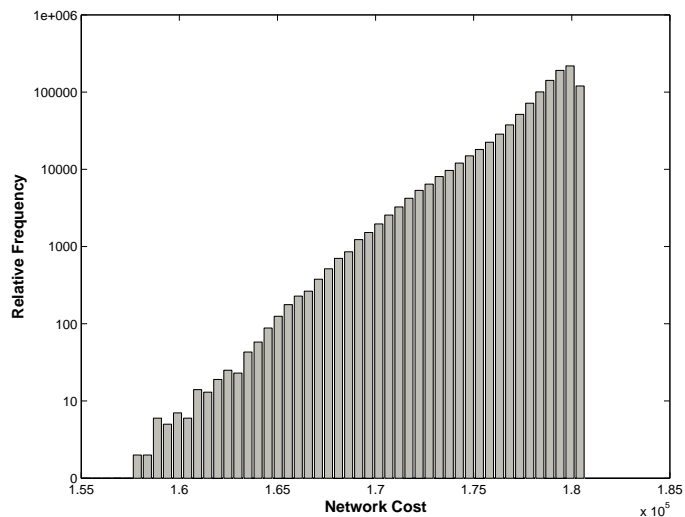


Fig. 5. RG2 histogram (with logarithmic frequency axis). The registration cost of the network is computed in total number of hand-offs incurred between the LAs per second.

TABLE I

SA, GS, RG1, AND RG2 COMPARISONS FOR DIFFERENT PROBLEMS BASED ON DATA OF AN EXISTING NETWORK

Data	SA	GS (Min)	RG1 (Min)	RG2 (Min)
Set1	450,882	792,769	1,870,809	2,772,498
Set2	12,161	24,693	134,410	161,974
Set3	15,352	27,930	138,889	161,365
Set4	11,514	20,839	109,852	131,300
Set5	7,308	14,910	84,571	101,962
Set6	901	1,956	22,402	31,508

TABLE II

SA AND RG1 COMPARISON FOR DIFFERENT PROBLEMS BASED ON DATA OF AN EXISTING NETWORK

<b>Data</b>	<b>SA</b>	<b>RG1 (Mean)</b>	<b>RG1(Std. Dev.)</b>
<b>Set1</b>	450,882	2,572,020	156,555
<b>Set2</b>	12,161	150,569	3,220
<b>Set3</b>	15,352	153,915	3,305
<b>Set4</b>	11,514	122,772	2,493
<b>Set5</b>	7,308	94,253	1,845
<b>Set6</b>	901	29,937	1,692

TABLE III

SA AND RG2 COMPARISON FOR DIFFERENT PROBLEMS BASED ON DATA OF AN EXISTING NETWORK

<b>Data</b>	<b>SA</b>	<b>RG2 (Mean)</b>	<b>RG2(Std. Dev.)</b>
<b>Set1</b>	450,882	3,379,986	61,265
<b>Set2</b>	12,161	178,509	1,993
<b>Set3</b>	15,352	182,433	2,034
<b>Set4</b>	11,514	145,526	1,636
<b>Set5</b>	7,308	111,731	1,206
<b>Set6</b>	901	43,621	1,527

TABLE IV  
SA, GS, RG1, AND RG2 RESULTS FOR RANDOM DATA

Data	SA	GS (Min)	RG1 (Min)	RG2 (Min)
RandSetA1	5,647	6,346	35,257	54,163
RandSetA2	5,495	6,844	37,306	57,575
RandSetA3	5,178	6,159	31,567	45,649
RandSetA4	5,824	7,285	36,210	53,616
RandSetA5	5,229	5,130	30,201	47,459
RandSetA6	5,536	5,232	31,974	52,479
RandSetA7	5,541	7,647	35,043	54,024
RandSetA8	5,265	8,289	36,925	54,223
RandSetA9	5,626	6,520	36,927	55,071
RandSetA10	5,548	6,509	32,331	61,336
RandLargeSet1	13,807	27,725	106,844	126,409
RandLargeSet2	14,122	29,409	110,059	131,725

TABLE V  
HOLAP, SA, GS, RG RESULTS OF RANDOM DATA FOR NETWORKS WITH 386 BSSs, 6 BSCs AND 6 MSCs

Data	HOLAP	SA	GS (Min)	RG1 (Min)	RG2 (Min)
RandSetB1	21,244	4,565	6,145	27,720	41,787
RandSetB2	20,624	4,302	5,800	25,788	40,935
RandSetB3	20,766	4,674	5,609	27,481	39,720
RandSetB4	21,317	3,886	4,177	27,161	39,691
RandSetB5	21,579	4,137	7,230	26,502	38,055
RandSetB6	21,859	4,426	6,752	28,120	39,883
RandSetB7	19,708	3,995	5,410	26,395	40,918
RandSetB8	20,686	4,532	5,125	26,693	39,722

TABLE VI

SA, GS, RG1, AND RG2 COMPARISON FOR RANDOM DATA FOR NETWORKS WITH DIFFERENT MOBILITIES

<b>Data</b>	<b>SA</b>	<b>GS (Min)</b>	<b>RG1 (Min)</b>	<b>RG2 (Min)</b>
<b>HighMob1</b>	8,823	11,566	62,470	91,930
<b>HighMob2</b>	10,277	11,104	63,045	99,258
<b>HighMob3</b>	8,480	14,825	57,981	96,295
<b>HighMob4</b>	9,866	12,224	72,970	108,841
<b>LowMob1</b>	2,267	3,055	17,407	27,959
<b>LowMob2</b>	2,426	3,125	18,474	26,956
<b>LowMob3</b>	2,612	3,627	18,564	28,057
<b>LowMob4</b>	2,366	3,012	18,393	28,308
<b>HighPag1</b>	4,279	7,265	35,573	55,106
<b>HighPag2</b>	4,698	5,280	28,393	54,486
<b>HighPag3</b>	3,604	5,085	27,893	54,700
<b>HighPag4</b>	4,840	6,015	37,314	56,418
<b>LowPag1</b>	4,743	6,665	32,698	57,668
<b>LowPag2</b>	4,027	7,418	33,632	54,470
<b>LowPag3</b>	4,206	6,410	29,844	52,548
<b>LowPag4</b>	4,776	6,508	36,805	49,232

TABLE VII

SA, GS, RG1, AND RG2 COMPARISON FOR RANDOM DATA FOR NETWORKS WITH DIFFERENT CALL TRAFFIC LOADS

<b>Data</b>	<b>SA</b>	<b>GS (Min)</b>	<b>RG1 (Min)</b>	<b>RG2 (Min)</b>
<b>HighTraff1</b>	3,513	6,582	24,165	35,854
<b>HighTraff2</b>	3,342	6,955	25,441	34,921
<b>HighTraff3</b>	3,185	5,997	24,023	30,835
<b>HighTraff4</b>	3,219	6,436	25,584	37,676
<b>LowTraff1</b>	4,710	8,267	39,735	58,474
<b>LowTraff2</b>	4,754	7,876	39,183	57,255
<b>LowTraff3</b>	4,355	6,887	37,507	54,169
<b>LowTraff4</b>	5,052	7,763	36,996	51,881
<b>HighTRX1</b>	3,968	6,492	25,867	38,212
<b>HighTRX2</b>	4,402	6,541	27,092	38,069
<b>HighTRX3</b>	4,320	6,939	26,204	38,170
<b>HighTRX4</b>	4,027	6,899	25,539	36,076
<b>LowTRX1</b>	4,543	5,020	37,637	61,006
<b>LowTRX2</b>	4,234	7,365	37,701	55,743
<b>LowTRX3</b>	5,091	8,735	35,798	51,698
<b>LowTRX4</b>	4,617	7,207	37,649	54,631