Wireless Networks 7, 401–411, 2001
© 2001 Kluwer Academic Publishers. Manufactured in The Netherlands.

Multi-Tier Cellular Network Dimensioning *

EYLEM EKICI ** and CEM ERSOY
Computer Engineering Department, Boğaziçi University, Istanbul, Turkey

Received January 2000

Abstract. Wireless communications systems enable the end users to be mobile. The majority of the wireless communications networks are cellular networks. Several methods are developed to increase the performance of the cellular networks, which depends on the correct determination of the design parameters as well as the architecture of the system and the traffic requirements. In this study, we introduce a Simulated Annealing (SA) based method to determine the design parameters of a multi-tier cellular network, for which the implementation cost is minimized. The cellular system employs guard channels and allows calls to overflow to upper tiers. We conducted experiments with the SA-based technique on different example problems in two-tier cellular networks and obtained promising results.

Keywords: cellular networks, multi-tier, optimal network design

1. Introduction

As the technology evolves, the demand for better and faster communication systems also increases. After the introduction of wireless cellular communication systems a decade ago its growth has been rapid [1]. The number of wireless communication service users as well as the spectrum of the available services increased with an unexpected rate. The main reason for this growth was the newly introduced notion of terminal and user mobility. It is expected that the wireless communications will be the dominant mode of data access technology in the next century [2].

The majority of existing terrestrial wireless communication systems are based on the cellular concept [3,4]. The underlying network structure is composed of a fixed network with wireless last hops between *Base Stations* (BSs) and *Mobile Terminals* (MTs). The fixed communication network connects the base stations to controllers, a.k.a. *Mobile Switching Centers* (MSCs), that manage the calls and track all mobile terminal activities in a cell [5,6]. In some systems, multiple base stations are used to serve the same area. Hence, a *multi-layer cellular network* is formed [7,8]. Figure 1 shows a two-tier system, where a macrocell covers seven microcells.

The next generation wireless systems will provide global coverage. ITU provisions the use of Hierarchical Cell Structure (HCS) for IMT 2000 [9]. HCS will consist of high capacity picocells for in-building communications, micro and macrocells for urban and suburban communications, and satellite cells to complete the global coverage. The terminal mobility is one of the most challenging aspects of the next generation wireless systems, both for within a single layer and across the hierarchical layers. The mobility management issues in the next generation wireless systems are discussed in [10]. A solution to the location update and paging problem multitier wireless systems is presented in [11]. Other work on multi-tier cellular systems, such as [7,8], mainly concentrates on performance calculation of the system once the parameters are determined.

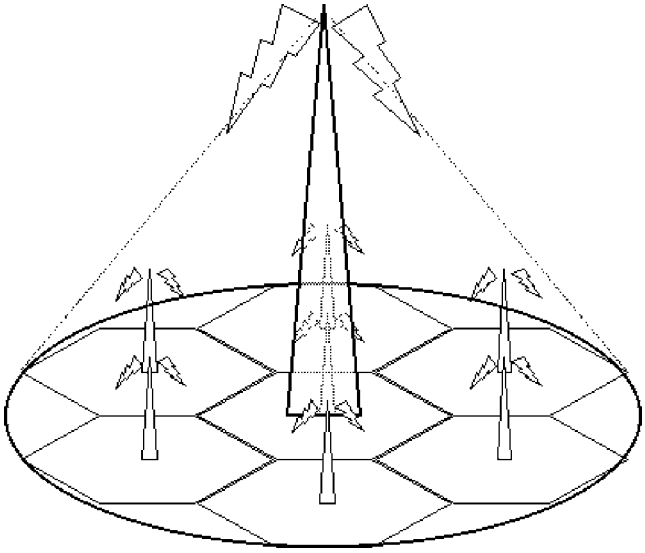

Figure 1. A macrocell covering seven microcells.

The switching of MTs from a cell to the other is called *handoff*. During this process, a channel is assigned in the new cell, while the channel used in the old cell is released. The so-called *call dropping* occurs when a call in progress is forcefully terminated due to lack of available channels in the new cell. A similar event, not serving a new call due to lack of available channels, is described as *call blocking*. Call blocking and dropping probabilities are used as performance measures while designing cellular systems. However, from the users' point of view, call dropping is less desirable than call blocking [12,13]. Therefore, in some TDMA-based

* This work is partially supported by the State Planning Organization of Turkey under the grant number DPT98K120890.

** While this work was carried out, Eylem Ekici was with the Computer Engineering Department of Boğaziçi University.

systems, the handoff calls are given higher priorities using *prioritized networks*. Several methods to provide prioritization of the handoff calls are outlined in the third chapter of [14]. One of these methods is to reserve some channels exclusively for handoff calls, also known as *guard channels*. In [15], an efficient way to minimize the call dropping probability while staying below a given call blocking probability threshold is presented. In [16], a variable reservation policy with handoff prioritization through guard channels is presented. Another handoff prioritization method is proposed in [17], where the number of guard channels are changed through hysteresis control. In CDMA-based systems, however, improvements in call dropping probability can also be achieved by controlling the call admissions to the neighboring cells as proposed in [18].

Another solution to call dropping problem is borrowing unused channels from neighboring cells [19,20]. However, this method can limit channel usage in the rest of the network and involves complicated control mechanisms. In order to reduce the global effects of channel borrowing, neighboring cells can be grouped as metacells [21]. With this method, only the metacells in the reuse distance can use the same pool of channels. The individual cells in the metacells can share the channels from the assigned pool without coordinating with other cells outside their metacell.

One of the challenges associated with multi-tier cellular networks is their design and determination of parameters. In [22], a multi-tier cellular network design algorithm is presented. The proposed method minimizes the total system cost of a multi-tier cellular network subject to call dropping and blocking probability constraints. With this method, the cell radii in each tier as well as channel partitioning among the tiers are determined. The method is based on exhaustively searching the parameter space. However, the tiers in the system are isolated and the performance increasing techniques such as guard channels and call overflow are not considered.

As described in [23], cellular network deployment has many aspects. In this work, we present a probabilistic optimization technique to determine the system parameters of multi-tier cellular networks, for which the implementation cost is minimized. The multi-tier cellular network is proposed for urban environments, where low mobility users, e.g., pedestrians, and high mobility users, e.g., mobile users in vehicles, are present in the same environment. The cellular network model has two tiers, employs guard channels in all tiers and allows calls to overflow to the upper tiers; hence, we aim to combine the advantages of multi-tier and prioritized networks. The optimization problem is subject to new call blocking and handoff call dropping probability bounds. Since the problem involves determination of multi-dimensional discrete and continuous parameters that depend on each other, it is not possible to solve the problem analytically. Furthermore, the large parameter space prohibits an exhaustive search approach. Therefore, a well-known artificial intelligence technique, *Simulated Annealing* (SA) [24], is chosen to solve this optimization problem. It is a neighborhood search technique with incorporated probabilistic behavior. SA combines the advantages of random and greedy search techniques.

The outline of this paper is as follows. In section 2, the design problem is formulated. Furthermore, the objective function that will be used in the solution techniques is described and the calculation methods are presented. Section 3 explains the solution technique extensively. In section 4, the results of the computational experiments are presented. The last section concludes this work.

2. Definition of the multi-tier cellular network design problem

The purpose of this work is to design a minimum cost multi-tier cellular network with call overflow and guard channels that satisfies certain performance constraints. In this work, we focused on the determination of system parameters that describe the multi-tier cellular network.

2.1. Assumptions

In the work presented, two classes of mobile terminals are assumed. The high mobility class represents the mobile terminals that are used in cars and other vehicles. The low mobility class is made up of users that are primarily pedestrians. The speeds of the mobile terminal users are exponentially distributed with mean values v_f and v_s, respectively. A similar mobility distribution was also used in [25]. Since the target deployment area is a metropolitan area, the distribution of the members of both mobility classes can be considered as uniform and all mobile terminals are assumed to move in any direction equally likely. The call duration is also exponentially distributed with mean $1/U_t$ for both mobility classes. The time spent in a cell, which is called *dwell time* U_d, is calculated as in [25], where r is the radius of the cell and v is the speed of the mobile terminal, resulting in the following identity:

$$\frac{1}{U_d} = \frac{r\pi}{2v}. \tag{2.1}$$

The call arrivals to the cells follow a Poisson distribution for both mobility classes. The mobile terminal density and the mean call generation rate of the individual mobile terminal users determine the mean call arrival rate. The antennas for both layers may be located at the same locations. Furthermore, the distinction between the mobility classes is made using the mean of the mean mobility rates as the threshold.

2.2. System description

The network to be designed is a two-tier cellular network. The cells of the lower tier are called microcells and their radii are smaller than those of the cells of the upper layer. Upper layer cells, macrocells, cover an integer number of

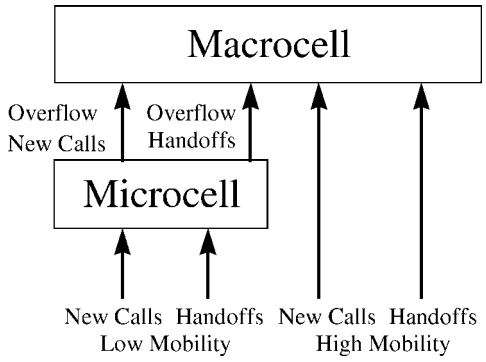

Figure 2. Call service pattern.

microcells. Furthermore, the radius of the macrocells is constrained to be an odd integer multiple of the radius of the microcells to ensue that an integer number of microcells are contained in every macrocell. The calculation of the number of covered microcells N_i, where $R/r = i$ can be expressed as

$$N_i = N_{i-4} + 6(i-2), \quad N_1 = 1, \qquad N_{-1} = 0. \tag{2.2}$$

The total channel spectrum is divided by the cluster size into channel sets. Each channel set is then divided among one microcell and one macrocell. This means that if the microcells have one less channel, then that channel will be used by the macrocells. After the splitting of the channel sets some of the channels will be assigned as guard channels. The number of guard channels cannot be changed once the system is operational.

The arriving calls are serviced as follows. The new calls of the slow MTs are primarily served by the appropriate microcells with an available non-guard channel. If only guard channels are available, then the new calls are overflowed to the macrocell that covers the microcell as *overflowed new call*. If handoff calls cannot be serviced in the microcells, then they are overflowed to macrocell. Figure 2 shows schematically how the calls are serviced and which order is followed in serving them.

At the macrocell level, all the calls of the high mobility terminals and the overflowed calls are serviced. The handoff calls of the high mobility class terminals and the overflowed handoff calls are treated equivalently. New calls of both classes may not use the guard channels upon their arrival. If no non-guard channel is available, new calls are blocked. The low mobility calls at macrocell level cannot return to microcell level even if channels become available in the microcells.

2.3. Constraints and parameters

The problem is to design a two-tier cellular network at lowest implementation cost. The performance parameters are chosen to be call blocking rate and call dropping rate. Any solution is said to be feasible as long as it is in accordance with the system description and the probability of call blocking and call dropping is below the specified thresholds.

Table 1
User-supplied parameters.

Parameter	Description
v_s	mean speed of low mobility users
v_f	mean speed of high mobility users
SAm^2	call arrival rate per second per m^2 for low mobility terminals
FAm^2	call arrival rate per second per m^2 for high mobility terminals
$1/U_t$	mean call duration
C_1	macrocell cost
C_2	microcell cost
A	total area
CS	cluster size
Ch_{total}	total number of available channels
P	radius increase/decrease factor
Cr	cooling rate
$P_{\text{b,max}}$	maximum allowable call blocking probability
$P_{\text{d,max}}$	maximum allowable call dropping probability

Table 2
Decision parameters.

Parameter	Description
C	total system cost
Ch_1	number of channels reserved for each microcell
Ch_2	number of channels reserved for each macrocell
G_1	number of guard channels reserved for each microcell
G_2	number of guard channels reserved for each macrocell
R	radius of a macrocell
r	radius of a microcell

The values for the system parameters are in the order of 10^{-2} to 10^{-3} for the call blocking and for the call dropping 10^{-3} to 10^{-4}. The input parameters are used to describe the call and physical medium characteristics and to set the performance requirements of the designer. These are used, in turn, to produce the parameters that describe the cellular system. The user-supplied parameters are presented in table 1 and the decision parameters are summarized in table 2.

2.4. Problem formulation

The minimum cost two-tier cellular network design problem can be formulated as follows:

$$\text{Minimize} \quad C = C_1 N_1 + C_2 N_2 \tag{2.3}$$

subject to the constraints

$$P_b \leqslant P_{b,\max}, \tag{2.4}$$

$$P_d \leqslant P_{d,\max}, \tag{2.5}$$

$$\pi R^2 N_1 \geqslant \text{Area}, \tag{2.6}$$

$$\pi r^2 N_2 \geqslant \text{Area}, \tag{2.7}$$

$$\frac{R}{r} = 2n + 1, \quad n \in Z. \tag{2.8}$$

The values C_1 and C_2 correspond to the cost of setting up a macrocell and a microcell, respectively. C is the total system cost. N_1 and N_2 are the number of the microcells

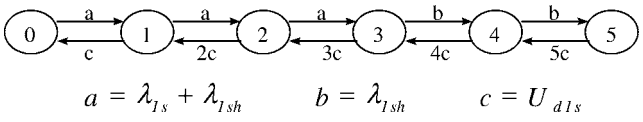

Figure 3. Sample state transition diagram for microcells (5 channels, 2 guard channels).

and macrocells respectively. R and r are the radii of macrocells and microcells, respectively. While minimizing the total system cost, the resulting probabilities for call blocking and call dropping should stay below the provided limits as expressed in inequalities (2.4) and (2.5). Inequalities (2.6) and (2.7), and equation (2.8) correspond to the coverage and integrality constraints. The calculation of the call blocking and dropping probabilities is explained in section 3.1.

3. Solution techniques

In order to solve the difficult optimization problem described in the last section primarily *Simulated Annealing* is used. In this section, we describe the details of the simulated annealing algorithm used as well as the description of the cost and performance calculation procedures.

3.1. Calculation of cost and performance measures

Given the total area, the number of microcells and macrocells that would cover the given total area is determined. If the resulting system has N_1 microcells and N_2 macrocells with C_1 and C_2 as their respective unit costs, the total cost C is computed as

$$C = C_1 N_1 + C_2 N_2. \tag{3.1}$$

In order to evaluate the performance of a configuration, the system is divided into two parts. The first part corresponds to the microcell layer of the system. This part of the system is represented with a Markov chain (M/M/s/s system) [26]. In this representation the state corresponds to the number of calls served by a microcell. The mean arrival rate to the system is denoted as λ_{1s}. λ_{1sh} is asymptotic handoff rate in the microcell tier. The dwell time of the low mobility users in the microcells U_{d1s} are calculated according to formula (2.1). A sample Markov chain for a microcell with five channels and two guard channels is presented in figure 3.

The steady state probabilities P_i can be calculated using Erlang-B formula [26] as shown in equations (3.2)–(3.4):

$$P_i = P_0 \left(\frac{a}{c}\right) \frac{1}{i!}, \quad i \leqslant Ch_1 - G_1, \tag{3.2}$$

$$P_i = P_0 \frac{a^{Ch_1 - G_1}}{c^i} \frac{b^{i - Ch_1 + G_1}}{n!}, \quad Ch_1 \geqslant i > Ch_1 - G_1, \tag{3.3}$$

$$P_0 = \left[\sum_{i=0}^{Ch_1} P_i \right]^{-1}. \tag{3.4}$$

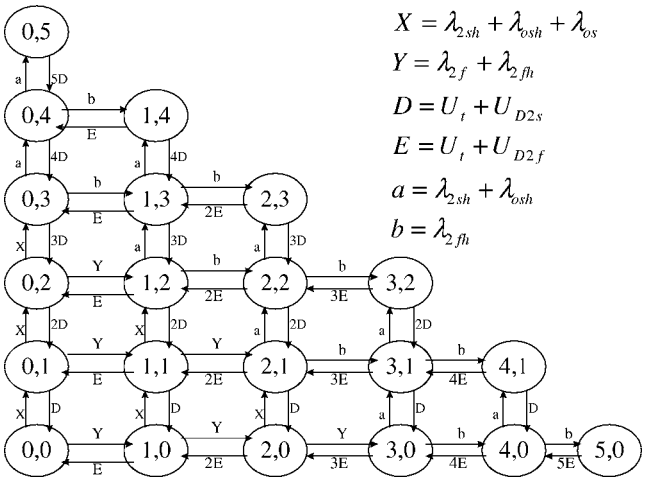

Figure 4. Sample state transition diagram for macrocells (5 channels, 2 guard channels).

The asymptotic handoff rate λ_{1sh} used in Erlang-B formula is calculated by iteration as described in [7], until Λ_{1sh} equals to λ_{1sh}:

$$\Lambda_{1sh} = \sum_{i=0}^{Ch_1} (i \cdot P_i \cdot U_{d1s}). \tag{3.5}$$

Having determined the probabilities for call blocking P_{b1} and call dropping P_{d1}, which correspond to the sum of the steady state probabilities for states $i = Ch_1 - G_1$ to Ch_1 and P_{ch1}, respectively, the total overflow new call and overflow handoff call traffic entering a macrocell (λ_{os} and λ_{osh}) covering N microcells can be calculated as follows:

$$\lambda_{os} = N \cdot \lambda_{1s} \cdot P_{b1}, \tag{3.6}$$

$$\lambda_{osh} = N \cdot \lambda_{1sh} \cdot P_{d1}. \tag{3.7}$$

The states in macrocell level correspond to the numbers of high and low mobility users (i and j) serviced by a macrocell. The state transition diagram for a macrocell with five ordinary and two guard channels is shown in figure 4.

The variables λ_{2f} and λ_{2fh} are the arrival rates of the new and handoff calls of high mobility users to a macrocell respectively. λ_{2sh} is the arrival rate of the handoff calls of the low mobility users once they entered the macrocells. $1/U_{d2s}$ and $1/U_{d2f}$ are the dwell times of low and high mobility users in the macrocells. The values for λ_{2fh} and λ_{2sh} are calculated using the same method that is described for the microcell handoff rate determination.

The system is solved for the steady state probabilities P_{ij} as described in [26]. For the real values of P_{ij} the handoff arrival rates should be calculated using an approach. The call blocking and call dropping probabilities for the macrocells (P_{b1} and P_{d1}) are calculated as follows:

$$P_{d1} = \sum_{i+j=Ch_2} P_{ij}, \tag{3.8}$$

$$P_{b1} = \sum_{i+j \geqslant Ch_2 - G_2} P_{ij}. \tag{3.9}$$

```
procedure SA
begin
reset the stopping criteria;
find initial feasible solution;
update the stopping criteria;
initialize temperature;
while termination criteria are not satisfied
begin
        generate neighbor;
        calculate cost;
        generate random number p;
        if (exp. (-(New cost - Old cost)/(Temperature)) > p) then
        if (Neighbor is feasible)
                then begin
                        accept the move;
                        reset the failure counter;
                        update the temperature;
                end;
                else begin
                        reject the move;
                        increase the failure counter;
                        update the stopping criteria with counter;
                end;
end;
end;
```

Figure 5. Pseudo-procedure of SA algorithm.

Finally, the overall call blocking P_{b} and call P_{d} dropping probabilities are calculated. The call blocking probability is calculated as in formula (3.10). The average call dropping rates for fast and slow mobility classes P_{ds} and P_{df} are calculated based on the same idea:

$$P_{\mathrm{b}} = \frac{N\lambda_{1\mathrm{s}} P_{\mathrm{b}1} P_{\mathrm{b}2} + \lambda_{2\mathrm{s}} P_{\mathrm{b}2}}{N\lambda_{1\mathrm{s}} + \lambda_{2\mathrm{s}}}. \tag{3.10}$$

3.2. Simulated annealing algorithm

The SA algorithm [24] is chosen because the design problem had multiple plateaus in the objective function. The algorithm starts with an initial feasible solution. The initial feasible solution is determined by inspecting randomly generated solutions until one is found. If all neighbors of an initial feasible solution are infeasible, the algorithm tries to find another feasible solution. The procedure proceeds with random moves within the range of the neighbors that can be reached from the current solution. Each step in the algorithm corresponds to visiting a feasible neighbor. Figure 5 shows the pseudo-procedure of the SA algorithm. The generation of the initial feasible solution is done by generating random radii for microcells and macrocells and by distributing the available channels among the tiers randomly. The available channels are split between macrocells and microcells randomly as well.

The neighbors of a system are determined by changing the decision parameters incrementally. The neighboring systems are either generated by transferring one channel from microcells to macrocells or vice versa, or by changing the microcell radius by a fixed percent value p supplied by the user, which effects also the macrocell radius or by changing the R/r ratio. The fixed ratio values constitute the plateau of the cost function. The SA algorithm is capable of searching beyond these plateaus for better solutions.

The SA algorithm stops in three different cases. First stopping criterion is the number of successive neighbors that

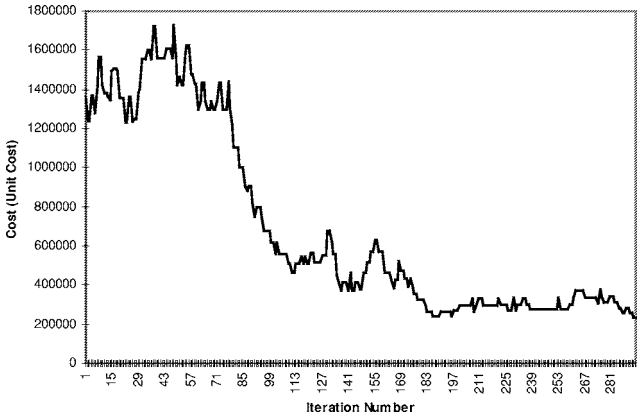

Figure 6. A typical run of SA algorithm.

are accepted due to their costs but do not confirm with the call blocking and dropping constraints. The second criterion is the number of moves that do not change the cost of the system. Finally, as a precaution, the algorithm counts also the number of moves made since it started. The maximum allowable number of moves is set to one thousand. The best value recorded is the output of the total algorithm.

The general behavior of the SA algorithm resembles the random search at the beginning and greedy search when the system is cooled down. Which of the mentioned techniques the SA algorithm is close to is determined by the cooling schedule. Figure 6 shows the cost of the solutions during a typical run of the SA algorithm.

3.3. Other search algorithms

In the literature, the multi-tier cellular network design is not addressed as an optimization problem. In order to compare the results that are obtained by the SA algorithm, other neighborhood search algorithms are also implemented. The Greedy Search (GS) algorithm is chosen for comparison because it is also a local search algorithm as our method. The greedy search algorithm is most of the time used for problems that have a convex cost function. If the cost function is not convex, like in this case, then the GS algorithm is run several times by starting from a different feasible solution. When the cost cannot be improved any further, then the algorithm is restarted. The Generate and Test (GAT) algorithm is chosen to provide statistical points-of-reference and assess the quality of the results of our SA-based technique.

Setting the initial temperature in the SA algorithm equal to zero yields in the GS algorithm. In order to set the approximate running times of SA and GS algorithms to be the same, the moves that are not accepted but for which the performance is calculated are also counted. The mean value turned out to be around 1500 for the SA algorithm. Therefore, the stopping criterion for the GS algorithm is chosen as 1500 performance calculations. Then the number of feasible moves lays around 620.

The aim of the GAT algorithm is to traverse the feasible solution space randomly. The values of the objective function obtained for these random solutions can be also used

Table 3
Parameters of the base problem.

Parameter	Value in the base problem
v_s	1 m/s
v_f	8 m/s
SAm^2	8×10^{-8} calls s^{-1} m^{-2}
FAm^2	2×10^{-8} calls s^{-1} m^{-2}
$1/U_t$	100 s
C_1	10 cost units
C_2	30 cost units
A	5000 km^2
CS	7
Ch_{total}	150
$P_{b,max}$	0.01
$P_{d,max}$	0.001

as a statistical quality measure of the heuristic algorithms. There is no rule that relates the generation of the successive points in the solution space. For the generation of the feasible solution the routine that calculates the initial feasible solution for the SA algorithm is used. In order to statistically assess the quality of the solutions produced by our algorithm, we used the GAT algorithm to generate 5000 feasible solutions. In order to observe 5000 feasible solutions, GAT algorithm generated 25000 random solutions on the average.

4. Computational experiments

The computational experiments are performed to assess the effect of the system parameters and the network load on the implementation cost. The first group of experiments deals with the effect of the isolated parameters on the objective function, i.e., the implementation cost. Secondly, the effects of the design decisions are examined. Lastly, the selected solution technique is compared with other alternative solution techniques and the single-tier systems. The obtained results are presented graphically.

In order to prepare the problem sets for comparison, a sample problem is chosen as the base problem. This base problem reflects a typical case that can be faced when designing a cellular network. The experiment sets are prepared by changing the values of the given system. To demonstrate the effect of a parameter, all parameters except for the parameter in question are selected equal to the parameter values of the base problem. Parameters of the base problem can be taken from table 3.

The resulting solution that is obtained from the SA algorithm gives all the necessary decision parameters needed to implement the two-tier cellular network. The values of the decision parameters are presented in table 4.

4.1. Effect of some parameters and design decisions

When the effect of the call arrivals on the cost is examined, first the call arrival rate of the low mobility platforms is increased from 5×10^{-8} to 15×10^{-8} calls $s^{-1}m^{-2}$ with a step size of 1×10^{-8} calls $s^{-1}m^{-2}$. The results obtained are represented in figure 7 graphically. The results obtained show that the effect of the low mobility call arrival rates on the cost are almost linear around the given parameters. The increase in the traffic is directly reflected to the cost.

Table 4
Values of decision parameters for the base problem.

Parameter	Result of SA algorithm
C	152060 cost units
Ch_1	8
Ch_2	13
G_1	0
G_2	1
R	1275 m
r	425 m
R/r	3

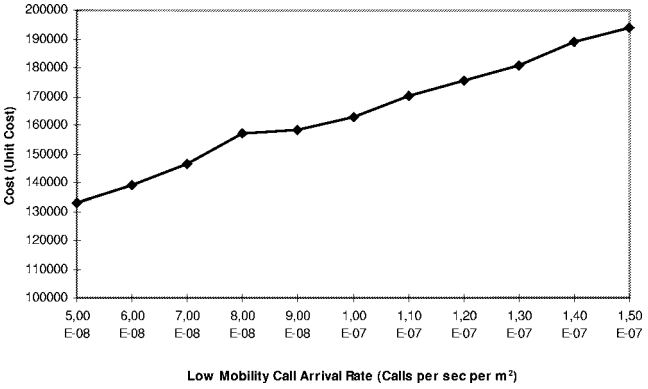

Figure 7. Low mobility call arrival rate versus cost.

Similarly, the call arrival rate of the high mobility platforms is examined. The remaining parameters are as in the base problem. The arrival rates are increased from 0.5×10^{-8} to 5×10^{-8} calls $s^{-1}m^{-2}$ with a step size of 0.5×10^{-8} calls $s^{-1}m^{-2}$. Increasing the arrival rate of high mobility calls show an identical behavior as the low mobility case. It can be concluded from these observations that the system cost increases almost linearly with the increasing rate of the arrival rate. The reason for this is the increasing number of microcells and macrocells due to decreasing cell radii. More cells fit into a given area, increasing the cost of the entire system. The main distinction between both mobility classes is the slope of the curves. It has been observed that the two-tier cellular network costs are more sensitive to the changes in the high mobility call arrivals.

The second parameter to focus on is the mean speed of both mobility classes. The speed changes of both mobility classes are examined separately. Keeping the other parameters as in the base problem, the mean speed of the low mobility users is increased from 0.25 to 2.5 m/s with a step size of 0.25 m/s. The resulting graph is shown in figure 8.

The cost of the system increases when the mean speed of the low mobility users increases. However, this increase starts with a higher slope for small values of mean speed and becomes less and less when the mean speed increases. This behavior may be an indication of the fact that the system

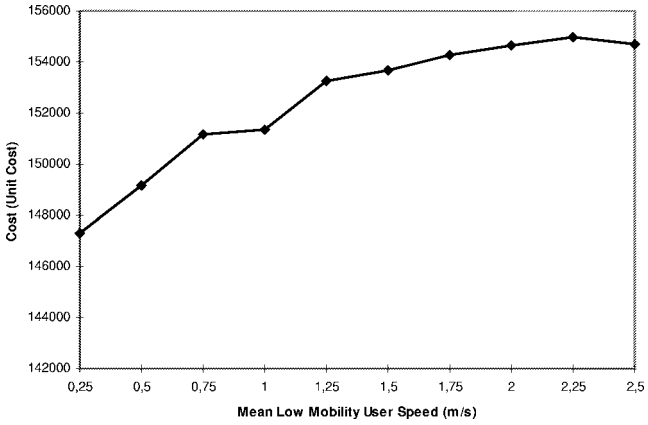

Figure 8. Mean low mobility user speed versus cost.

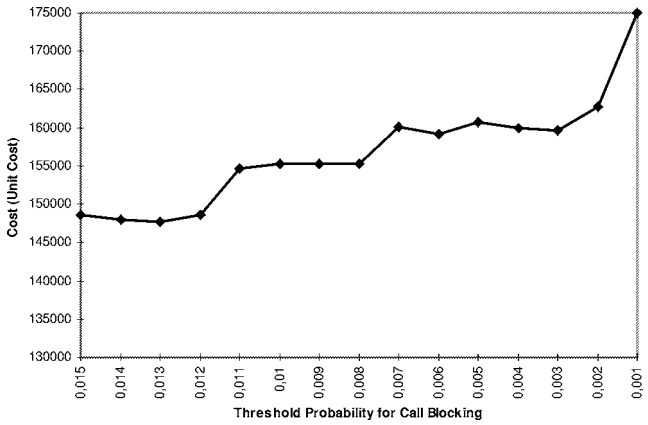

Figure 9. Threshold probability for call blocking versus cost.

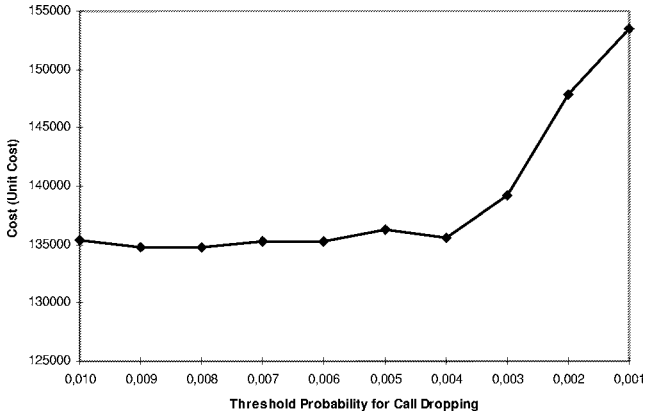

Figure 10. Threshold probability for call dropping versus cost.

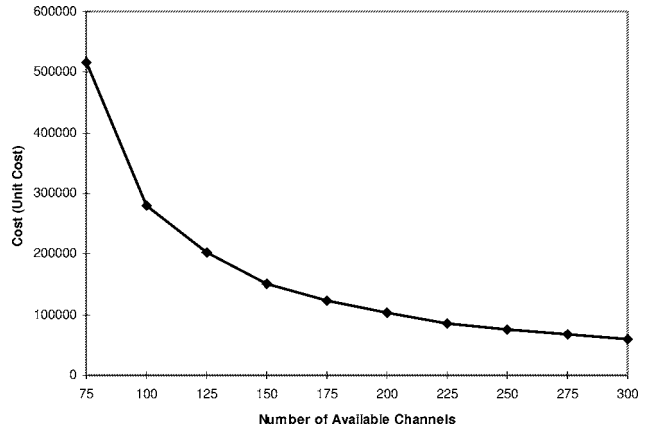

Figure 11. Number of available channels versus cost.

starts to benefit from the second tier of the network as a place to forward the excess calls instead of decreasing the cell radii and increasing the channel amount per m^2.

When the average speed of the high mobility users is increased, since the arriving high mobility calls cannot be overflowed to another tier, the system should react with incorporating more resources by paying more. In the previous case, the system could use the excess resources available in the upper tier. In this case, the excess traffic should be handled again within the same tier. Hence, the performance requirements can be met only if the number of available channels per unit area is increased. It can also be concluded that increases in the mobility do not effect the system setup cost too much, since the users change cells equally likely and the lack of resources are temporary.

The maximum allowable call blocking probability is one of the constraints of the problem. In order to demonstrate the effect of maximum allowable call blocking probability, its value is decreased starting at 0.015 down to 0.001 with a step size of 0.001. The results of this experiment are shown graphically in figure 9. As the constraint gets stricter, the cost of the system goes over to different plateaus. It does not make any difference to choose a point belonging to a plateau. The value for maximum allowable call blocking threshold should be chosen such that it is as strict as possible within the limits of a plateau, since relaxing the constraints has no effect on cost as long as one stays on a plateau.

The next parameter to be inspected is the maximum allowable call dropping threshold. The value for this threshold is decreased from 0.01 to 0.001 with a step size of 0.001. The resulting values are shown in figure 10. For a given call blocking rate, the call dropping rate has no effect on the cost until it reaches a certain value. Only if it is set to be smaller than that value, the cost of the system starts to increase. The break point is set primarily by the call blocking rate. If that call blocking rate is satisfied, then the call dropping thresholds that are larger than the break point are satisfied anyway.

The size of the area only effects the number of cells necessary, its effect is only on the cost. The relationship is linear and the decision parameters are not affected. A similar, linear behavior is observed also for increased values of the mean call duration.

In order to observe the effect of additional resources on the system, the number of total available channels is increased from 75 to 300 with 25-channel increments. Figure 11 summarizes the results. Increasing the number of available channels decreases the system setup cost. The gain per unit increase in the number of channels is high where the channel numbers are small. The increase decreases for larger numbers of available channels. The cost differences indicate the need for additional channels to relieve the system.

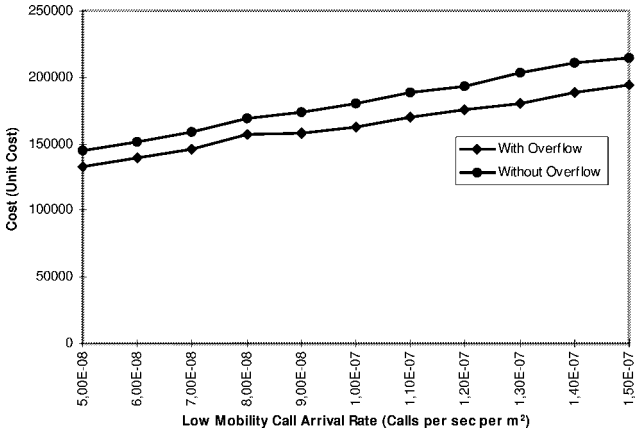

Figure 12. Low mobility call arrival rate versus cost (overflow comparison).

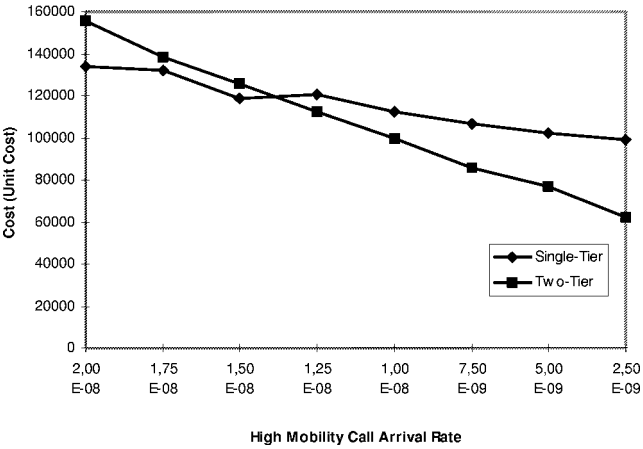

Figure 13. High mobility arrival rate versus single- and two-tier costs.

One of the design decisions was to allow the overflow of the calls from the lower layer to the upper layer. This decision is supposed to improve the performance of the system. In order to demonstrate this behavior, tests are performed. Call arrival rate increase tests are also applied to the sample system with no overflow allowed. Figure 12 presents the effect of the increase in low mobility call arrival rate. Similar observations are also made for the high mobility call arrival rate increase tests. The system with overflow yields better solutions in terms of cost. The gain in cost obtained by allowing the calls to overflow to the upper layer is almost constant for both cases. There is only a small increase in the cost gap towards the end of the values tested. Hence, these results justify the use of overflow mechanism.

In order to see the effect of the decreasing high mobility call arrival rates, we experimented with the systems, where the arrival rate decreases from its value in the base problem with 2.5×10^{-9} calls $s^{-1}m^{-2}$. This is done in order to increase the ratio of the low mobility call arrival rate to the high mobility call arrival rate. For values that are less than 1.25×10^{-9} calls $s^{-1}m^{-2}$, the two-tier system setup cost becomes less than the single-tier system cost. As expected, the system cost decreases when the load on the system decreases. However, the two-tier system has a cost graph with

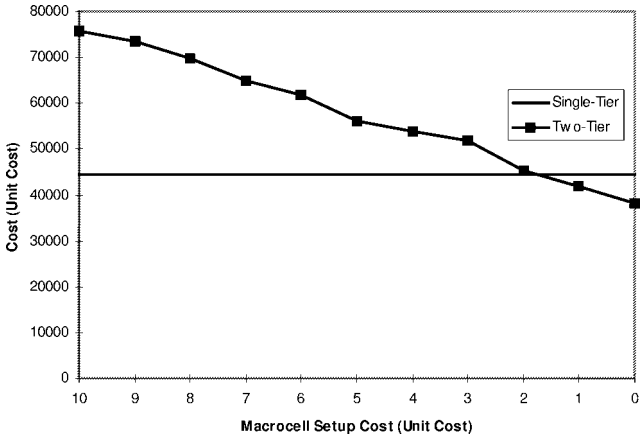

Figure 14. Macrocell setup cost versus single- and two-tier costs.

a more negative slope. Hence, the two-tier system is indeed cost effective for the problems for which the high mobility call arrival rates are far less than the low mobility call arrival rate. The cost curves for both systems are presented in figure 13.

In order to see the effect of the macrocell setup costs, the base problem is modified such that the macrocell cost is decreased from 10 unit costs to 0 unit costs as shown in figure 14. It is assumed that the microcell layer is already present and for each macrocell, an additional price is paid. When the additional macrocell setup cost is less than 20% of the microcell cost, the two-tier system becomes cost effective. A similar behavior is observed when the macrocell cost is five times the microcell setup cost. Hence, it can be deduced from these results that the cost ratios less than 0.2 or greater than 5.0 causes systems like the one described in the base problem be handled cheaper with two-tier cellular networks.

4.2. Assessment of the SA algorithm

In this section, the results of several tests are presented. All of these tests are based on the base problem defined in the introduction of section 4. In order to demonstrate the behavior of the two-tier cellular network in question, several nominal and extreme parameter configurations are tested and solutions are compared with other solution techniques.

In order to compare the GS and SA algorithms fairly, the basic SA algorithm is run until 1000 accepted moves are reached. This may correspond to multiple runs of the basic SA algorithm. It is fair to run the basic SA algorithm more than once, since the results do not change very much with increasing number of runs. By doing this, the number of evaluated solutions is determined approximately. Then the number of performance evaluations in the GS algorithm is set equal to the mean number of performance evaluations in the SA algorithm. Hence, the running times of both algorithms are normalized.

All the problems presented are variations of the base problem. The changed parameter is given in the description column as well as in the text. The obtained results are shown

Table 5
Results obtained with SA, GS and GAT algorithms for several problems.

Problem No.	Description	SA	GS	GAT Min	GAT Avg	GAT Max
1	Base problem	152060	155390	151890	2118037	10895180
2	Area = 1000 km^2	30990	30380	30570	429610	2193440
3	Area = 50 km^2	1550	1660	1550	20789	108860
4	Slow low mobility, speed = 0.25 m/s	147000	154050	148950	2099563	10901150
5	Fast low mobility, speed = 3 m/s	157200	161260	157360	2145840	10925040
6	Slow high mobility, speed = 5 m/s	149900	154540	150460	2094752	10889240
7	Fast high mobility, speed = 20 m/s	162440	165560	164150	2204543	10942990
8	Low mobility, arrival rate = 2×10^{-8}	112590	118280	112300	2041311	10967040
9	Low mobility, arrival rate = 1.5×10^{-7}	195400	195860	193880	2278864	10919060
10	High mobility, arrival rate = 5×10^{-9}	76990	77380	92300	1325920	10913080
11	High mobility, arrival rate = 8×10^{-8}	489980	774880	484280	3516816	10997170
12	Frequent call arrivals	646790	651960	645940	3765970	10979070
13	More high mobility users	218300	212540	213240	2529019	10948990
14	Short duration = 40 s	54580	56100	93250	1393697	10818160
15	Long duration = 200 s	356570	372160	338460	2952800	10954990
16	Loose performance limits	124360	123950	120970	1901178	10973050
17	Strict performance limits	161180	165680	161100	2177883	10979070
18	Very strict performance limits	172760	180290	173660	2261686	10942990
19	Cluster size = 3	47690	60690	89760	1400909	10889240
20	Cluster size = 19	1054040	1069140	1052850	4531172	10991140
21	Same unit price	121960	123350	120960	1920096	8792900
22	Price ratio = 1 : 5	181300	188020	182160	2301842	13196610
23	No macrocell setup cost	37500	38020	105990	1821174	7698010

in table 5. First, we focused on the effect of the area. The results for different areas are calculated. The base problem is chosen as a large area. A medium sized city of 1000 km^2 and a part of the city of size 50 km^2 are chosen as targets in problems 2 and 3, respectively. When the input area is changed, the solutions remain actually the same except for the total number of cells that are needed. This information can easily be extracted from the cost ratios. The ratios of the areas for the problems one, two and three behave like 100 : 20 : 1. Almost the same ratio is observed for the cost values in the SA column. One can see that the radii and channel partitioning are the same for all three problems. Hence it can be concluded that the area has no effect on the decision variables but the cost.

In problems 1 and 3, the GS algorithm finds slightly worse solutions whereas it finds a lucky starting point for the second problem and finds a slightly better result. When 5000 feasible solutions are generated, the best results obtained are close to the solutions found by the SA algorithm. The mean and the maximum costs are examples for how large the feasible solution space can be. For the first problem, the mean rate is 14 times larger than the best solution. Similar ratios can be found also for other problems.

The terminal mobility is a parameter that changes the number of handoffs during a call. In problems 4–7, cases with different mobility rates are examined. For these cases, two parameters are altered independently. For the low mobility terminals, slow, moderate and high mean speeds are selected as 0.25, 1 and 3 m/s, respectively. For the high mobility terminals, slow, moderate and high mean speeds are selected as 5, 8 and 20 m/s, respectively. Problem 1 is the moderate speed rate for both mobility class comparisons. For the low mobility mean speeds, the system cost increases as the mean speed increases. In problems 1, 4 and 5, as the mobility increases, the radii of cells in both layers decrease in order to increase the channels available per m^2. Hence, the system creates more resources for frequent handoffs. In this case, the frequent handoffs may create problems in the signaling part of the network, which is out of scope of this work.

The solutions generated by the GAT algorithm are worse in problems 4–7. The case for the GS algorithm is on the same course: none of the solutions obtained outperform the results of the SA algorithm. Hence, the SA algorithm produces better results in considerably less time. The call arrival rates are the main source of load on the cellular networks. In problems 8–13, the arrival rates originated by users of both mobility classes are changed to observe changes in the cost. The low, moderate and high call arrival rates of the low mobility class users are selected as 2×10^{-8}, 8×10^{-8} and 1.5×10^{-7} calls s^{-1} m^{-2}, respectively. The low, moderate and high call arrival rates of the high mobility class users are selected as 5×10^{-9}, 2×10^{-8} and 8×10^{-8} calls s^{-1} m^{-2}, respectively. The moderate arrival rates for both cases correspond to the base problem. In addition to these problems, another problem is inspected, where the load is very high. For this problem, the low mobility arrival rate is 3×10^{-7} and the high mobility arrival rate is 8×10^{-8} calls s^{-1} m^{-2}. Another case, where the low mobility arrival rate is less than the high mobility case, is presented. The call arrival rates are 2×10^{-8} and 4×10^{-8} calls s^{-1} m^{-2} for low and high mobility users, respectively. As expected, as the call arrival rates increase, the system cost also increases. Furthermore, according to the results of the problems 1, 9 and 11, when the call arrival rates of the high and low mobility users is multiplied approximately by two, the system cost is effected

most for the increase of the arrival rates of the high mobility calls.

In most of the tested problems, SA algorithm yields better results than the GS algorithm, sometimes with big differences. The solutions of the GAT algorithm indicate that the solutions of the SA algorithm are not very far from the best solutions found. Indeed, in problems 8, 9 and 11–13, the best solutions for these problems belong to the GAT algorithm.

Another parameter that affects the load on the system is the duration of the calls. In the test problems 1, 14 and 15, three different call duration values are considered. The mean duration of short calls is 40 s. Medium length calls have a mean duration of 100 s, which corresponds to the base problem. Long calls last 200 s on the average. The solutions of the SA algorithm are better than the solutions of the GS algorithm in all three cases. The results of the GAT algorithm indicate that the system is capable of finding better results for shorter mean call duration values with both the GS and GAT algorithms. This may be caused by the increased search space created by the decreased call duration.

The maximum allowable call blocking and dropping thresholds set the limits of the solution space. In problems 1, 16, 17 and 18, four different models are considered. Problem 16 corresponds to loose limits with $P_{\mathrm{b,max}} = 0.02$ and $P_{\mathrm{d,max}} = 0.01$. The normal limits are the limits used in the base problem. The performance limits are $P_{\mathrm{b,max}} = 0.01$ and $P_{\mathrm{d,max}} = 0.001$. The strict limits are $P_{\mathrm{b,max}} = 0.01$ and $P_{\mathrm{d,max}} = 0.0001$ in problem 17 and very strict limits are $P_{\mathrm{b,max}} = 0.005$ and $P_{\mathrm{d,max}} = 0.0001$ in problem 18. The solutions obtained with the GS algorithm are worse except for problem 16. From the comparisons with the GAT algorithm, it can be concluded that the success of the SA algorithm increases as the performance limits become stricter, since the SA algorithm makes a more intelligent search than the GS and GAT algorithms.

Other factors affecting the solutions are cluster size that is used and the cost ratio of the microcells and macrocells. In problems 19, 1 and 20, cluster sizes are chosen as 3, 7 and 19. Problems 21, 1 and 22 correspond to the cost ratios 1 : 1, 1 : 3 and 1 : 5. In addition to these ratios, the case where the microcell antennas are capable of handling both tiers are considered. In the latter case, there is no macrocell setup cost. Since the number of available channels is split into smaller chunks when cluster size increases, the cost of the system also increases. The GS algorithm never produces any better results than the SA algorithm for different cluster sizes. The GAT algorithm fails in cases where the system has much larger capacity than the offered load, as in problem 19. In other problems, the GAT and SA algorithms produce results close to each other.

5. Conclusion

In this work, we concentrated on the design of a two-tier cellular network. The network has single direction call overflow capability and employs guard channels. The resulting cellular networks are generated by minimizing the cost. The performance constraints are the maximum allowable call blocking and call dropping probabilities. In order to find near-optimum solutions, the SA algorithm is used. The performance of the SA algorithm is compared with the performance of the GS algorithm and the GAT method.

Computational experiments showed that the SA algorithm outperforms the GS algorithm in most of the cases. The quality of the solutions is compared with the results of the GAT method. The cost values obtained with the SA algorithm are either better than the solutions of GAT, or they are very close to each other. The effects of several parameters on the cost of the system are studied. Additionally, the effect of design decisions, having guard channels and allowing call overflow, are also considered.

As a future work, we plan to investigate alternative solution techniques, including Taboo Search and Genetic Algorithms. The time complexity of the algorithm will be studied in detail. Furthermore, different neighborhood functions will be developed for the SA algorithm. Results in this study were for the two-tier systems. However, the SA algorithm can be modified to handle more tiers.

References

[1] R.H. Katz, Adaptation and mobility in wireless information systems, IEEE Personal Communications Magazine 1(1) (1994) 6–17.

[2] M. Shafi, A. Hashimoto, M. Umehira, S. Ogose and T. Murase, Wireless communications in the twenty-first century: A perspective, Proceedings of the IEEE 85(10) (October 1997) 1622–1639.

[3] T.S. Rappaport, *Wireless Communications* (Prentice-Hall, Englewood Cliffs, NJ, 1996).

[4] M.F. Catedra and J.P.-Arriaga, *Cell Planning for Wireless Communications* (Artech House, 1999).

[5] N. Passas, S. Paskalis, D. Vali and L. Merakos, Quality-of-Service-oriented medium access control for wireless ATM networks IEEE Communications Magazine (November 1997) 42–50.

[6] M. Naghshineh and M. Willebeck-LeMair, End-to-end QoS provisioning in multimedia wireless/mobile networks using an adaptive framework, IEEE Communications Magazine (November 1997) 72–81.

[7] L. Hu and S.S. Rappaport, Personal communication systems using multiple hierarchical cellular overlays, IEEE Journal on Selected Areas in Communications 13(2) (February 1995) 406–415.

[8] R. Coombs and R. Steele, Introducing microcells into macrocellular networks: A case study, IEEE Transactions on Communications 47(4) (April 1999) 568–576.

[9] K. Buchanan, R. Fudge, D. McFarlane, T. Phillips, A. Sasaki and H. Xia, IMT 2000: Service provider's perpective, IEEE Personal Communications 4 (August 1997) 8–13.

[10] I.F. Akyildiz, J. McNair, J.S.M. Ho, H. Uzunalioglu and W. Wang, Mobility management in next-generation wireless systems, Proceedings of IEEE 87 (August 1999) 1347–1384.

[11] W. Wang and I.F. Akyildiz, Intersystem location update and paging schemes in multitier wireless networks, in: *Proceedings of IEEE MobiCom 2000* (August 2000) pp. 99–109.

[12] Ş. Tekinay and B. Jabbari, A measurement-based prioritization scheme for handovers in mobile cellular networks, IEEE Journal on Selected Areas in Communications 10(8) (October 1992) 1343–1350.

[13] M. Sidi and D. Starobinski, New call blocking versus handoff blocking in cellular networks, in: *Proceedings of the Conference on Computer Communications (IEEE INFOCOM '96)* (1996).

[14] L. Ortigoza-Guerrero and A.H. Aghvami, *Resource Allocation in Hierarchical cellular Systems* (Artech House, Boston/London, 2000).
[15] S. Oh and D. Tcha, Prioritized channel assignment in a cellular radio network, IEEE Transactions on Communications 40(7) (July 1992) 1259–1269.
[16] M. Oliver and J. Borras, Performance evaluation of variable reservation policies for hand-off prioritization in mobile networks, in: *Proceedings of IEEE INFOCOM '99*, Vol. 3 (1999) pp. 1187–1194.
[17] S. Choi and K. Sohraby, Analysis of a mobile cellular system with hand-off priority and hysteresis control, in: *Proceedings of IEEE INFOCOM 2000*, Vol. 1 (March 2000) pp. 217–224.
[18] Y. Ma, J.J. Han and K.S. Trivedi, Call admission control for reducing dropped calls in code division multiple access (CDMA) cellular systems, in: *Proceedings of IEEE INFOCOM 2000*, Vol. 3 (March 2000) pp. 1481–1490.
[19] T.-S.P. Yum and W.-S. Wong, Hot-spot traffic relief in cellular systems, IEEE Journal on Selected Areas on Communications 11(6) (August 1993) 934–939.
[20] I. Katzela and M. Naghshineh, Channel assignment schemes for cellular mobile telecommunication systems: A comprehensive survey, IEEE Personal Communications (June 1996) 10–31.
[21] J. Li, N.B. Shroff and E.K.P. Chong, The study of a channel sharing scheme in wireless cellular networks including handoffs, in: *Proceedings of IEEE INFOCOM '99*, Vol. 3 (1999) pp. 1179–1186.
[22] A. Ganz, C.M. Krishna, D. Tang and Z.J. Haas, On optimal design of multitier wireless cellular systems, IEEE Communications Magazine 35(2) (February 1997) 88–93.
[23] S. Faruque, Science, engineering and art of cellular network deployment, in: *Proceedings of the IEEE Conference on Personal, Indoor, Mobile, Radio Communications (PIMRC'98)*, Boston (September 1998).
[24] S.P. Brooks and B.J.T. Morgan, Optimization using simulated annealing, The Statistician 44 (1995) 241–257.
[25] L. Ortigoza-Guerrero and A.H. Aghvami, On optimal spectrum partitioning in a microcell/macrocell layout with overflow, in: *Proceedings of the IEEE Global Telecommunications Conference (IEEE GLOBECOM '97)*, Arizona (1997).
[26] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. (Prentice-Hall, Englewood Cliffs, NJ, 1992).

Eylem Ekici received his BS and MS degrees in computer engineering from Boğaziçi University in 1997 and 1998, respectively. Currently, he is enrolled in the PhD program of the School of Electrical and Computer Engineering, Georgia Institute of Technology. His research interests include satellite communications, wireless networks and routing protocols.

Cem Ersoy received his BS and MS degrees in electrical engineering from Boğaziçi University in 1984 and 1986, respectively. He received his PhD in electrical engineering from Polytechnic University in 1992. Currently, he is an Associate Professor in the Computer Engineering Department of Boğaziçi University. His research interests include performance evaluation and topological design of communication networks, wireless and multimedia communications.
E-mail: ersoy@boun.edu.tr