Telecommunication Systems 14 (2000) 95–106

A channel allocation for cellular mobile radio systems using simulated annealing

Soo-Hyun Kim [a], Kun-Nyeong Chang [b] and Sehun Kim [c]

[a] *Techno-Economics Department, Electronics and Telecommunications Research Institute, 161 Kajong-Dong, Yusong-Gu, Taejon 305-350, Korea*
E-mail: kimsoo@etri.re.kr

[b] *Department of Management Information Systems, Silla University, 1-1 Kwaebop-Dong, Sasang-Gu, Pusan 617-736, Korea*
E-mail: knchang@lotus.pwu.ac.kr

[c] *Department of Industrial Engineering, Korea Advanced Institute of Science and Technology, 373-1 Kusong-Dong, Yusong-Gu, Taejon 305-701, Korea*
E-mail: shkim@mgt.kaist.ac.kr

We consider the channel allocation problem, which is one of the most interesting problems in mobile radio systems. This problem is known to be NP-complete and a couple of heuristic algorithms have been developed. In this paper, we convert the problem into a simpler form through the concept of pattern, a set of cochannel cells. We suggest another algorithm based on simulated annealing for this simplified problem. The algorithm is applied into different benchmark problems that have appeared in the literature. The presented examples illustrate that our method works very well. Computational results using our formulation and simulated annealing algorithm are reported.

1. Introduction

The increasing demand for mobile communication services and a finite spectrum allocated to such services, call for a new generation of technology to meet future demand. Many new modulation, therefore, and multiple access techniques have been developed. However, for a given spectrum and a specific technology used, the traffic-carrying capacity of a cellular system depends on how the frequency channels are managed [10]. The used term “channel” is general. This channel could be a fixed radio frequency (FDMA), a specific time slot within a frame (TDMA), or a particular code (CDMA), depending on the multiple access technique used by the system [5].

There are two aspects of channel management. A set of channels is allocated to each cell on a nominal basis and each cell assigns a specific channel to carry a specific call. The specific channel being assigned normally is a nominal channel of that cell. But if none is available, borrowing from a neighboring cell may be allowed. A variety of channel assignment strategies are available [13].

Besides the channel borrowing strategies, a good allocation of nominal channels to the cells can also significantly increase the traffic-carrying capacity of a system [14]. This is because with good nominal channel allocation, the need for channel borrowing will be reduced. Thus, in this paper, we consider the channel allocation problem. The problem can be stated as follows: allocate nominal channels to the cells in such a way that the average blocking in the entire system is minimized. Given the traffic distribution of a cellular mobile system, searching the optimal nominal channel allocation pattern is known as an NP-complete graph coloring problem. Since it is unlikely to find a polynomial algorithm which gives the optimal allocation pattern, heuristic algorithms that can give near-optimal solutions are needed.

Most previous investigations concerning the channel allocation problem were based on graph theoretic or heuristic approaches [1,2,4,7]. The graph theoretic approach has several disadvantages in applicability and flexibility [8]. Investigations based on neural networks [8] yield excellent results in special cases, but under certain conditions, principally only suboptimal solutions can be found [9].

In this paper, we propose a quite satisfactory approach called SA based on *simulated annealing* for the allocation of nominal channels. This method has been applied by Duque-Antón et al. [3], but they use a completely different model. In section 2, the optimal channel allocation problem is defined and formulated mathematically. Also a reduced problem using the concept of pattern is suggested. The solution procedure using simulated annealing is described in section 3. In section 4, heuristic procedures for generating candidate patterns considered in the reduced problem are suggested. The results of performance comparison are reported in section 5.

2. The optimal channel allocation problem

Suppose that we have a cellular system consisting of N cells and M channels. Let λ_i be the traffic demand in erlangs of cell i and let the number of channels available in the cell be m_i, then the call blocking probability in the cell is given by the Erlang B formula as

$$B(\lambda_i, m_i) = \left[\sum_{k=0}^{m_i} \frac{\lambda_i^k}{k!}\right]^{-1} \frac{\lambda_i^{m_i}}{m_i!}.$$

The weighted average blocking probability in the cellular system is given by

$$R = \sum_{i=1}^{N} w_i B(\lambda_i, m_i),$$

where $w_i = \lambda_i / \sum_{h=1}^{N} \lambda_h$ is the traffic weighting factor [6,14]. The channel allocation problem (CAP) which minimizes the weighted average blocking probability subject to cochannel interference constraints is as follows [6]:

$$\min \sum_{i=1}^{N} w_i B(\lambda_i, m_i)$$

$$\text{s.t.} \quad m_i = \sum_{j=1}^{M} f_{ij}, \quad \text{for } i = 1, \dots, N, \tag{1}$$

$$f_{sj} + f_{tj} \leqslant 1, \quad \text{for } j = 1, \dots, M \text{ and all interfering cell pairs } (s,t), \tag{2}$$

$$f_{ij} = 0 \text{ or } 1, \quad \text{for } i = 1, \dots, N \text{ and } j = 1, \dots, M. \tag{3}$$

This is a nonlinear integer programming problem. The decision variable f_{ij} is a binary integer variable indicating channel allocation, where $f_{ij} = 1$ represents that channel j is allocated to cell i and $f_{ij} = 0$ otherwise. Equation (1) states that for each cell i, the sum of the individual channels allocated to cell i equals the decision variable m_i, the number of channels available to cell i. Equation (2) means that the same channel j should not be allocated to different cells s and t simultaneously, if the cells s and t are within interference zone of each other.

The problem (CAP) is known to be NP-complete. The number of binary integer variables in the problem (CAP) is MN. In the case of $M = 100$ and $N = 50$, to get the optimal solution of the problem (CAP), it is necessary to enumerate explicitly or implicitly all possible combinations of size $2^{MN} \approx 10^{1500}$. Even if all the infeasible choices are eliminated, the resulting number is still astronomical. Besides, the cochannel interference constraints (2) are very complex. Thus it may be impossible to find any polynomial algorithm to solve the problem (CAP) optimally.

To deal with the proposed problem more conveniently, we introduce the concept of pattern. A channel cannot be allocated to adjacent cells simultaneously because of the cochannel interference. If a channel is allocated to some cells without causing cochannel interference, these cells are called the cochannel cells of that channel. This set of cochannel cells forms a pattern. The minimum distance between cochannel cells is called the minimum reuse distance.

Now, suppose that we generate P patterns such that every cell belongs to at least one of these patterns. Then the channel allocation problem reduces to the problem of allocating M channels to P patterns. Let the decision variable x_p denote the number of channels allocated to pattern p. Then using the P patterns, the problem (CAP) reduces to the following problem (CAP_1):

$$\min \sum_{i=1}^{N} w_i B(\lambda_i, m_i)$$

$$\text{s.t.} \quad m_i = \sum_{p \in S_i} x_p, \quad \text{for } i = 1, \dots, N, \tag{4}$$

$$\sum_{p=1}^{P} x_p \leqslant M, \tag{5}$$

$$x_p = 0, 1, 2, \ldots, M, \quad \text{for } p = 1, \ldots, P, \tag{6}$$

where S_i is the set of patterns which cover cell i. Equation (4) states that the number of channels available to cell i is equal to the sum of the individual channels allocated to the patterns that cover cell i. Equation (5) means that the total number of channels allocated to the patterns generated is less than or equal to the number of channels available in the system. Due to the property of a pattern, a solution of the problem (CAP_1) always satisfies the cochannel interference constraints. If all feasible patterns are considered, the problem (CAP_1) is equivalent to the original problem (CAP), and the combinations that these patterns generate are similar to the combinations generated by the original formulation.

However, this is not true if only special candidate patterns, for example, the patterns generated by the pattern generation procedures in section 4, are considered. In the case of $M = 100$, $N = 50$, and $P = 50$, to get the optimal solution of the problem (CAP_1), it is necessary to enumerate explicitly or implicitly all nonisomorphic combinations of size

$$\binom{M+P-1}{P-1} \approx 6.7 \times 10^{39},$$

which is significantly less than $2^{MN} \approx 10^{1500}$. This property is not dependent on the specifics of the cell topology, if only special candidate patterns are considered.

3. Simulated annealing

Simulated annealing is a general method for the approximate solution of difficult (i.e., NP-complete) combinatorial optimization problems. It has been applied in such diverse areas as computer aided design of integrated circuits, imageprocessing, code design, etc. [3].

Generally, a combinatorial optimization problem consists of a set $\mathbf{X}$ of solutions and a cost function C which determines for each $\mathbf{x} \in \mathbf{X}$ the cost $C(\mathbf{x})$, i.e., a real number. Simulated annealing can be considered as a generalization of the iterative improvement scheme (local search). For performing a local search, one needs to know the neighbors $\bar{\mathbf{x}}$ of $\mathbf{x}$. Thus one has to define a neighborhood structure N_e on $\mathbf{X}$. $N_e(\mathbf{x})$ determines, for each solution $\mathbf{x}$, a set of possible transitions which can be proposed by $\mathbf{x}$.

For classical local search, starting from an arbitrary solution $\mathbf{x}$, in each step of iterative improvement, a neighbor $\bar{\mathbf{x}}$ of $\mathbf{x}$ is proposed at random. Then, $\mathbf{x}$ is replaced by $\bar{\mathbf{x}}$ only if cost does not rise, i.e., $C(\bar{\mathbf{x}}) \leqslant C(\mathbf{x})$. Obviously, this procedure terminates in a local minimum, i.e., in a solution whose neighbors do not offer any improvement in cost. Unfortunately, such a local minimum may have a substantially higher cost than the global one.

To avoid this trapping in poor local optima, simulated annealing occasionally allows solutions of higher cost according to the Metropolis criterion [12]. More precisely, if $\mathbf{x}$ and $\bar{\mathbf{x}} \in N_e(\mathbf{x})$ are the two solutions, then the algorithm continues with solution $\bar{\mathbf{x}}$ with a probability given by $\min\{1, \exp(-(C(\bar{\mathbf{x}}) - C(\mathbf{x}))/T)\}$ (acceptance probability), where T is a positive control parameter called "temperature" and is gradually decreased to zero during the execution of the algorithm. Note that the acceptance probability decreases for increasing values of $C(\bar{\mathbf{x}}) - C(\mathbf{x})$ and for decreasing values of T, and that cost-decreasing transitions are always accepted. Parameter T is initially set to a relatively large value so that the transition from $\mathbf{x}$ to $\bar{\mathbf{x}}$ occurs more frequently, and then it is gradually decreased as the search proceeds. When T becomes sufficiently small and the solution does not change for many iterations, it is concluded to be "frozen", and the best solution available by then is outputted as the computed approximate solution.

The entire scheme is now summarized as follows.

Step 1: Determine an initial temperature T.

Step 2: Given an $\mathbf{x}$, pick randomly a solution $\bar{\mathbf{x}}$ in $N_e(\mathbf{x})$, and let Δ be the change in cost value of $\bar{\mathbf{x}}$ from that of $\mathbf{x}$.

Step 3: If $\Delta \leqslant 0$, then $\mathbf{x} := \bar{\mathbf{x}}$. Otherwise, let $\mathbf{x} := \bar{\mathbf{x}}$ with probability $e^{-\Delta/T}$.

Step 4: If it is concluded that a sufficient number of trials have been made with the current T (i.e., in *equilibrium*), then go to step 5. Otherwise return to step 2 with the current $\mathbf{x}$.

Step 5: If the current T is concluded to be sufficiently small (i.e., *frozen*), then go to step 6. Otherwise reduce the T (e.g., $T := rT$ with a constant r satisfying $0 < r < 1$) and return to step 2.

Step 6: Halt after outputting the best solution obtained so far as the computed approximate solution $\mathbf{x}$.

In order to apply simulated annealing to (CAP_1), we have to define the corresponding discrete solution space X, the cost function C and the neighborhood structure N_e.

3.1. Solution space

Suppose that we have P patterns generated by the pattern generation procedures in section 4. A P-dimensional vector $\mathbf{x}$ represents the current state of the channel allocation,

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_p \\ \vdots \\ x_P \end{bmatrix}.$$

A component x_p of vector $\mathbf{x}$ denotes the number of channels assigned to pattern p where $1 \leqslant p \leqslant P,\ 0 \leqslant x_p \leqslant M$, and $\sum_{p=1}^{P} x_p = M$.

3.2. *Cost function*

The channel allocation algorithm introduced here is aimed at minimizing the weighted average blocking probability. Therefore, the cost function is as follows:

$$C(x) = \sum_{i=1}^{N} w_i B(\lambda_i, m_i).$$

This is the weighted average blocking probability defined in section 2.

3.3. *Neighborhood structure*

Here, given an $\mathbf{x} = (x_1, x_2, \ldots, x_P)^{\mathrm{T}}$, the neighborhood $N_e(\mathbf{x})$ is defined as follows:

$$N_e(\mathbf{x}) = \left\{ \bar{\mathbf{x}} \mid \bar{\mathbf{x}} = (x_1, x_2, \ldots, x_p + 1, \ldots, x_q - 1, \ldots, x_P)^{\mathrm{T}} \right\},$$

where $p \neq q,\ 1 \leqslant p, q \leqslant P$, and $x_q - 1 \geqslant 0$. And we propose two strategies for picking a solution $\bar{\mathbf{x}}$ in $N_e(\mathbf{x})$.

- (Random picking): Select $\bar{\mathbf{x}}$ randomly in $N_e(\mathbf{x})$. That is, choose any two numbers p and q such that $p \neq q,\ 1 \leqslant p, q \leqslant P$, and $x_q \geqslant 1$. Set $\bar{\mathbf{x}} = \mathbf{x}$. Update $\bar{x}_p$ and $\bar{x}_q$ as follows:

$$\bar{x}_p = \bar{x}_p + 1, \qquad \bar{x}_q = \bar{x}_q - 1.$$

- (Proportional picking): Let NC_p be the number of cells covered with pattern p and let $TNC = \sum_{p=1}^{P} NC_p$. Then $(NC_p)/(TNC)$ is the proportion of the number of cells covered with pattern p to the total number of cells covered with all patterns. Since we can infer that the pattern p, which has the largest value of $(NC_p)/(TNC)$, may have more influence than other patterns, we determine the p and q as follows:

$$p = \mathrm{Argmax}\left\{ \frac{NC_h}{TNC},\ h = 1, 2, \ldots, P \right\},$$
$$q = \mathrm{Argmin}\left\{ \frac{NC_h}{TNC},\ x_h \geqslant 1,\ h = 1, 2, \ldots, P \right\}.$$

  Set $\bar{\mathbf{x}} = \mathbf{x}$. Update $\bar{x}_p$ and $\bar{x}_q$ as stated above.

4. Pattern generation

The problem (CAP_1) in section 2 is an approximation to the original problem (CAP). The choice of patterns considered in the problem (CAP_1) may have an influence on the quality of solutions relative to the problem (CAP). Thus it is important to find good candidate patterns. In finding good patterns, we consider the mutual distances between adjacent cochannel cells and the traffic demand distribution.

The mutual distance between adjacent cells, $D(g,k)$, is defined as the Euclidean distance between cell g and cell k. This is computed by

$$D(g,k) = \sqrt{(k_1 - g_1)^2 + (k_1 - g_1)(k_2 - g_2) + (k_2 - g_2)^2}\,,$$

where (g_1, g_2) and (k_1, k_2) are center coordinates of cells g and k, respectively, in the plane with inclined axes [4,11]. Figure 1 shows two cellular systems with regular and irregular cells. The formula for the mutual distance between adjacent cells can be applied to both systems. Examining figure 1(a), it is clear that $D(g,k)$ is $\sqrt{21}r_\mathrm{c}$, where r_c is the cell radius.

We first suggest two heuristic procedures to find patterns such that the mutual distances between adjacent cochannel cells are made as short as possible. The minimum reuse distance is denoted by δ.

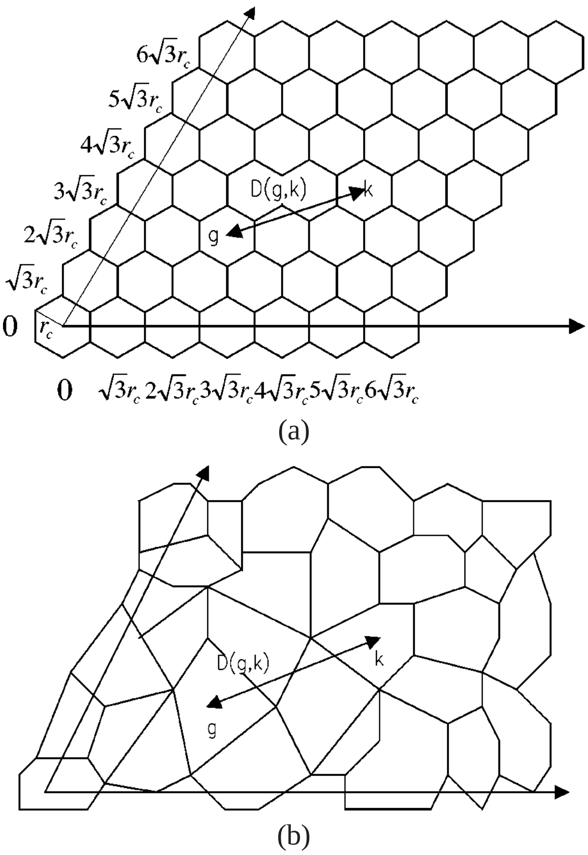

Figure 1. The mutual distance between adjacent cells: (a) a cellular system layout with regular cells; (b) a cellular system layout with irregular cells.

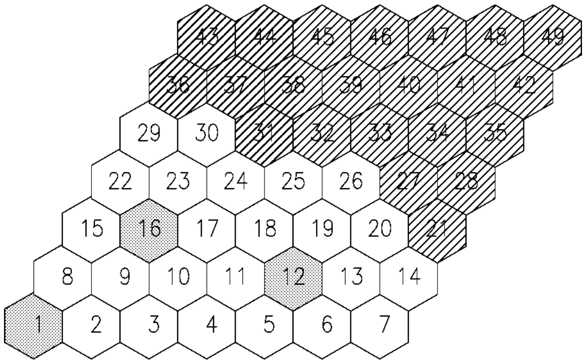

Figure 2. An example of pattern generation procedure A.

4.1. *Pattern generation procedure A*

Let $I_P(i)$ be the set of selected cochannel cells of cell i, and I_R be the set of cells that are not within the interference range of all cells in $I_P(i)$. This procedure selects a cell in I_R, which is nearest to each cell in $I_P(i)$.

Step 0: The set of all cells $I := \{1, \ldots, N\}$; $i := 1$.

Step 1: $I_P(i) := \{i\}$.

Step 2: $I_R = \{g \mid \min\{D(g,k);\ k \in I_P(i)\} \geqslant \delta;\ g \in I - I_P(i)\}$;
Set $g^* = \text{Argmin}\{\sum_{k \in I_P(i)} D(g,k);\ g \in I_R\}$.

Step 3: $I_P(i) := I_P(i) \cup \{g^*\}$; $I_R := I_R - \{g^*\}$; If $I_R = \emptyset$ then go to step 4; otherwise return to step 2.

Step 4: Check whether the pattern $I_P(i)$ is identical to one of patterns generated already or not; If so, discard the pattern $I_P(i)$; If $i = N$ then stop; otherwise $i = i + 1$, return to step 1.

An example of step 2 of the procedure A is depicted in figure 2. In this example, we can see $I_P(i) = \{1,\ 12,\ 16\}$, and $I_R = \{21, 27, 28, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49\}$. In order to determine g^*, we need to calculate $D(g,1) + D(g,12) + D(g,16)$ for each g in I_R. Therefore, $g^* = 31$.

4.2. *Pattern generation procedure B*

This procedure is a simple modification of the procedure A. In this procedure, a cell which is nearest to the first chosen cell is selected repeatedly. The procedure B is identical to the procedure A except step 2.

Step 2: $I_R = \{g \mid \min\{D(g,k);\ k \in I_P(i)\} \geqslant \delta;\ g \in I - I_P(i)\}$;
Set $g^* = \text{Argmin}\{D(g,i);\ g \in I_R\}$.

The procedures A and B do not consider the traffic demand distribution. It may be desirable to generate a pattern considering the traffic distribution of the system. We suggest a heuristic procedure C to find these patterns.

4.3. Pattern generation procedure C

Let I_D be the set of cells whose demand-difference is minimum. The procedure C is identical to the procedure A except step 2.

Step 2: $I_R = \{g \mid \min\{D(g,k);\ k \in I_P(i)\} \geqslant \delta;\ g \in I - I_P(i)\}$;
$I_D = \{g \mid \min \sum_{k \in I_P(i)} |\lambda_g - \lambda_k|;\ g \in I_R\}$;
Set $g^* = \text{Argmin}\{\sum_{k \in I_P(i)} D(g,k);\ g \in I_D\}$.

This procedure selects, among cells satisfying the cochannel interference constraints, a cell whose demand-difference from already selected cells is as small as possible. If a tie happens, select the closest cell from already selected cells. The procedures A, B, and C can be used separately or together. In the test of section 5, we will use all these procedures together.

5. Performance comparison

In order to test our algorithm SA we have considered a cellular system, which consists of $N = 49$ regular hexagonal cells as shown in figure 1(a). We assume that a total of $M = 70$ channels are available for the system. The arrival of calls is a Poisson process and the call duration is exponentially distributed with a mean of three minutes. We also assume that the minimum reuse distance δ is $\sqrt{21}r_c$. Thus, cells having a mutual distance not smaller than δ can use the same channels.

In order to apply simulated annealing, initial T and r is set to 10 and 0.65, respectively. We have used the simple approach of permitting 100 state transitions at each T level. When T is less than 10^{-12}, we conclude it is frozen. We have used the proportional picking strategy for picking a next solution. The number of patterns, P, generated using procedures in section 4 is 38. Even though the computation time is not critical because this (CAP) is a design problem, too much time-consuming methods are undesirable. In the case of the examples in this section, the computation time of our method was within a few minutes.

Let us start with a simple uniform traffic distribution of 100 calls/h in all 49 cells. By applying our method SA and the allocation strategies proposed in [13] to this simple problem, we found the set of channels allocated in each cell. Table 1 shows the weighted average blocking probability R and the total number of simultaneously usable channels z for each allocation strategy. Here, z is obtained by adding up the number of allocated channels in the 49 cells.

Next, consider a particular nonuniform traffic distribution shown in figure 3. The number in each cell represents the call arrival rates and ranges from 20 to 200 calls/h.

Table 1
Channel allocation with uniform traffic.

Strategy	R^{**} (%)	z^{***}
Uniform*	1.84	490
Compact*	1.84	490
Hybrid/CB*	1.97	490
SA	1.81	513

* The allocation strategies proposed in [13].
** The weighted average blocking probability.
*** The total number of simultaneously usable channels.

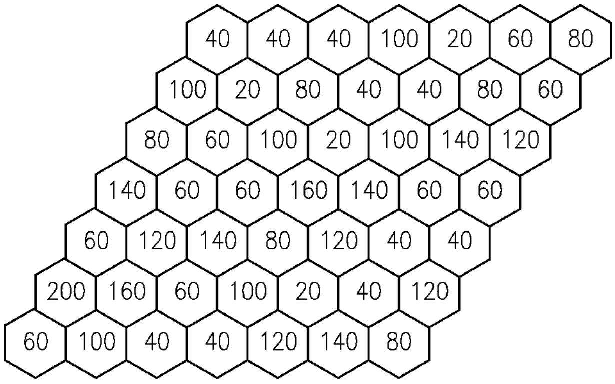

Figure 3. A nonuniform traffic distribution.

Table 2
Channel allocation with nonuniform traffic distribution.

Strategy	R^{**} (%)	z^{***}
Uniform*	4.45	490
Compact*	1.94	490
Hybrid/CB*	1.28	489
SA	1.20	516

* The allocation strategies proposed in [13].
** The weighted average blocking probability.
*** The total number of simultaneously usable channels.

This is a benchmark problem that has appeared in many papers [13,14]. The weighted average blocking probability and the total number of simultaneously usable channels for each allocation strategy are listed in table 2.

Among the allocation strategies considered in this paper, tables 1 and 2 show that SA gives the lowest overall blocking probability and the largest number of simultaneously usable channels in a 49 cell system with both uniform and nonuniform traffic rates. Especially, with regard to the total number of simultaneously usable channels,

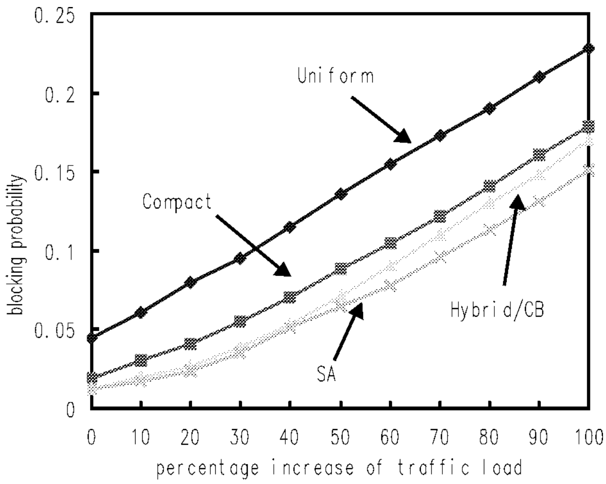

Figure 4. Comparison of channel allocation strategies.

SA shows very good performance. This result means that SA can significantly increase the traffic-carrying capacity.

Finally, figure 4 shows the average blocking probability of the allocation strategies considered in this paper as a function of traffic load. The base traffic load is shown in figure 3. This traffic load is then increased by 10%, 20%, ..., 100% over the base load. We find that the blocking probability using our method is always lower than that using other methods considered in [13]. In addition, figure 4 reveals that the relative quality of our solutions is better as the traffic load increases.

6. Conclusions

An optimal channel allocation problem, which minimizes the weighted average blocking probability subject to cochannel interference constraints in a cellular mobile system, has been suggested. The problem has been converted into a simpler form through the concept of pattern. We have applied a simulated annealing procedure to the simplified problem. It is found that our method provides the best performance among the strategies considered in this paper. The proposed channel allocation method can be used for TDMA systems with small modifications.

References

[1] T. Abe, K. Yoji, M. Sengoku, S. Tamura and S. Shinoda, Graph theoretical considerations of channel offsets systems in a cellular mobile system, IEEE Transactions on Vehicular Technology 40 (1991) 405–411.

[2] F. Box, A heuristic technique for assigning frequencies to mobile radio nets, IEEE Transactions on Vehicular Technology 27 (1978) 57–64.

[3] M. Duque-Antón, D. Kunz and B. Rüber, Channel assignment for cellular radio using simulated annealing, IEEE Transactions on Vehicular Technology 42 (1993) 14–21.

[4] A. Gamst, Homogeneous distribution of frequencies in a regular hexagonal cell system, IEEE Transactions on Vehicular Technology 31 (1982) 132–144.

[5] A.I. Giortzis and L.F. Turner, A mathematical programming approach to the channel assignment problem in radio networks, in: *IEEE Vehicular Technology Conference* (1996) pp. 736–740.

[6] D. Hong and S.S. Rappaport, Heuristic channel assignments for cellular land mobile radio systems, in: *GLOBECOM* (1985) pp. 997–1001.

[7] S. Kim and S.-L. Kim, A two-phase algorithm for frequency assignment in cellular mobile systems, IEEE Transactions on Vehicular Technology 43 (1994) 542–548.

[8] D. Kunz, Channel assignment for cellular network using neural networks, IEEE Transactions on Vehicular Technology 40 (1991) 188–193.

[9] D. Kunz, Suboptimum solutions obtained by the Hopfield–Tank neural network algorithm, Biol. Cybernetics 65 (1991) 129–133.

[10] W.C.Y. Lee, *Mobile Cellular Telecommunications Systems* (McGraw-Hill, New York, 1989).

[11] V.H. MacDonald, Advanced mobile phone service: The cellular concept, Bell System Technical Journal 58 (1979) 15–41.

[12] N. Metropolis, A. Rosenbluth, A. Teller and E. Teller, Equation of state calculations by fast computing machines, The Journal of Chemical Physics 21 (1953) 1087–1092.

[13] K.L. Yeung and T.S. Yum, The optimization of nominal channel allocation in cellular mobile systems, in: *IEEE International Conference on Communications* (1993) pp. 915–919.

[14] M. Zhang and T.S. Yum, The nonuniform compact pattern allocation algorithm for cellular mobile systems, IEEE Transactions on Vehicular Technology 40 (1991) 387–391.