Channel Assignment in Cellular Radio Networks

Rudolf Mathar and Jürgen Mattfeldt

Abstract— **We investigate algorithms based on simulated annealing to solve the channel assignment problem for cellular radio networks. The blocking probability of a network is chosen as the optimization criterion. In order to check the quality of the solutions obtained by simulated annealing, we examine some special types of networks which allow an effective calculation of optimal solutions by tailored algorithms. Our investigations show that simulated annealing is a very powerful tool for solving channel assignment problems.**

I. Introduction

CHANNEL assignment is one of the most important problems in the design of cellular radio networks. Since the number of cells of forthcoming networks will rapidly increase, this problem will be of even greater importance in the future [6]. From a mathematical point of view, the channel assignment problem (CAP) can be described as a combinatorial optimization problem with constraints. Many different benefit functions and constraints are worth considering, and consequently a broad spectrum of models can be found in the literature. Strictly speaking, there is not a channel assignment problem, but a whole class of channel assignment problems. Hale [12] gives a classification of some important channel assignment problems.

Most previous investigations concerning the CAP were based on graph theoretic or heuristic approaches [2], [3], [7]–[9]. Graph theoretic methods are near at hand since the channel assignment problem can be considered as a generalization of the vertex coloring problem. By an appropriate reduction, this shows that CAP is NP-hard [22]. Moreover, our Lemma 1 shows that there does not exist an efficient algorithm which determines a solution half as good as the optimal one in polynomial time, provided $P \neq NP$.

The graph theoretic approach has several disadvantages in applicability and flexibility (cf. [4], [16] for a detailed discussion). This is why recently some other methods have been examined. Investigations based on neural networks [16] yield excellent results in special cases, but under certain conditions, principally only suboptimal solutions can be found [17].

In this paper, we examine a quite satisfactory approach based on simulated annealing (SA). This method has been applied independently by Duque–Anton *et al.* [4], [5], but they use a completely different model and different neighborhood relations. We briefly mention the main differences in Section IV. Our model to tackle the channel assignment problem is presented in detail in Section II.

Using simulated annealing to solve an NP-hard problem is linked to the question of how to control the quality of the obtained solutions. In general, it is impossible to solve this task in an effective manner. In Section III we develop polynomial time algorithms for certain types of networks. Readers more interested in the application of simulated annealing as a general tool for optimizing cellular networks may skip the proofs of optimality of the corresponding algorithms and continue with Section IV. This section is self-contained and provides a complete description of the SA-algorithm. In Section V we compare the solutions obtained by simulated annealing with the optimal ones in special cases. This comparison proves SA is a powerful tool in optimizing cellular radio networks.

Manuscript received March 19, 1993; revised May 11, 1993. This work was supported in part by the Deutsche Forschungsgemeinschaft under Grant Ma 1184/2-3.

The authors are with the Aachen University of Technology, Stochastik, insbesondere Anwendungen in der Informatik, RWTH Aachen, Wullnerstrasse 3, D-5100 Aachen, Germany.

IEEE Log Number 9211593.

II. Formulation of the Channel Assignment Problem

We first introduce some standard definitions from graph theory. A simple graph G is an ordered pair (V, E) with a finite set of vertices V and edges $E \subset V^{(2)} = \{e \subset V || e| = 2\}$. A set $W \subset V$ is called independent if $W^{(2)} \cap E = \emptyset$. $\mathcal{C} = \{W_1, \cdots, W_n\}$ is called a covering of G, if all sets $W_i \subset V$ are independent, and $\bigcup_{i=1}^{n} W_i = V$ holds. If additionally $W_i \cap W_j = \emptyset$ for all $i \neq j$, then $\mathcal{C}$ is called a coloring of G. $\aleph(G) = \min\{|\mathcal{F}| | \mathcal{F}$ is a coloring of $G\}$ is defined as the chromatic number of G. A graph G with chromatic number k is called k -colorable.

In what follows, we consider a cellular network $\mathcal{Z}$ consisting of z cells $Z_1, \cdots, Z_z$ (cf. [18], [19]). Our aim is to allocate N channels to the individual cells of the network such that certain constraints are satisfied. The constraints are described by an interference graph $G_{\mathcal{Z}} = (V, E)$ with

$$V = \{Z_1, \cdots, Z_z\}, \quad \text{and}$$
$$E = \{\{Z_i, Z_j\} | \text{interference between } Z_i \text{and } Z_j \text{is possible}\}.$$

For the applications, this graph can be constructed either from measurements or from theoretical models of the propagation of electromagnetic waves. Let $A = (a_{ij})_{i,j=1}^{z,z}$ be the adjacency matrix of $G_{\mathcal{Z}}$, i.e., $a_{ij} = 1$ iff $\{Z_i, Z_j\} \in E$, and $a_{ij} = 0$ otherwise.

Definition 1: A matrix $M_N = (m_{ij})_{i,j=1}^{z,N} \in \{0,1\}^{z \times N}$ is called an N-channel design, if each column of M_N induces an independent set in $G_{\mathcal{Z}}$, i.e. the sets $W_j = \{Z_i | m_{ij} = 1, i = 1, \cdots, z\}$ are independent in $G_{\mathcal{Z}}$ for all $j = 1, \cdots, N$.

$m_{ij} = 1$ means that the jth channel is allocated to cell Z_i, while $m_{ij} = 0$ excludes channel j for cell Z_i. The number a_i

of channels available for cell Z_i using pattern M_N is given by

$$a_i = \sum_{j=1}^{N} m_{ij}, \qquad i = 1, \cdots, z. \tag{2.1}$$

Observe that we consider only co-channel interferences. Nevertheless, our algorithms can be extended to the more general case of adjacent channel interferences.

Definition 2: A channel design M_N is called admissible if there exists no channel design $M'_N \neq M_N$ with $m'_{ij} \geq m_{ij}$ for all i and j. Otherwise it is called inadmissible.

To rank different channel designs with respect to their performance we assume that an ordered pair (f_i, w_i) is assigned to each cell Z_i. f_i is a probability measure on $\mathbb{N}$ and the numbers $w_i \geq 0$ with $\Sigma_{i=1}^{z} w_i = 1$ are weights, expressing e.g. the relative importance of cell i. The blocking probability $B_{Z_i}(a_i)$ of cell Z_i with a_i allocated channels is given by

$$B_{Z_i}(a_i) = 1 - \sum_{k=1}^{a_i} f_i(k). \tag{2.2}$$

Example 1: Let λ_i (in Erlangs) be the utilization of cell Z_i. Assume that demands in different cells Z_i are created according to independent Poisson processes with arrival rate μ_i, and that service times are independent and exponentially distributed with rate $\nu_i, i = 1, \cdots, z$. This yields $\lambda_i = \mu_i/\nu_i$. In the stationary case one can calculate the blocking probability $B_{Z_i}(a_i)$ of cell Z_i by the well known Erlang-B formula [25]. In our notation (2.2) this leads to

$$f_i(k) = b_i(k-1) - b_i(k), k \in \mathbb{N},$$

where

$$b_i(k) = \frac{\lambda_i^k/k!}{\sum_{\ell=0}^{k} \lambda_i^\ell/\ell!}. \tag{2.3}$$

It is quite natural to choose weights proportional to the traffic utilization in individual cells, as in [27]

$$w_i = \frac{\lambda_i}{\sum_{\ell=1}^{z} \lambda_\ell}, \qquad i = 1, \cdots, z. \tag{2.4}$$

In what follows we will call a sequence $(E_n(\lambda))_{n \in \mathbb{N}_0}$ an Erlang-B sequence if

$$E_n(\lambda) = \frac{\lambda^n/n!}{\sum_{j=0}^{n} \lambda^j/j!}, n \in \mathbb{N}_0, \lambda > 0. \tag{2.5}$$

We will measure the performance of a channel design $M_N = (m_{ij})_{i,j=1}^{z,N}$ by the weighted sum of individual blocking probabilities in each cell. Let $a_i = \Sigma_{j=1}^{N} m_{ij}, i = 1, \cdots, z$ be the number of channels in cell Z_i. The function

$$\Im(M_N) = \sum_{i=1}^{z} w_i B_{Z_i}(a_i) = 1 - \sum_{i=1}^{z} w_i \sum_{j=1}^{a_i} f_i(j). \tag{2.6}$$

is called the benefit function. To find an optimal plan with maximum benefit we obviously may restrict our attention to admissible designs.

The importance of (2.6) originates from its interpretation in an open network of servers with customers in FCFS discipline and equal priority. This has influenced many authors [14], [23], [26], [27], to use criteron (2.6). Of course, other benefit functions could be meaningful as well, depending on the particular goal under consideration. We already here emphasize that the performance of simulated annealing (cf. Section IV) does not strongly depend on the particular form of the benefit function.

We are now prepared to formulate the channel assignment problem (CAP) in our model.

CAP$(N), N \in \mathbb{N}$:

Instance: An interference graph G_Z and rational cell parameters $(f_i, w_i)_{i=1}^{z}$.

Problem: Determine an N-channel design M_N^* with $\Im(M_N^*) \leq \Im(M_N)$ for all N-channel designs M_N.

In the HCA case [26], the set of all channels is divided into two disjoint groups G_1 and G_2, where channels in the first group have their assignment fixed, and in the second group channels are assigned dynamically. In this model, it is necessary to allocate the channels in G_1 optimally to achieve good performance. So for this strategy CAP$(|G_1|)$ arises as a subproblem.

CAP(N) is NP-hard for every fixed N because it can easily be reduced to INDEPENDENT SET [10]. Presumably there is even no efficient algorithm to find a approximately optimal solution of CAP, as may be seen by the following Lemma.

Lemma 1: Provided $P \neq NP$ holds, for any $\alpha > 1$ there is no polynomially time bounded algorithm which for every CAP (N) determines a solution M_N satisfying

$$\Im(M_N) \leq \alpha \Im(M_N^*) \tag{2.7}$$

where M_N^* is an optimal solution.

The proof follows easily from [13, ex. 13.19] (cf. [7]).

III. Networks of Special Structure

Though CAP(N) is NP-hard in general, for certain types of networks there are effective algorithms to determine optimal channel allocations. In this chapter, we first investigate designs which are related to a fixed coloring of the interference graph. This is important if one is interested, *a priori*, only in designs with certain symmetry properties. Furthermore, we deal with star systems and linear networks. In all cases efficient algorithms are derived. Of course, these results are of interest by themselves. Moreover, in special cases they allow us to compare the best results obtained by the algorithms in Section IV with the corresponding optimal solutions.

To give a formal description, we first introduce a fixed covering $\mathcal{C} = \{W_1, \cdots, W_m\}$ of the interference graph $G_{\mathcal{Z}}$.

Definition 3: A N-channel design $M_{N,\mathcal{C}}$ is called $\mathcal{C}$-design if each column of $M_{N,\mathcal{C}}$ induces a set in $\mathcal{C}$, i.e., $U_j = \{Z_i | m_{ij} = 1, i = 1, \cdots, z\} \in \mathcal{C}$ holds for all $j = 1, \cdots, N$.

Using combinations with repetitions (see e.g. [15]), one can see that there are $\binom{N+m-1}{m-1}$ nonisomorphic designs. We call two $\mathcal{C}$-designs isomorphic if they coincide after renumbering the channels. So the number of $\mathcal{C}$-designs increases rapidly with N. For example, $N = 100$ and $m = 19$ yields about $9.93 \cdot 10^{20}$ $\mathcal{C}$-designs. These values are typical for hexagonal structures in practice (cf. [19], [27]).

We now deal with the problem of finding an optimal $\mathcal{C}$ - design $M^*_{N,\mathcal{C}}$ with $\Im(M^*_{N,\mathcal{C}}) \leq \Im(M_{N,\mathcal{C}})$ over all $\mathcal{C}$-designs $M_{N,\mathcal{C}}$.

In the special case of hexagonal structures and regular coverings this problem has been considered by Zhang and Yum [27]. The following " **G**eneralized **Z**hang **Y**um algorithm" aims at determining an optimal $\mathcal{C}$-design by a greedy type strategy.

Algorithm GZY:

```
begin
  Initialize m_ij := 0; a_i := 1 for all 1 ≤ i ≤ z, 1 ≤ j ≤ N;
  for j := 1 to N do
  begin
    Choose W* ∈ {W_1, ..., W_m} with
    Σ_{Z_k ∈ W*} w_k f_k(a_k) maximal;
    for all Z_k ∈ W* do begin a_k := a_k + 1;
                              m_kj := 1      end;
  end;
  a_i := a_i − 1 for all 1 ≤ i ≤ z;
end.
```

GZY allocates channels successively to those groups W_ℓ of cells which yield the largest reduction in blocking probability. Now the question arises in which cases this algorithm finds an optimal $\mathcal{C}$-design.

Theorem 1: Let $\mathcal{C}$ be a coloring of $G_{\mathcal{Z}}$ and f_i nonincreasing for all $i = 1, \cdots, z$. Then the GZY-algorithm determines an optimal $\mathcal{C}$ -design.

Proof: For any $\mathcal{C}$-design we have $a_j = a_{j'} = a_{W_i}$, say, whenever cells j and j' belong to the same group W_i. Let $F_{W_i}(j) = \Sigma_{\ell \in W_i} w_\ell f_\ell(j)$. Since $\mathcal{C}$ is a coloring, to find an optimal design means to

$$\text{maximize} \sum_{i=1}^{m} \sum_{j=1}^{a_i} F_{W_i}(j)$$

such that

$$\sum_{i=1}^{m} a_{W_i} = N.$$

Because all $f_j, j = 1, \cdots, z$, are decreasing, the functions $F_{W_i}, i = 1, \cdots m$, are decreasing, too. This is the reason why the above greedy algorithm always finds an optimal $\mathcal{C}$-design. ■

In case of Erlangian blocking probabilities, we immediately get the following result.

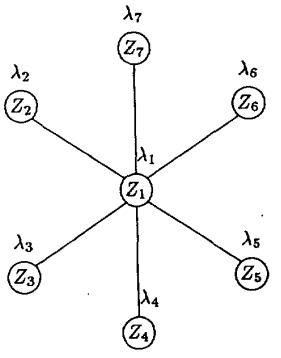


Fig. 1. Interference graph of a star system.

Corollary 1: Let $\mathcal{C}$ be a coloring of $G_{\mathcal{Z}}$, $(f_i(n))_{n \in \mathbb{N}} = (E_{n-1}(\lambda_i) - E_n(\lambda_i))_{n \in \mathbb{N}}$, and $\lambda_i > 0, i = 1, \cdots, z$. Then the GZY-algorithm finds an optimal $\mathcal{C}$-design.

Proof: For any $\lambda > 0$ the sequence $(E_n(\lambda))_{n \in \mathbb{N}}$ is decreasing and convex [21]. Thus the first order differences $(f_i(n))_{n \in \mathbb{N}}$ of (2.3) are decreasing, which by Theorem 1 establishes optimality. ■

Remark 1: The covering considered in [27] is not a coloring. Counterexamples illustrating that the algorithm in [27] fails to find an optimal solution can be easily constructed. Roughly speaking, this is due to the absence of a matroid structure [24] in that case.

Next we examine the CAP for interference graphs of special structure. Robinson [23] proves optimality results for certain types of 2-colorable graphs. A special class of such graphs is given by so called star systems, i.e. networks of the type depicted in Fig. 1.

As in Example 1, for this network class we assume Erlangian blocking probabilities and weights proportional to the utilization. As shown in [23], if

$$\sum_{i=2}^{z} \frac{\nu_i^N / N!}{\sum_{j=0}^{N} \nu_i^j / j!} \geq 1$$

an optimal channel design is given by the strategy: allocate all channels to the surrounding cells and none to the center. This still holds true even if dynamic channel assignment is applied. Mixed strategies (assign n channels to the center and $N - n$ channels to the surrounding cells, $n > 0$) are never optimal with dynamic channel assignment, whatever the rates ν_i are (though for arbitrary ν_i in the case of fixed channel assignment, mixed strategies could be optimal). There is an effective way of calculating optimal design for star systems.

Lemma 2: For star systems an optimal N-channel design can be determined in $O(\log_2 N)$ steps.

Proof: An optimal design is determined by the solution of $\min_{1 \leq n \leq N} h(n)$, where $h(n) = (\Sigma_{i=1}^{z} \lambda_i)^{-1}(f(n) + g(n))$, and $f(n) = (\lambda_1^{n+1}/n!)/(\Sigma_{j=0}^{n} \lambda_1^j / j!), g(n) = \Sigma_{i=2}^{z} (\lambda_i^{N-n+1}/(N-n)!)/(\Sigma_{j=0}^{N-n} \lambda_i^j / j!), n \in \mathbb{N}$. h is convex since f and g have this property. So the minimum can be found by binary search (successively halving intervals). Start with calculating $h(\lfloor N/2 \rfloor)$. If $h(\lfloor N/2 \rfloor - 1)$ is smaller than

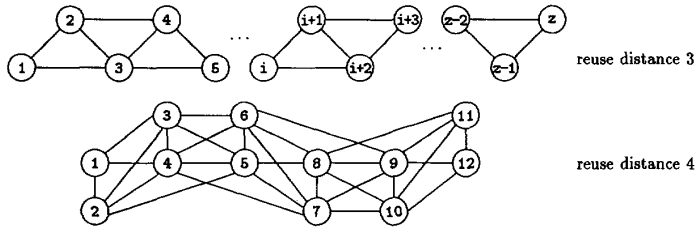


Fig. 2. Interference graph of linear networks ($k = 2, 3$).

the previous value, calculate $h(\lfloor N/4 \rfloor)$, otherwise $h(\lfloor 3N/4 \rfloor)$, etc. The minimum is reached after $O(\log_2 N)$ steps. ■

The next type of networks where we can give an effective algorithm to calculate optimal designs are linear networks with reuse distance $k+1$. Here interference is possible between cells Z_i and $Z_{i'}$ when $|i - i'| \leq k$ holds. Examples with reuse distance 3 and 4 are shown in Fig. 2.

Theorem 2: For linear networks with reuse distance $k+1$ CAP(1) can be solved in $O(z^{2+\log_2(2k+1)})$ steps.

Proof: Let $P = (p_1, \cdots, p_z)', p_i \in \{0, 1\}, 1 \leq i \leq z$, be a 1-channel design. By $P_{i,j}, 1 \leq i \leq j \leq z$, we denote the subdesign $(p_i, \cdots, p_j)'$ of $P = P_{1,z}$. If $i > j$ let $P_{i,j}$ denote the empty design. For $i \leq k \leq j$ write $P_{i,j} = P_{i,k}P_{k+1,j}$ to concatenate $P_{i,k}$ and $P_{k+1,j}$. Let $P^* = (p_1^*, \cdots, p_z^*)'$ be an optimal design which w.l.o.g. may be assumed admissible. Then there exists a number $\ell \in \{-k, -k+1, \cdots, k-1, k\}$ and a decomposition $P^* = P^1\,P^2\,P^3$ with the following three properties.

$$P^1 = P^*_{1,\lfloor (z+k)/2 \rfloor + \ell - k} \text{is an optimal 1-channel design w.r.t. the subnet } \{Z_1, \cdots, Z_{\lfloor (z+k)/2 \rfloor + \ell - k}\}. \quad (3.1)$$

$$P^2 = P^*_{\lfloor (z+k)/2 \rfloor + \ell - k + 1, \lfloor (z+k)/2 \rfloor + \ell} = (0, \cdots, 0)'. \quad (3.2)$$

$$P^3 = P^*_{\lfloor (z+k)/2 \rfloor + \ell + 1, z} \text{is an optimal 1-channel design w.r.t. the subnet } \{Z_{\lfloor (z+k)/2 \rfloor + \ell + 1}, \cdots, Z_z\} \text{ with } p^*_{\lfloor (z+k)/2 \rfloor + \ell + 1} = 1. \quad (3.3)$$

Such a decomposition exists since an admissible optimal design has at most $2k$ zeros between two consecutive ones. On the other hand, all consecutive ones are separated by at least k zeros. Therefore, at least one of $p^*_{\lfloor (z+k)/2 \rfloor - k + 1}, \cdots, p^*_{\lfloor (z+k)/2 \rfloor + k + 1}$ is equal to 1, say $p^*_{\lfloor (z+k)/2 \rfloor + \ell + 1} = 1$. The three subdesigns induced by ℓ must have the above properties since otherwise P^* would not be optimal.

We assume that calculation time is proportional to the number of steps. Let $t_k(z)$ be the time in the worst case necessary to determine an optimal 1-channel design for a linear network with reuse-distance $k+1$, where $t_k(z) = 0$ if $z < 0$. Recursive application of the above decomposition scheme yields

$$t_k(z) = \sum_{\ell=-k}^{k} \left(t_k\left(\left\lfloor \frac{z+k}{2} \right\rfloor + \ell - k \right) + t_k\left(z - \left\lfloor \frac{z+k}{2} \right\rfloor - \ell \right) \right)$$

$$\leq 2(2k+1) t_k\left(\left\lfloor \frac{z+k}{2} \right\rfloor + 1 \right)$$

$$\leq 2^2 (2k+1)^2 t_k\left(\left\lfloor \frac{\lfloor (z+k)/2 \rfloor + 1 + k}{2} \right\rfloor + 1 \right)$$

$$\leq 2^2 (2k+1)^2 t_k\left(\left\lfloor \frac{z}{4} + \frac{3k}{4} + \frac{3}{2} \right\rfloor \right)$$

$$\vdots$$

$$\leq 2^{\lfloor \log_2(z) \rfloor} (2k+1)^{\lfloor \log_2(z) \rfloor} \cdot t_k\left(\left\lfloor \frac{z}{2^{\lfloor \log_2(z) \rfloor}} + \frac{(2^{\lfloor \log_2(z) \rfloor} - 1)k}{2^{\lfloor \log_2(z) \rfloor}} + \frac{2^{\lfloor \log_2(z) \rfloor} - 1}{2^{\lfloor \log_2(z) - 1 \rfloor}} \right\rfloor \right)$$

$$\leq z^2 z^{\log_2(2k+1)} t_k(k+4)$$

$$= O(z^{2+\log_2(2k+1)}).$$

■

From the proof of Theorem 2, it is clear how to construct a corresponding algorithm. We do not give a full implementation but only the basic description. For $\ell = -k, \cdots, k$ iterate decomposing P: array$[1 \cdots z]$ into $P_1^{(\ell)}, P_2^{(\ell)}$, and $P_3^{(\ell)}$ where the first index of $P_2^{(\ell)}$ is $\lfloor (z+k)/2 \rfloor + \ell - k + 1$ and $P_2^{(\ell)}$ has length k. Set all entries of $P_2^{(\ell)} = 0$. Determine an optimal design for the right subarray $P_3^{(\ell)}$. If the first element of $P_3^{(\ell)}$ is 1, then determine an optimal design for $P_1^{(\ell)}$, otherwise let $\ell := \ell + 1$. To determine optimal channel designs in the left and right subarrays, apply the above described procedure recursively until the remaining arrays are so small (e.g., $\leq 2k$ elements) that optimal designs can be determined immediately. The above estimation considers worst cases, we do not take into account cutting the left subtree if a 0 is achieved as the first element of the right subarray.

Remark 2: The number of admissible 1-channel designs for linear networks can be described by generalized Fibonacci numbers which grow exponentially. Furthermore, the following decision problem is NP-complete: Given $b \in \mathbb{Q}$, decide whether there exists a 1-channel design M_1 with $\Im(M_1) = b$.

Remark 3: If $N > 1$ a successive application of the above algorithm does not necessarily yield an optimal N-channel design. But for monotone benefit functions f_i (as e.g. for the Erlang-B benefit) this approach seems to give nearly optimal results for a variety of parameters (cf. Theorem 1 and Section V).

Combining the results obtained so far, one can also treat more complex networks as is illustrated in Fig. 3. We have four subnets of stations 1–5 (linear, reuse distance 2), 6–13 (linear, reuse distance 3), 14–16 (star system), and 17–20 (star-

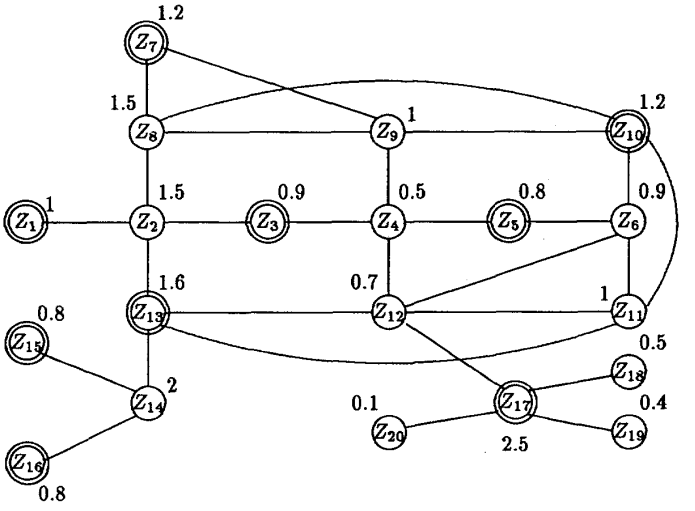


Fig. 3. Optimal 1—channel design of a combined network.

system); the utilization is denoted beside each cell in Erlangs. For each subnet, optimal channel allocations are determined by the above methods. Circled stations use the available channel. Since there is no interference between assigned stations of the individual optimal subdesigns, the depicted channel assignment is optimal for the whole network (cf. [23]). Using this method, it is possible to construct arbitrarily large networks with known optimal channel assignment. This will be important for our analysis of the solutions obtained by simulated annealing.

IV. Solution of the CAP by Simulated Annealing

The channel assignment problem is an example of a combinatorial optimization problem. An instance of such a problem is given by an ordered pair (S, f), where $S = \{1, \cdots, r\}$ is a finite set and $f : S \to \mathbb{R}$ is a cost function defined on the state space S. The aim is to find a solution $i_{\text{opt}} \in S$ with $f(i_{\text{opt}}) \leq f(i)$ for all $i \in S$. Let $S_{\text{opt}} = \{i_{\text{opt}} \in S | f(i_{\text{opt}}) \leq f(i) \text{ for all } i \in S\}$ denote the set of optimal solutions. In the present case the state space S is given by the set of (admissible) N-channel designs and $\Im$ serves as the cost function.

Simulated annealing (SA) is a general approach for solving combinatorial optimization problems. The corresponding algorithm is well described in [1], and there are several strong convergence results, see [1], [11], [20]. The basic ingredients are a subroutine commonly called `generate`(j from i), and an appropriate cooling schedule for the control parameter. Acceptance probabilities are chosen in a standard way, resembling the exponential form of Metropolis' approach.

The procedure `generate`(j from i) creates randomly with probability G_{ij} a new state j from the current one $i, 1 \leq i, j \leq r$. Let the random variable X_k describe the state achieved after the kth iteration of the algorithm. Convergence in distribution (i.e. $\lim_{k\to\infty} P(X_k \in S_{\text{opt}}) = 1$) can be guaranteed under general circumstances [1] provided the generation probabilities satisfy the following conditions.

$$G_{ij} = G_{ji} \text{ for all } i, j \in S, \tag{4.1}$$

and for all $i, j \in S$ there exist $p \geq 1$ and $\ell_0, \ell_1, \cdots, \ell_p \in S$ with $\ell_0 = i, \ell_p = j$ such that

$$G_{\ell_k \ell_{k+1}} > 0 \text{ for all } k = 0, \cdots, p-1. \tag{4.2}$$

We will formulate versions of the SA-algorithm which satisfy these conditions and thus guarantee convergence. Instead of choosing S as the set of admissible N -channel designs in the sense of Definition 1, we consider the set $S_{\text{Perm}} = \{\Pi = (\Pi_1, \cdots, \Pi_N) | \Pi_i \in \text{Perm}_z, i = 1, \cdots, N\}$, where Perm_z denotes the set of permutations of $\{1, \cdots, z\}$. The following procedure establishes a correspondence between S_{Perm} and S. For $\Pi \in S_{\text{Perm}}$ let M_Π define the unique admissible channel design given by the algorithm below.

Algorithm ($\Pi \to M_\Pi$):

```
for i := 1 to N do
  for j := 1 to z do
  begin
    if {Z_{Π_i(j)}} ∪ {Z_{Π_i(ℓ)} | m_{Π_i(ℓ),i} = 1, 1 ≤ ℓ < j}
        is an independent set in G_Z
        then m_{Π_i(j),i} = 1 else m_{Π_i(j),i} = 0;
  end;
```

Fig. 4 contains a small example illustrating how this algorithm works.

Obviously for a given admissible N-channel design M_N, there is at least one $\Pi \in S_{\text{Perm}}$ with $M_\Pi = M_N$. The cost function f on S_{Perm} is defined by $f(\Pi) = \Im(M_\Pi)$ for $\Pi \in S_{\text{Perm}}$.

Now let $\Psi = \{\Phi_1, \cdots, \Phi_t\} \subseteq \text{Perm}_z$. By the following procedure we generate a new channel design from the actual one $\Pi = (\Pi_1, \cdots, \Pi_N)$.

`generate_one` (Π' from Π):

1. Randomly choose a channel $r \in \{1, \cdots, N\}$ according to a uniform distribution.

$z = 5,\ N = 2,$

$\Pi = (\Pi_1, \Pi_2)$

$\Pi_1 = (12354)$

$\Pi_2 = (23)$

$$M_\Pi = \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}$$

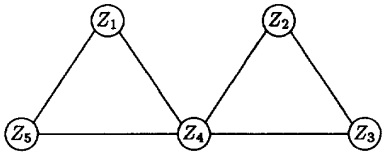


Fig. 4. Illustration of the $\Pi \rightarrow M_\Pi$ correspondence.

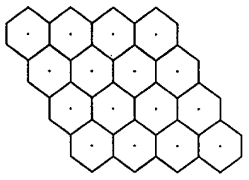


Fig. 5. Hexagonal structure.

2. Independently choose $i \in \{1, \cdots, t\}$ according to a uniform distribution and select a permutation $\Phi_i \in \Psi$.
3. For $\Pi = (\Pi_1, \cdots, \Pi_r, \cdots, \Pi_N)$, define $\Pi' = (\Pi_1, \cdots, \Phi_i \Pi_r, \cdots, \Pi_N)$.

If a generator contains along with each permutation its inverse, the assumptions for theoretical convergence are fulfilled, as the following Lemma shows.

Lemma 3: Let $\Psi \in Perm_z$ be a generator of $Perm_z$ and suppose that

$$\Phi \in \Psi \Rightarrow \Phi^{-1} \in \Psi \tag{4.3}$$

holds. Then the probabilities G_{ij} corresponding to `generate_one` *satisfy (4.1) and (4.2).*

Proof: (4.1) follows immediately from (4.3) and the distributional symmetry of the procedure `generate_one`. To prove (4.2) it suffices to consider the case $z = 1$. Let $\Pi_1, \Pi_2, \in \mathrm{Perm}_z$. Since Ψ is a generator of Perm_z there exist permutations $\Phi_{k_1}, \cdots, \Phi_{k_\ell} \in \Psi$ such that $\Phi_{k_\ell} \cdots \Phi_{k_1} = \Pi_2 \Pi_1^{-1}$. This implies $\Phi_{k_\ell} \cdots \Phi_{k_1} \Pi_1 = \Pi_2$, and, therefore, each permutation $\Pi_2 \in \mathrm{Perm}_z$ is reachable with positive probability in a finite number of steps from every other $\Pi_1 \in \mathrm{Perm}_z$, which is (4.2). ■

The following generators of Perm_z are examples which fulfill condition (4.3):

$$\Psi_1 = \{(i(i+1)) | 1 \le i < z\},$$
$$\Psi_2 = \{(i(i+1)\cdots(j-1)j) | 1 \le i < j \le z\} \cup \{(j(j-1)\cdots(i+1)i) | 1 \le i < j \le z\},$$
$$\Psi_3 = \{(i,j)((i+1)(j-1))\cdots((i+k)(j-k)) \cdot | 1 \le i < j \le z, k = \lfloor (j-i-1)/2 \rfloor\}.$$

Using Lemma 3 and applying one of the generators Ψ_1, Ψ_2, or Ψ_3, with an appropriate cooling schedule `generate_one` yields a convergent sequence of channel designs. A drawback is that updating is relatively expensive. Examples of generate procedures which are less expensive (but for which we cannot prove theoretical convergence) are obtained as follows. We return to the original state space of N-channel designs.

`generate_two` (M' from M):

1. Choose a channel $r \in \{1, \cdots, N\}$ at random according to a uniform distribution.
2. Choose a permutation $\Phi_i \in \Psi_3, 1 \le i \le \binom{z}{2}$, at random according to a uniform distribution.
3. Let Fix_{Φ_i} denote the set of fixed points of Φ_i. M' is updated from M by modifying the rth column according to the following algorithm.

Initialize $m'_{jr} = \begin{cases} m_{jr}, & \text{if } j \in \mathrm{Fix}_{\Phi_i}, \\ 0, & \text{if } j \notin \mathrm{Fix}_{\Phi_i}. \end{cases}$

```
for j = 1 to z do
  if j ∉ Fix_{Φ_ij} then
    if ({Z_{Φ(ij)}} and {Z_{Φ(iℓ)} | m'_{ℓr} = 1, 1 ≤ ℓ ≤ z}
        are independent)
      then m'_{jr} = 1 else m'_{jr} = 0;
```

Compared with `generate_one`, in this procedure only local updating is necessary. The next generate procedure shows how updating may be carried out very fast. The decrease in performance will be investigated in chapter 5 (see Example 2).

`generate_three` (M' from M):

1. Choose at random a channel $r \in \{1, \cdots, N\}$ according to a uniform distribution.
2. Choose $i \in \{1, \cdots, z\}$ at random according to a uniform distribution and select cell Z_i.
3. Assign channel r to cell Z_i and remove channel r from all cells adjacent to cell Z_i.

Note that the last two generate procedures do not in general satisfy the symmetry condition (4.1), and thus the convergence theory [1] cannot be applied. Nevertheless, we have observed good results also with the relaxed updates from `generate_two` and `generate_three`.

The main differences between our model and [4], [5] show up in the treatment of constraints and the benefit function. In [4], [5] constraints concerning the interference graph are embedded in the benefit function $\mathfrak{H}$ by adding a penalty term. This makes optimization of $\mathfrak{H}$ complicated, and for many updates, and even possibly for obtained solutions, interference restrictions are violated. The aim in [4], [5] is to minimize interference while simultaneously assigning a certain prescribed number of channels per cell. For this purpose free parameters are needed, which determine the relative importance of the individual criteria. We think that it is sometimes difficult to

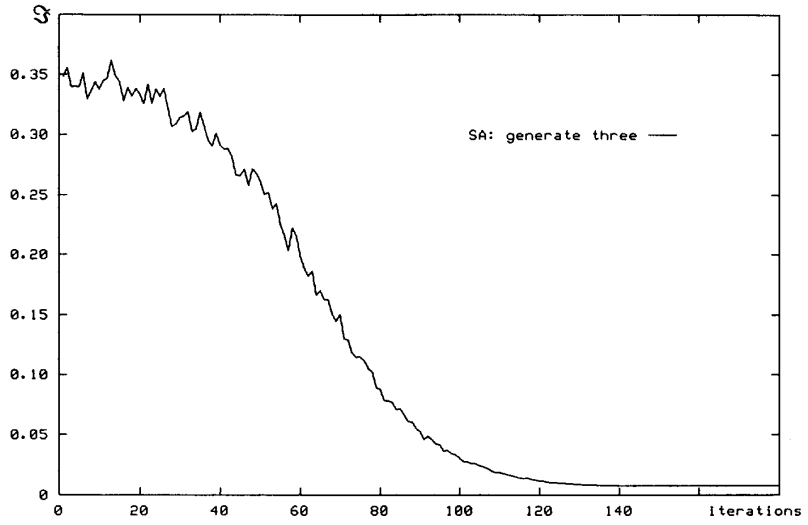


Fig. 6. Typical behavior of SA.

TABLE I

0.67	1	1
0.09	0	0
0.33	0	0
0.28	0	0
0.54	0	0
0.80	1	1
0.81	0	0
0.88	0	0
0.63	1	1
0.09	0	0
0.09	0	0
0.84	1	1
0.33	0	0
0.28	0	0
0.54	1	1
0.54	0	0
0.61	0	0
0.41	1	1
0.18	0	0
0.50	0	0
0.67	1	1
0.52	0	0
0.33	0	0
0.28	0	0
0.54	0	0
0.81	1	1
0.54	0	0
0.59	0	0
0.47	1	1
0.09	0	0
0.01	0	0
0.04	0	0
0.28	1	1
0.54	0	0
0.54	0	0
0.91	1	1
0.09	0	0
0.04	0	0
0.50	1	1
0.16	0	0
0.33	0	0
0.09	0	0
0.66	1	1
0.47	0	0
0.09	0	0
0.09	0	0
0.66	1	1
0.50	0	0
0.50	0	0
0.50	1	1

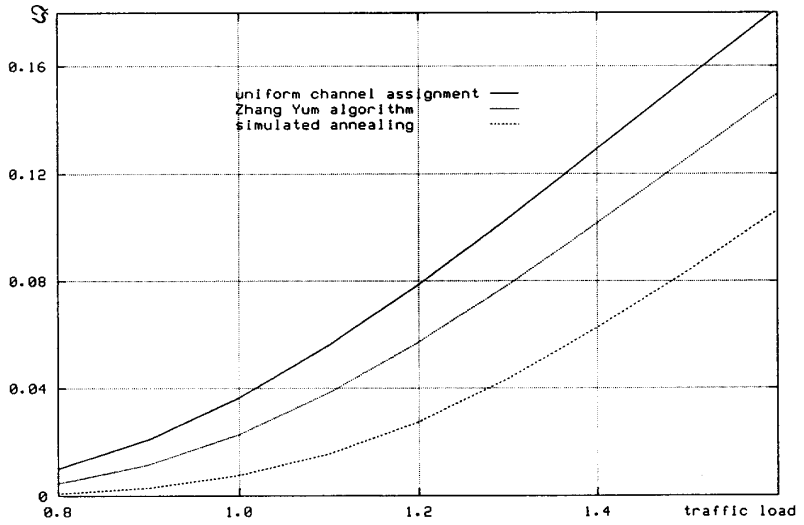


Fig. 7. Blocking probabilities with varying utilization.

argue which values of these parameters in the benefit function are reasonable.

In contrast, we use a well justified benefit function, but a more complicated state space. In updating, we never leave the space of admissible solutions, which of course makes the generate procedure more time consuming. We employ a very convenient representation of the state space, which allows us to treat a variety of neighborhood relations by using different generators (see e.g. Φ_1, Φ_2, Φ_3). This makes our approach very flexible, retaining theoretical convergence results.

Our numerical experiences with concrete implementations of simulated annealing show that this method is a very powerful tool for designing cellular radio networks.

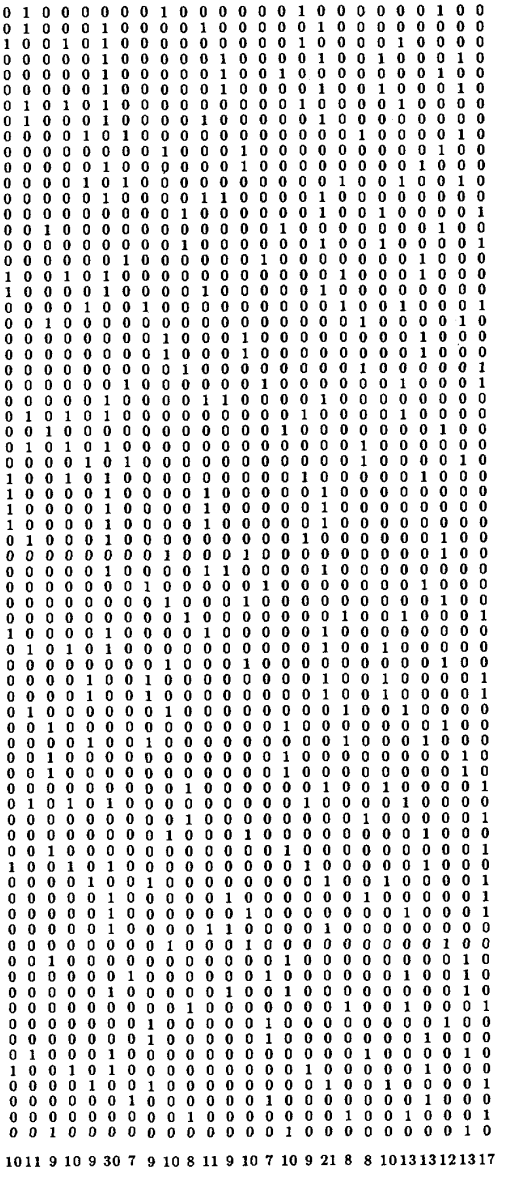
V. Numerical Experiences

This chapter tests the quality of simulated annealing solutions on different concrete examples. We treat three instances

TABLE II

0.67	1	1	1
0.09	0	0	0
0.33	0	0	0
0.28	0	0	0
0.54	0	0	0
0.80	0	0	0
0.81	0	0	0
0.88	1	1	1
0.62	0	0	0
0.09	0	0	0
0.09	0	0	0
0.83	1	1	1
0.33	0	0	0
0.28	0	0	0
0.54	0	0	0
0.54	0	0	0
0.61	1	1	1
0.41	0	0	0
0.18	0	0	0
0.50	0	0	0
0.67	1	1	1
0.52	0	0	0
0.33	0	0	0
0.28	0	0	0
0.54	1	1	0
0.81	0	0	1
0.54	0	0	0
0.59	0	0	0
0.47	1	1	0
0.09	0	0	1
0.01	0	0	0
0.40	0	0	0
0.28	0	0	0
0.54	0	1	0
0.54	1	0	0
0.90	0	0	1
0.09	0	0	0
0.04	0	0	0
0.50	1	1	0
0.16	0	0	0
0.33	0	0	0
0.09	0	0	0
0.66	1	1	1
0.47	0	0	0
0.09	0	0	0
0.09	0	0	0
0.66	1	1	1
0.50	0	0	0
0.50	0	0	0
0.50	0	0	0
0.50	0	0	0
0.83	1	1	1
0.50	0	0	0
0.50	0	0	0
0.33	0	0	0
0.09	0	0	0
0.83	1	1	1
0.66	0	0	0
0.50	0	0	0
0.79	0	0	0
0.83	1	1	1
0.75	0	0	0
0.79	0	0	0
0.66	0	0	0
0.66	1	1	1
0.33	0	0	0
0.47	0	0	0
0.66	0	0	0
0.75	1	1	1
0.75	0	0	0
0.66	0	0	0
0.50	0	0	0
0.85	1	1	1
0.75	0	0	0
0.75	0	0	0
0.50	0	0	0
0.75	0	0	0
0.33	0	0	0
0.66	0	0	0
0.79	1	1	1

TABLE III



rigorously and compare the results with competing approaches in the literature.

In all cases we have used the following simple cooling schedule. The temperature is decreased geometrically with factors around 0.94. The inner homogeneous loops of the corresponding Markov chains have length proportional to the problem size $z \cdot N$. We did not yet try to optimize the cooling schedule. Nevertheless the calculation times were moderate, most times less than 10 minutes on a SPARC 1+.

A. Example I

The first example deals with a regular structure of 12×12 hexagonal cells (see Fig. 5). We compare the SA assignment with the designs obtained by the algorithm of Zhang and Yum [27], developed for regular hexagonal structures, on the basis of the following scenario: $N = 150$ channels, the Erlang–B benefit function, cell utilization (in Erlangs) according to a uniform distribution on [2, 20], interferences occur between neighboring cells when the Euclidean distance between centers does not exceed$\sqrt{3}$ (cell radius = 1).

Fig. 6 shows the typical behavior of the benefit function in the run of SA. The bad starting value is due to an empty initial design and high temperature at the beginning. Escapes from local minima are clearly observable. This also happens at large iteration steps, here not observable because of the large scale of the y-axis. This behavior is typical

TABLE IV

cell number	6,9,17,22	4,7,18,19,23,25	16,21	8,14,15,20,24	10,11	3,5,12	1,13	2
channel demand	4	5	6	7	8	9	10	11
utilization λ	1.65	2.16	2.70	3.27	3.85	4.45	5.07	5.70

across different versions of SA: other versions show the same behavior.

In Fig. 7 we have varied the basic cell load (corresponding to 1.0 on the x-axis) by factors ranging from 0.8 to 1.6 in steps of 0.1. The curves show the blocking probability of the solutions corresponding to the indicated algorithms.

Uniform channel assignment allocates channels w.r.t. a minimal coloring of the regular structure (see [27]), independently of the utilization. This is a standard test often used in the literature.

Zhang and Yum [27] use the same approach via minimal colorings, but take into consideration the given utilization. These authors conjecture that the resulting designs are hardly improvable. That seems not to be true, as the comparison with SA results shows. We observe a significant improvement, uniformly over the whole utilization range. `generate_three` has been used to calculate the depicted curve, other generate procedures yield even slightly better results.

B. Example 2

The purpose of this example is to compare the quality of different generating procedures. We have investigated two linear networks, the first one with 50 stations and reuse distance 3 ($k = 2$), and the second one consisting of 80 stations with reuse distance 4 ($k = 3$), see Table I and II, respectively. One channel has to be assigned optimally under the standard model of Example 1 in Section II. In both tables, the first column contains the numbers $w_i f_i(1)$—derived from certain λ_i's and appropriately normalized—which occur in the benefit function (2.6). Optimal designs were calculated by the algorithm suggested by Theorem 2. These results are given in the second column.

In the third column of the 50 cells example, the solution found by SA is given. All three variants of `generate` produce the same optimal solution for this example.

In the third column of the 80 cells example, we see the solution of SA using `generate_one` and `generate_two`. This solution is also optimal. In contrast, the design represented in column four is not optimal. It has been determined by SA using the fast and simple `generate_three` procedure. This indicates that the assumptions for theoretical convergence of SA are quite sharp and by no means practically irrelevant.

In the 80 cells case we have also assigned 28 channels by iteratively using the optimal algorithm of Theorem 2. The resulting blocking probability is 2.48%. The best design achieved by SA has corresponding 2.44% blocking probability, which is a negligible difference (see Remark 3).

Note that there is a big jump in complexity between the two instances, in the first case we have roughly $1.37 \cdot 10^6$ admissible designs, in the second case $1.25 \cdot 10^9$.

C. Example 3

In the last example we deal with a real-world network. The basic data may be found in [16]. We have 25 cells and 73 channels, but do not take into account co-site constraints. In [16] optimal designs are obtained on the basis of a different benefit function. The key point is to satisfy local traffic demands. To satisfy certain prescribed local demands, at least 73 channels are necessary (see [16]). But even without co-site restrictions, there are at least 73 channels necessary, as may be seen by finding a corresponding clique. The cell utilizations themselves are not published in [16].

We apply the approach of Example 1, Section II, and assign the utilizations of Table IV to individual cells. SA yields very satisfactory results using `generate_one` with each of Ψ_1, Ψ_2, and Ψ_3 (cf. Lemma 3).

Table III shows the resulting design (transposed w.r.t. the notation of Section II). The last line contains the number of channels in each cell. As may be easily seen, the local traffic demands of [16] are fulfilled by this solution. The overall blocking probability is 1.088%. It is clear that a design which exactly satisfies the local channel demands—as is requested in [16]—is easily obtained by thinning the solution of Table III. Of course, we then end up with an inadmissible design.

In conclusion, we have investigated three completely different examples. In all cases SA gave very good results, though the procedure was not adapted to special networks, e.g. with respect to neighborhood structures. This indicates that SA is widely applicable with uniformly high performance and worth considering for future development.

REFERENCES

[1] E. Aarts and J. Korts, *Simulated Annealing and Boltzmann Machines.* New York: Wiley, 1989.

[2] T. Abe, K. Yoji, M. Sengoku, S. Tamura, and S. Shinoda, "Graph theoretical considerations of channel offsets systems in a cellular mobile system," *IEEE Trans. Veh. Technol.,* vol. 40, pp. 405–411, May 1991.

[3] F. Box, "A heuristic technique for assigning frequencies to mobile radio nets," *IEEE Trans. Veh. Technol.,* vol. 27, pp. 57–64, May 1978.

[4] M. Duque-Anton, D. Kunz, and B. Rüber, "Channel assignment using simulated annealing," *MRC,* pp. 121–128, Nov. 1991.

[5] M. Duque-Anton, D. Kunz, and B. Rüber, "Channel assignment for cellular radio using simulated annealing," *IEEE Trans. Veh. Technol.,* vol. 42, pp. 14–21, Feb. 1993.

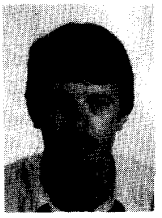
[6] Y. Furuya *et al.,* "2.6-Ghz band multipath characteristics measurement in a residential area for micro-cellular systems," in *Proc. IEEE Veh. Technol. Conf.,* Denver, CO, May 1992, pp. 423–426.

[7] A. Gamst, "Homogeneous distribution of frequencies in a regular hexagonal cell system," *IEEE Trans. Veh. Technol.,* vol. 31, pp. 132–144, Aug. 1982.

[8] A. Gamst, "Some lower bounds for a class of frequency assignment problems," *IEEE Trans. Veh. Technol.,* vol. 35, pp. 8–14, Feb. 1986.

[9] A. Gamst and K. Ralf, "Computational complexity of some interference graph calculations," *IEEE Trans. Veh. Technol.,* vol. 39, pp. 140–149, May 1990.

[10] M. R. Garey and D. S. Johnson, *Computers and Intractability.* New York: Freeman, 1979.
[11] B. Hajek, "Cooling schedules for optimal annealing," *Mathematics of OR,* vol. 13, pp. 311–329, May 1988.
[12] W. K. Hale, "Frequency assignment: Theory and applications," in *Proc. IEEE,* vol. 68, pp. 1497–1514, Dec. 1980.
[13] J. E. Hopcroft and J. D. Ullman, *Introduction to Automata Theory, Languages and Computation.* Reading, MA: Addison-Wesley, 1979.
[14] F. P. Kelley, "Blocking probabilities in large circuit-switched networks," *Adv. Appl. Prob.,* vol. 18, pp. 473–505, 1986.
[15] D. E. Knuth, *The Art of Computer Programming,* vol. 1. Reading, MA: Addison-Wesley, 1973.
[16] D. Kunz, "Channel assignment for cellular network using neural networks," *IEEE Trans. Veh. Technol.,* vol. 40, pp. 188–193, Feb. 1991.
[17] D. Kunz, "Suboptimum solutions obtained by the Hopfield-Tank neural network algorithm," *Biol. Cybern.,* vol. 65, pp. 129–133, 1991.
[18] W. C. Y. Lee, *Mobile Cellular Telecommunication Systems.* New York: McGraw-Hill, 1989.
[19] V. H. MacDonald, "The cellular concept," *Bell Syst. Techn. J.,* vol. 58, pp. 15–41, Jan. 1979.
[20] R. Mathar and D. Pfeifer, *Stochastik für Informatiker.* Stuttgart: Teubner-Verlag, 1990.
[21] E. J. Messerli, "Proof of a convexity property of the Erlang B formula," *Bell Syst. Techn. J.,* vol. 51, pp. 951–953, Apr. 1972.
[22] R. J. Pennotti, "Channel assignment in cellular mobile telecommunication systems," Master's thesis, Polytechnic Inst. of New York, Brooklyn, 1976.
[23] D. Robinson, "The optimality of fixed channel assignment policies for cellular radio systems," *Adv. Appl. Prob.,* vol. 24, pp. 474–495, 1992.
[24] D. J. A. Welsh, *Matroid Theory.* London: Academic Press, 1976.
[25] R. W. Wolf, *Stochastic Modeling and the Theory of Queues.* Englewood Cliffs, NJ: Prentice-Hall, 1989.
[26] W. Yue, "Analytical methods to calculate the performance of a cellular mobile radio communication system with hybrid channel assignment," *IEEE Trans. Veh. Technol.,* vol. 40, pp. 453–460, May 1991.
[27] M. Zhang and T. P. Yum, "The nonuniform compact pattern allocation algorithm for cellular mobile systems," *IEEE Trans. Veh. Technol.,* vol. 40, pp. 387–391, May 1991.



Rudolf Mathar was born in Kalterherberg, Germany, in 1952. He received his Dipl.Math. and Dr.Rer.Nat. degree in mathematics from Aachen University of Technology in 1978 and 1981, respectively.

In 1986/87 he worked at the European Business School as a lecturer in computer science, and in 1988/89 he joined a research group in applied optimization at the University of Augsburg. In October, 1989, he joined the faculty at Aachen University of Technology, where he is currently a Professor of Stochastics. He is especially interested in applications to computer science. His research interests are probabilistic performance analysis of networks and mobile data transmission systems.

Jürgen Mattfeldt was born in Stade, Germany, in 1965. He received his diploma degree in mathematics from Aachen University of Technology in 1991.

He is currently working at the department of Stochastics at Aachen University of Technology. His research interests include mobile communications systems, queueing theory, applied probability, and optimization.vskip-1.76pt