Channel Assignment for Mobile Communications Using Stochastic Chaotic Simulated Annealing

Sa Li[1] and Lipo Wang[2]

[1] School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798
p149552752@ntu.edu.sg
[2] School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798
elpwang@ntu.edu.sg
http://www.ntu.edu.sg/home/elpwang

Abstract. The channel assignment problem (CAP), the task to assign the required number of channels to each radio cell in such a way that interference is precluded and the frequency spectrum is used efficiently, is known to be an NP-complete optimization problem. In this paper, we solve CAP using a stochastic chaotic neural network that we proposed recently. The performance of stochastic chaotic simulated annealing (SCSA) is compared with other algorithms in several benchmark CAPs. Simulation results showed that this approach is able to further improve on results obtained by other algorithms.

1 Introduction

In recent years, there is a steady increasing demand for cellular mobile telephone communication systems. But the usable frequency spectrum is limited. Thus optimal frequency channel assignment is becoming more and more important and it can greatly enhance the traffic capacity of a cellular system.

Gamst and Rave defined a general form of channel assignment problems in an arbitrary inhomogeneous cellular radio network [1]: minimize the span of channels subject to demand and interference-free constraints. The CAP is usually solved by graph coloring algorithms [3]. Kunz used the Hopfield neural network for solving the CAP [2]. The algorithm minimizes an energy or cost function representing interference constraints and channel demand. Since then, various techniques have been explored for the application of neural networks to CAP. Funabiki used a parallel algorithm which does not require a rigorous synchronization procedure [10]. Chan et al proposed an approach based on cascaded multilayered feedforward neural networks which showed good performance in dynamic CAP [11]. Kim et al proposed a modified Hopfield network without fixed frequencies [12]. Smith and Palaniswami reformulated the CAP as a generalized quadratic assignment problem [8]. They then found remarkably good

J. Mira and A. Prieto (Eds.): IWANN 2001, LNCS 2084, pp. 757-764, 2001.

solutions to CAPs using simulated annealing, a modified Hopfield neural network, and a self-organizing neural network. Potential interferences considered in their paper come from the cochannel constraint (CCC), the adjacent channel constraint (ACC), and the co-site constraint (CSC) [1].

Simulated annealing is a general method for obtaining approximate solutions of combinatorial optimization problems [6]. Since the optimization processing is undertaken in a stochastic manner, it is also called stochastic simulated annealing (SSA). Convergence to globally optimal solutions is guaranteed if the cooling schedule is sufficiently slow, i.e., no faster than logarithmically [7]. In recent years, a large amount of work has been done in chaotic simulated annealing (CSA) ([12]-[21]). CSA can search efficiently because of its reduced search spaces. Chen and Aihara proposed a transiently chaotic neural networks (TCNN) which adds a large negative self-coupling with slow damping in the Euler approximation of the continuous Hopfield neural network so that neurodynamics eventually converge from strange attractors to an equilibrium point [13]. The TCNN showed good performance in solving traveling salesman problem (TSP). But CSA is deterministic and is not guaranteed to settle down at a global minimum no matter how slowly the annealing takes place. In [15], we generalized CSA by adding a decreasing random noise into the neuron inputs in the TCNN to combine the best feasures of both SSA and CSA, thereby abtaining stochastic chaotic simulated annealing (SCSA). In this paper, we show that SCSA can lead to further improvements on solutions for CAP.

This paper is organized as followings. Section 2 reviews the static channel assignment problem and its mathemmatical formulation as given in [8]. In section 3, the SCSA is reviewed. In the section 4, we apply SCSA to several benchmarking CAPs. Finally in section 5, we conclude this paper.

2 Static Channel Assignment Problem

In this section, we review the formulation of CAP as given by Smith and Palaniswami [8]. Suppose the number of available channels in an N-cell static mobile communication system is M and the number of channels requiremed in cell i is D_i. The minimum distances in the frequency domain by which two channels must be separated in order to guarantee an acceptably low signal/interference ratio in each region are stored in the compatibility matrix $C = \{C_{ij}\}$, an $n \times n$ symmetric matrix, where n is the number of cells in the mobile networks and C_{ij} is the minimum frequency separation between cell i and one in cell j. In most of cases, the number of available channels is less than the lower bound, i.e., the minimum number of channels required for an interference-free assignment. A useful version of the CAP is defined as follows [8]:

minimize	severity of interferences
subject to	demand constraints

The output of neuron j, k is:

$$X_{j,k} = \begin{cases} 1, \text{if cell j is assigned to channel } k & ; \\ 0, \text{otherwise} . \end{cases}$$

A cost tensor $P_{j,i,m+1}$ is used to measure the degree of interference between cells j and i caused by such assignments that $X_{j,k} = X_{i,l} = 1$ [8], where $m = |k - l|$ is the distance in the channel domain between channels k and l. The cost tensor P can be calculated recursively as follows:

$$P_{j,i,m+1} = \max\ (0, P_{j,i,m} - 1), \qquad for \quad m = 1, \cdots, M-1 \ , \tag{1}$$

$$P_{j,i,1} = C_{ji}, \qquad \forall j, i \neq j \ , \tag{2}$$

$$P_{j,j,1} = 0, \qquad \forall j \ . \tag{3}$$

Thus the CAP can be formulated to minimize the total interference of all assignments in the network:
minimize

$$F(X) \ = \sum_{j=1}^{N}\sum_{k=1}^{M} X_{j,k} \sum_{i=1}^{N}\sum_{l=1}^{M} P_{j,i,(|k-l|+1)} X_{i,l} \ , \tag{4}$$

subject to

$$\sum_{k=1}^{M} X_{j,k} \ = D_j, \qquad \forall j = 1, \cdots, N \ . \tag{5}$$

where $F(X)$ is the total interference.

3 Stochastic Chaotic Simulated Annealing

The stochastic chaotic simulated annealing (SCSA) is formulated as follows [15]:

$$x_{jk}(t) = \frac{1}{1 + e^{-y_{jk}(t)/\varepsilon}} \ , \tag{6}$$

$$y_{jk}(t+1) = k y_{jk}(t) + \alpha\Big(\sum_{i=1, i\neq j}^{N} \sum_{l=1, l\neq k}^{M} w_{jkil} + I_{ij}\Big) - z_{jk}(t)(x_j k(t) - I_0) + n(t) \ , \tag{7}$$

$$z_{jk}(t+1) = (1-\beta) z_{jk}(t) \qquad (i = 1, \cdots, n) \ , \tag{8}$$

$$A[n(t+1)] = (1-\beta) A[n(t)] \ , \tag{9}$$

where
x_{jk} : output of neuron j,k ;
y_{jk} : input of neuron j,k ;
$w_{jkil} = w_{iljk};\ w_{jkjk} = 0;\ \sum_{l=1, l\neq k}^{M} w_{jkil} + I_{ij} = -\partial E / \partial x_{jk}$:
connection weight from neuron j,k to neuron i,l ;
I_{jk} : input bias of neuron j, k ;

k : damping factor of nerve membrane ($0 \leq k \leq 1$);
α : positive scaling parameter for inputs ;
β : damping factor of the time dependent ($0 \leq \beta \leq 1$);
$z_{jk}(t)$: self-feedback connection weight or refractory strength ($z(t) \geq 0$) ;
I_0 : positive parameter;
ε : steepness parameter of the output function ($\varepsilon > 0$) ;
E : energy function;
$n(t)$: random noise injected into the neurons, in [-A, A] with a uniform distribution;
$A[n]$: the noise amplitude.

If n(t)=0 for all t, then the above noisy chaotic neural network reduces to the transiently chaotic neural network of Chen and Aihara [13].

The corresponding energy function E for CAP is given by (1) and (2):

$$E = \frac{W1}{2}\sum_{j=1}^{N}(\sum_{k=1}^{M} X_{jk} - D_j)^2 + \frac{W2}{2}\sum_{j=1}^{N}\sum_{k=1}^{M} X_{jk}\sum_{i=1}^{N}\sum_{l=1}^{M} P_{j,i,(|k-l|+1)}X_{il} \quad (10)$$

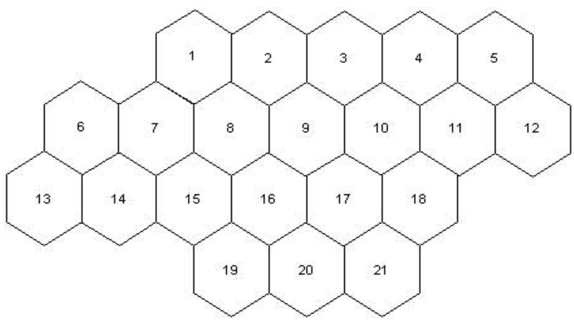

Fig. 1. Layout of a 21-cell hexagonal network

4 Application of SCSA to Benchmark CAPs

4.1 Descriptions of Various Problems

First we use the data set suggested by Sivarajan [3], denoted as EX1. The number of cells is $N = 4$, the number of channels available is $M = 11$, the demand of channels is given by $D^T = (1, 1, 1, 3)$. We also use a slightly larger extension of EX1, denoted as EX2 [8]:

$N = 5, M = 17, D^T = (2, 2, 2, 4, 3)$.

The Second example considered is the 21-cell cellular system (HEX1-HEX4) found in [4] (Fig.1). We used two sets of demands for the 21-cell as follows.

$D_1^T = (2, 6, 2, 2, 2, 4, 4, 13, 19, 7, 4, 4, 7, 4, 9, 14, 7, 2, 2, 4, 2)$;

$D_2^T = (1, 1, 1, 2, 3, 6, 7, 6, 10, 10, 11, 5, 7, 6, 4, 4, 7, 5, 5, 5, 6)$.

The details of HEX1-HEX4 are shown in Table 1 [8]. There are two different compatability matrices for HEX problems by considering the first two rings of cells around a particular cell as interferences [8]. The first one does not include ACC, so the relative off-diagonal terms C_{ij} are 1. The second one includes ACC, the off-diagonal terms of C_{ij} in second matrix are 1 or 2 respectively corresponding to CCC and ACC.

Table 1. Descriptions for hexagonal CAPs

Problem	N	M	D	co-channel	adjacent	C_{ii}
HEX1	21	37	D_1	yes	no	2
HEX2	21	91	D_1	yes	yes	3
HEX3	21	21	D_2	yes	no	2
HEX4	21	56	D_2	yes	yes	3

The final set of CAP is generated from the topographical data of an actual 24×21 km area around Helsinki, Finland [22]. Kunz calculated the traffic demand and interference relationships between the 25 regions around the base stations of this area. The compatability matrix is abtained from Kunz data as C_3 [8]. The demand vector is:

$D_3^T = (10, 11, 9, 5, 9, 4, 5, 7, 4, 8, 8, 9, 10, 7, 7, 6, 4, 5, 5, 7, 6, 4, 5, 7, 5)$.

This benchmarking CAP is divided into four classes by considering only the first 10 regions (KUNZ1), 15 regions (KUNZ2), 20 regions (KUNZ3), and the entire area (KUNZ4) (Table 2) [8].

4.2 The Simulation Results Of SCSA

We chose a set of parameters for each CAP as in Table 3.

Table 2. Descriptions for KUNZ problems

Problem	N	M	C	D
KUNZ1	10	30	$[C_3]_{10}$	$[D_3]_{10}$
KUNZ2	15	44	$[C_3]_{15}$	$[D_3]_{15}$
KUNZ3	20	60	$[C_3]_{20}$	$[D_3]_{20}$
KUNZ4	25	73	C_3	D_3

Our results are presented in Table 4. For comparison, Table 4 also includes results given in [8], i.e., the performances of GAMS/MINOS-5 (labeled GAMS), the traditional heuristics of steepest descent (SD), simulated annealing (SA), the original Hopfield network (HN) (with no hill-climbing), hill-climbing Hopfield

Table 3. The parameters for various CAPs

Problem	K	ε	I_0	α	β	z(0)	n(0)	W1	W2
EX1	0.9	1/250	0.65	0.0045	0.0005	0.1	0.5	1.0	0.02
EX2	0.9	1/250	0.65	0.0045	0.0005	0.1	0.5	1.0	0.02
HEX1	0.9	1/250	0.05	0.05	0.0005	0.08	0.08	1.0	0.25
HEX2	0.9	1/250	0.08	0.05	0.0005	0.08	0.08	1.0	0.25
HEX3	0.9	1/250	0.05	0.05	0.0005	0.08	0.08	1.0	0.2
HEX4	0.9	1/250	0.07	0.05	0.0005	0.08	0.08	1.0	0.3
KUNZ1	0.9	1/250	0.06	0.05	0.0004	0.08	0.08	1.0	0.45
KUNZ2	0.9	1/250	0.01	0.05	0.0005	0.08	0.08	1.0	0.45
KUNZ3	0.9	1/250	0.05	0.05	0.0005	0.08	0.08	1.0	0.45
KUNZ4	0.9	1/250	0.05	0.05	0.0005	0.08	0.08	1.0	0.45

network (HCHN), and the self-organizing neural network (SONN). Each of the heuristics is run from ten different random initial conditions. In Table 4, "Min" means the minimum total interference (eq. 4) found during these ten times, and "Av" is the average total interference [8]. We also computed the standard deviations (STDD) during these ten runs, as shown in Table 5.

Table 4. The results of SCSA and other techniques

	GAMS	SD		SA		HN		HCHN		SONN		SCSA	
problem	Min	Av.	Min	Av.	Min	Av.	Min	Av.	Min	Av.	Min	Av.	Min
EX1	2	0.6	0	0.0	0	0.2	0	0.0	0	0.4	0	0.0	0
EX2	3	1.1	0	0.1	0	1.8	0	0.8	0	2.4	0	0.0	0
HEX1	54	56.8	55	50.7	49	49.0	48	48.7	48	53.0	52	47.7	47
HEX2	27	28.9	25	20.4	19	21.2	19	19.8	19	28.5	24	18.5	18
HEX3	89	88.6	84	82.9	79	81.6	79	80.3	78	87.2	84	77.3	76
HEX4	31	28.2	26	21.0	17	21.6	20	18.9	17	29.1	22	17.2	16
KUNZ1	28	24.4	22	21.6	21	22.1	21	21.1	20	22.0	21	20.0	19
KUNZ2	39	38.1	36	33.2	32	32.8	32	31.5	30	33.4	33	30.3	30
KUNZ3	13	17.9	15	13.9	13	13.2	13	13.0	13	14.4	14	13.0	13
KUNZ4	7	5.5	3	1.8	1	0.4	0	0.1	0	2.2	1	0.0	0

The results in Table 4 show that the SCSA is able to further improve on results obtained by other approaches.

5 Conclusion

In this paper, we showed that stochastic chaotic simulated annealing (SCSA) [15] is very effective in solving combinatorial optimization problems, such as channel assignment problems in radio network planning. This approach has both noisy nature and chaotic

Table 5. The standard deviations for various CAPs

Problem	HEX1	HEX2	HEX3	HEX4	KUNZ1	KUNZ2	KUNZ3	KUNZ4
STDD	0.265	0.173	0.267	0.354	2.095	0.173	0.0	0.0

searching characteristics, so it can search in a smaller space and continue to search after the disappearance of chaos. Implementation of SCSA to solve other practical optimization problems will be studied in future work.

References

1. A. Gamst and W. Rave, "On frequency assignment in mobile automatic telephone systems," *Proc. GLOBECOM'82* , pp. 309-315, 1982.
2. D. Kunz, "Channel assignment for cellular radio using neural networks", *IEEE Trans. Veh. Technol.*, vol. 40, no. 1, pp. 188-193, 1991.
3. K. N. Sivarajan, R. J. McEliece and J.W. Ketchum, "Channel assignment in cellular radio", *Proc. 39th IEEE Veh. Technol. Soc. Conf.*, pp. 846-850, May, 1989.
4. A. Gamst, "Some lower bounds for a class of frequency assignment problems", *IEEE Trans. Veh. Tech.*, vol. VT-35, pp. 8-14, Feb., 1986.
5. W. K. Hale, "Frequency assignment: theory and application", *Pro. IEEE*, vol. 68, pp. 1497-1514, Dec., 1980.
6. M. Duque-Anton, D. Kunz, and B. Ruber, "Channel assignment for cellular radio using simulated annealing", *IEEE Trans. Veh. Technol.*, vol. 42, no. 1, February, 1993.
7. S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721-741, 1984.
8. K. Smith and M. Palaniswami, "Static and dynamic channel assignment using neural network", *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 2, Feb., 1997.
9. N. Funabiki and Y. Takefuji, "A neural network parallel algorithm for channel assignment problems in cellular radio networks", *IEEE Trans. Veh. Technol.*, vol. 41, no. 4, November, 1992.
10. P. Chan, M. Palaniswami and D. Everitt, "Neural network-based dynamic channel assignment for cellular mobile communication systems", *IEEE Trans. Veh. Technol.*, vol. 43, no. 2, May, 1994.
11. J. Kim, S. H. Park, P. W. Dowd and N. M. Nasrabadi, " Cellular radio channel assignment using a modified Hopfield network", *IEEE Trans. Veh. Technol.*, vol. 46, no. 4, November, 1997.
12. L. Chen and K. Aihara, "Chaotic simulated annealing by a neural network model with transient chaos", *Neural Networks*, vol. 8, no. 6, pp. 915-930, 1995.
13. L. Chen and K. Aihara, "Transient chaotic neural networks and chaotic simulated annealing", in M. Yamguti(ed.), *Towards the Harnessing of Chaos.* Amsterdam, Elsevier Science Publishers B.V. pp. 347-352, 1994.
14. L. Chen and K. Aihara, "Global searching ability of chaotic neural networks", *IEEE Trans. Circuits and Systems-I: Fundamental Theory and Applications*, vol. 46, no. 8, August, pp. 974-993, 1999.

15. L. Wang and F. Tian, "Noisy chaotic neural networks for solving combinatorial optimization problems", *Proc. International Joint Conference on Neural Networks.* (IJCNN 2000, Como, Italy, July 24-27, 2000)
16. F. Tian, L. Wang, and X. Fu, "Solving channel assignment problems for cellular radio networks using transiently chaotic neural networks", *Proc. International Conference on Automation, Robotics, and Computer Vision.* (ICARCV 2000, Singapore)
17. L. Wang, "Oscillatory and chaotic dynamics in neural networks under varying operating conditions", *IEEE Transactions on Neural Networks,* vol. 7, no. 6, pp. 1382-1388, 1996.
18. L. Wang and K. Smith, "On chaotic simulated annealing", *IEEE Transactions on Neural Networks,* vol. 9, no. 4, pp. 716-718, 1998.
19. L. Wang and K. Smith, "Chaos in the discretized analog Hopfield neural network and potential applications to optimization", *Proc. International Joint Conference on Neural Networks,* vol. 2, pp. 1679-1684, 1998.
20. T. Kwok, K. Smith and L. Wang, "Solving combinatorial optimization problems by chaotic neural networks", C. Dagli et al.(eds) *Intelligent Engineering Systems through Artificial Neural Networks* vol. 8, pp. 317-322, 1998.
21. T. Kwok, K. Smith and L. Wang, "Incorporating chaos into the Hopfield neural network for combinatorial optimization", *Proc. 1998 World Multiconference on Systemics, Cybernetics and Informatics*, N. Callaos, O. Omolayole, and L. Wang, (eds.) vol. 1, pp. 646-651, 1998.
22. T. Kohonen, "Self-organized formation of topologically correct feature maps", *Biol. Cybern.*, vol. 43, pp. 59-69, 1982.