Experiments with a Randomized Algorithm for a Frequency Assignment Problem

Janez Zerovnik

Laboratoire de l'Informatique du Parallélisme

Ecole Normale Supérieure de Lyon
Unité de recherche associée au CNRS n°1398

Experiments with a Randomized Algorithm for a Frequency Assignment Problem

Janez Žerovnik

Septembre 1997

Research Report N° 97-27

Ecole Normale Supérieure de Lyon
46 Allée d'Italie, 69364 Lyon Cedex 07, France
Téléphone : (+33) (0)4.72.72.80.00 Télécopieur : (+33) (0)4.72.72.80.80
Adresse électronique : lip@lip.ens−lyon.fr

Experiments with a Randomized Algorithm for a Frequency Assignment Problem

Janez Žerovnik

Septembre 1997

Abstract

The problems of assigning frequencies to transmitters can be naturally modelled by generalizations of graph coloring problems. We start with a randomized graph coloring algorithm of Petford and Welsh and propose a randomized algorithm for minimizing the number of constraints violated when a set of frequencies available is fixed. Experiments on instances of various types relevant to mobile communication networks are reported.

Keywords: frequency assignment problem, randomized heuristics, cellular telecommunications system

Résumé

Le problème de la planification de fréquence se traduit de façon naturelle en une généralisation du problème de coloriage de graphe. En partant d'un algorithme randomizé proposé par Petford and Welsh nous l'adaptons à la minimisation des violations de contraintes induites par la planification de fréquences. Des expérimentations ont été conduites sur des exemples variés significatif des problème de planification cellulaire.

Mots-clés: plannification de fréquences, heuristique randomizé, reseau cellulaire de radiocommunication

Experiments with a Randomized Algorithm for a Frequency Assignment Problem*

Janez Žerovnik[(1,2)]

September 16, 1997

(1)FME, University of Maribor, Smetanova 17, 2000 Maribor, Slovenia.
(2)IMFM, Jadranska 19, 1111 Ljubljana, Slovenia; janez.zerovnik@imfm.uni-lj.si

Abstract

The problems of assigning frequencies to transmitters can be naturally modelled by generalizations of graph coloring problems. We start with a randomized graph coloring algorithm of Petford and Welsh and propose a randomized algorithm for minimizing the number of constraints violated when a set of frequencies available is fixed. Experiments on instances of various types relevant to mobile communication networks are reported.

Keywords: frequency assignment problem, randomized heuristics, cellular telecommunications system.

1 Introduction

The frequency assignment (or frequency allocation) problem is one of the key applications in mobile networks engineering. The difficulty of practical problems comes from the fact that an acceptable solution must satisfy many constraints and the set of frequencies available is limited.

*Part of this work was done while the author was visiting Ecole Normale Supérieure de Lyon, supported by the French ministry of Education and Research. Partially supproted by the Ministry of Science and Technology of Slovenia, grant no. J2-7516-0101-97.

Many versions of the problem are intractable and therefore heuristics for finding near optimal solutions are sought. Various heuristics were recently used for frequency assignment including simulated annealing [6], genetic algorithms [13], mathematical programming [9], tabu search [4], etc. Many of these are compared in [11]. Most of the results published in the literature concern the span optimization, probably because strong lower bounds were developed which enable decisions to be made on the optimality of the results (see [2] for an overview of lower bounding techniques). The problem where a fixed spectrum is given and the objective is to minimize the number of violations is less studied. There are only few papers on this problem where the instances are either fully explained or available on the internet. A 7-cell cluster instance was used for testing performance of simulated annealing in [6]. Simulated annealing, tabu search and a genetic algorithm are compared in [12], where a set of benchmark problem instances is proposed. Here we compare performance of our algorithm to the results of these two papers.

In this paper we develop an algorithm for minimizing the number of constraints violated in a frequency assignment from a fixed set of available frequencies. The algorithm is based on the graph coloring algorithm of Petford and Welsh [15]. It is not unlike to more well known general optimization techniques such as simulated annealing and generalized Boltzmann machines. The main difference is that here the 'temperature' is fixed, which results in a simpler tuning of the parameter(s). On the other hand, this may be one of the reasons for good performannce because it is known that the probability of finding an optimal solution with simulated annealing algorithm is asymptoticaly worse that the same probability for local search [7].

The paper is organized as follows: Section 2 introduces the frequency assignment problem while Section 3 presents the algorithm; Section 4 reports on experiments and Section 5 gives conclusions.

2 The frequency assignment problem

In this section we give precise meaning to our frequency assignment problem for which our algorithm was designed. It is a combinatorial optimization problem in which can be naturally defined using weighted graphs. We use the usual graph theory terminology, see for example [24].

Throughout the paper we assume that frequencies are taken from a fixed subset of natural numbers, $F \subset \mathcal{N}$. This covers a usual practical situation where frequency bands are given but there are also some reserved frequencies which may not be used. These so called *forbidden frequencies* may exist due to technological or environmental constraints (government regulation, other radio systems...).

Specific constraints may be defined on neighboring pairs of transmitters. In general, a constraint associated with an edge can be any set of *forbidden differences*. Usually, frequency separation constraints exist between geographically close transmitters in the service area. In this case we speak of far-site interference and the sets of forbidden differences are of the form $\{0, 1, \ldots, w-1\}$.

Another important type of constraints is based on co-site interference. Namely, if there are more frequencies needed at the same location, they must be separated at least by co-site difference, which is usually larger than other.

In general, it is possible to define more complicated constraints modelling different types of interference which is due to various interference mechanisms such as harmonic constraints, adjacent channel constraints, co-site frequency constraints, intermodulation products and spurious emisions and responses. In this paper we will consider only co-site and far-site interference, because these two represent the most important interference problems to avoid.

We therefore have a set of locations (transmitters) with given demands and given constraints between pairs of transmitters. This can be modeled as a problem of multicoloring a graph with constraints.

If we regard a location with demand d as d vertices of a graph G, the problem reduces to finding one frequency for each vertex. Edges of G connect pairs of vertices, for which there exists a constraint.

For arbitrary two nodes i and j we call w_{ij} the *weight* of the edge ij.

A *frequency planning function* $c : F \to V(G)$ assigns a frequency to every vertex of G.

If c is the frequency planning function, every constraint can be written as

$$|c(i) - c(j)| \geq w_{ij} \qquad (1)$$

A frequency planning function c is *proper*, if it violates no constraint. $E(c)$, the *cost* of c is the number of constraints c violates.

Clearly $E(c) \geq 0$ and $E(c) = 0$ if and only if c is proper.
The frequency assignment problem studied here is

Problem: (MINIMAL COST) FREQUENCY ASSIGNMENT *Input: weighted undirected graph G and a set F* *Question: find an assignment which minimizes the cost E*

A closely related problem is the problem of minimizing the *span*, i.e. minimizing the difference between the largest and the smalest frequency used.

Problem: MINIMAL SPAN *Input: weighted undirected graph G and a set F* *Question: find a proper assignment which minimizes the span*

Note that these problems are clearly at least as hard as the graph coloring problems. (Just take all weights equal to 1 and $F = \{1, 2, \ldots, k$ or $F = \mathcal{N}$.)

3 The Algorithm

Petford and Welsh proposed a randomized algorithm for 3-coloring which mimics the behavior of a physical process based on multi-particle system of statistical mechanics called the antivoter model [15]. The algorithm starts with a random initial 3-coloring of the input graph and then applies an iterative process. In each iteration a vertex creating a conflict is randomly taken uniformly and recolored according to some probability distribution. This distribution favors colors which are less represented in the neighborhood of the chosen vertex. There is a straightforward generalization of this algorithm to k-coloring, which behaves reasonably good on various types of graphs [26, 18, 20, 5].

We use the same main idea, with some natural generalizations. A set of colors is replaced by a finite subset of natural numbers corresponding to available frequencies. Simple constraints requesting different colors or frequencies at adjacent vertices are generalized to constraints depending on the edge weights and applying to frequencies assigned to adjacent vertices.

The frequency assignment algorithm is

Algorithm PW(T, Time_limit)

```
assign available frequencies to nodes of the graph uniformly random;
while not stopping condition do
      select a bad vertex v (randomly)
      assign a new frequency to v
end while
```

Bad vertex is selected uniformly random among vertices which are endpoints of some edge which violates a constraint. A new frequency is assigned at random from the set F.

Sampling is done according to the probability distribution defined as follows:

The probability of frequency $i \in F$ to be chosen as a new frequency of the vertex v is proportional to

$$exp(-S_i/T) = \Theta^{-S_i}$$

where S_i is the number of edges with one endpoint at v and violating the constraint provided frequency of v is set i. T is parameter of the algorithm, called *temperature* for reasons explained later.

The second parameter of the algorithm is the time limit, given as the maximal number of iterations of the while loop. This is at the same time the number of calls to the function which computes a new frequency and also the number of feasible solutions of the problem generated (some of them may be counted more than once).

The stopping condition we choose is: either a proper assignment was found or a time limit was reached. In the later case, the best solution found (with fewest constraints violated) is reported as an approximate solution.

In the rest of the section we give some remarks on the parameters of the algorithm.

In the coloring problem, S_i is simply the number of neighbors of vertex v colored by i. The original algorithm of Petford and Welsh (for coloring) uses probabilities proportional to 4^{-S_i}, which corresponds to $T = 0.72\ldots$. Larger values of T result in higher probability of accepting a move which increases the number of bad edges. With low values of T, the algorithm behaves very much like iterative improvement.

T is a parameter of the algorithm, which may be called temperature because of the analogy to the temperature of the simulated annealing algorithm and to the temperature of the Boltzmann machine neural network. This analogies follow from the following simple observation. Let us denote the old color of the current vertex by i and the new color by j. The number of bad edges E' after the move is

$$E' = E - S_j + S_i$$

where E is the number of bad edges before the change. We define $\Delta E = E - E' = S_j - S_i$. At each step j is fixed and hence S_j and E are fixed. Consequently, it is equivalent to define the probability of choosing color i to be proportional to either $\exp(-S_i/T)$, $\exp(\Delta E/T)$ or $\exp(E'/T)$.

Finally, recall that the number of bad edges is a usual definition of energy function in simulated annealing and Boltzmann machine. Therefore, the algorithm PW is in close relationship to constant temperature operation of the generalized Boltzmann machine (for details, see [17] and the references there). The major difference is in the 'firing' rule. While in Boltzman machine all neurons are fired with equal probability, in PW algorithm only bad vertices are activated. The algorithm PW differs from constant temperature simulated 'annealing' in the acceptance criteria for the moves improving the cost function. These are always accepted by simulated annealing and only according to some (high) probability in the present algorithm.

Choosing the temperature and the time limit is in general an open problem. However, very simple tuning which is explained in the section on experiments was enough to obtain good results on instances tested.

4 Problem instances and results

We give results of experiments with various datasets. Where available, we compare our results with the results reported in the literature. This includes two implemantations of simulated annealing, a tabu serach and a genetic algorithm. For some datasets, in lack of other information, we compare results to a similar type algorithm for minimizing the span[22]. The algorithm of [22] uses similar probability distribution for choosing the new frequency, but it starts with a greedy assignment and tries to reduce the span. In the examples, where the optimal span is known, it is possible to compare the results of the algoritm PW with results of [22]. Comparison in terms of the number of steps is fair, because the basic steps of both algorithms are of the same time complexity (a change of a frequency of one vertex, both according to some probability distribution).

4.1 Random generated benchmark problems

First dataset is the instances tested in [12], which are available on the internet.

We first did a simple tuning of the parameter T. With relatively small time limit (set to $1000n$ iterations, where n is the number of vertices of the graph) we run the algorithm on a medium size instance (410 vertices) at temperatures $T = 0.1, 0.2, 0.3, 0.4$. Since the results were promising around $T = 0.2$, we checked also $T = 0.15$ and $T = 0.25$.

T	min	max	average
0.1	383	396	389.6
0.15	370	401	383.9
0.2	372	387	379.2
0.25	360	394	378.7
0.3	379	404	386.6
0.4	395	432	416.7

We have no formal argument why to start with these temperatures. We know from experiments that $T = 0.72$, the original temperature of Petford and Welsh was often too high and that some lower temperatures do better in the case of graph coloring [18, 19]. There is at least one seemingly more advanced approach for tuning the temperature. One could check some values of T by short runs measuring the ratio between 'up' and 'down' moves. If the temperature is relatively low, the algorithm will behave like iterative improvement and this ratio will be close to 0. On the other hand, if the temperature is too high, the ratio is close to 1. It is therefore possible to get at least some reasonable starting value for T by bisection.

Since the temperatures $T = 0.2$ and $T = 0.25$ seemed to be good, we tested the algorithm with these two values on the whole available dataset of [12].

Because the results above were good we did not increase the time limit. We compare the results to those given in [12] in the following table.

	PW, $T = 0.2$			PW, $T = 0.25$			SA[12]		GA[12]		TS[12]	
N	min	max	avg.	min	max	avg.	min	avg.	min	avg.	min	avg.
252	22	27	24.2	24	28	25.8						
282	120	137	128.3	130	140	134.8	121	128.5	501	506.9	128	134
410	372	387	379.2	360	394	378.7	367	380.4	1160	1173.0	379	400.8
450	160	170	164.7	157	172	165.7	151	161.4	845	849.0	176	188.0
490	629	670	648.2	624	657	642.9	625	636.1	1828	1834.4	658	678.9
726	1512	1620	1562.9	1522	1593	1549.1	1488	1509.8	3947	3960.8	1535	1602.8

The quality of results of the algorithm PW is comparable or a little lower that that of the simulated annealing algorithm (SA). It is definitely better that quality of the results obtained by tabu search (TS) and by genetic algorithm (GA).

To make a comparison fair, we have take into account the number of assignments tested as reported in [12], which corresponds to the number of iterations in our algorithm. GA explored 2×10^5 assignments for all instances. The numbers of assignments for the other two problems were between 2.7×10^6 and 2.4×10^8 for SA and between 1.3×10^8 and 2.1×10^8 for TS.

For this reason we increased the time limit by 10 (to $10000n$) to let the algorithm PWA generate about the same number of assignments as SA and TS. It should be recalled that the number of assignments allowed for GA in [12] was lower, but the quality of the results of GA was also much lower.

Next table gives the results of this experiment. (The best solutions known are in boldface.) In the last column we put the best solutions found by the FASoft software package [3]. It is claimed that the wall clock time for the experiment reported in [3] was the same as in [12].

	PW,$T=0.2$			PW,$T=0.25$			FASoft[12]
n	min	max	average	min	max	average	min
252	19	21	19.700	20	21	20.500	**9**
282	**110**	116	113.100	112	119	115.500	114
410	**343**	366	354.300	347	357	315.600	361
450	147	154	151.100	140	154	145.400	**130**
490	607	644	625.800	601	640	616.800	**580**
726	1465	1568	1515.000	**1463**	1527	1497.600	1466

Two remarks are in order here. The number of configurations visited for some larger instances was in our experiment much lower than reported in [12]. Second, we wish to say that we used very simple tuning of temperature T (and on one graph only). Probably, even better performance can be achieved if T is tuned with some more effort.

4.2 7-cluster hexagonal torus

The second example is a uniform 7-cell cluster of 14×14 cells arranged in a doubly periodic array - or torus - where interference extends to the second ring of neighboring cells. The channel assignment task is to equip each cell with two out of fourteen channels. The 7-cluster is a standard test case for regular cell assemblies [6].

10 runs of the algorithm at $T = 0.2, 0.3$ and 0.4 and time limit set to 392000 iterations gave the following results:

- no success at $T = 0.2$,
- 8 out of 10 successes at $T = 0.3$ with average number of iterations 307738 and
- no success at $T = 0.4$.

10 longer runs gave average 241350.90 of 10 successful runs (min= 70285, max= 673334, time limit 3920000 iterations).

For comparison, we recall the results with the algorithm of [22] on the same instance: with temperature $T = 0.3$ we also obtained 10 successes with 156 636 average number of steps (only 2 successes at $T = 0.2$ and no success at $T = 0.4$).

There is no information on the number of assignments tested by the simulated annealing implementation of [6]. It is reported that time used was several hours on T800. In our experiment, the average run (approx. 250.000 iterations) corresponds to approximately 5 minutes of wall clock time on SPARC 5 workstation.

4.3 Triangular lattice graphs with random demand

Typical examples which are studied in the context of frequency assignment are triangular lattices with blowups. The demand for the number of calls to be served at the same time is usually not uniform. All constraints are of the form: $|c(i) - c(j)| \geq 1$ if $ij \in E(G)$. This corresponds to a problem of assigning a set of frequencies to each transmitter, which is a multicoloring problem of the corresponding graph.

The same problem is usually presented in terms of coloring of graphs as follows. Each vertex, corresponding to a transmitter with demand > 0 is expanded to a clique, i.e. a complete graph and each vertex of such a clique is connected to all neighbors of the original vertex. Vertices of demand 0 are deleted. The problem of coloring of the resulting graph is equivalent to the multicoloring problem of the original graph.

In our experiment we generated the instances as follows:

1. generate $a \times b$ triangular lattice
2. assign a random number $q(v)$ between q_1 and q_2 to every vertex.
3. replace each vertex with v by a $q(v)$ clique.

For graphs generated by the above procedure, it is easy to compute their clique number, i.e. the size of maximal clique. This is because every clique in the resulting graph is emerged from a clique (a triangle or an edge) of the original graph and its size is exactly the sum of $q(.)$'s of its original vertices.

In the following table we give results on instances used in [22]. Since $T = 0.2$ was good choice there, we took the same value of T here. We compare the number of iterations needed with those needed by the algorithm for minimizing the span of [22] in the following table.

instance	n	$\chi(G) = \omega(G)$	PW,$T = 0.2$	$T = 0.2$[22]
(10x10 lattice, demand 1-3)	199	9	1875.77(100)	2015.71(100)
(5x5 lattice, demand 20-40)	741	29	13522.26(100)	15558.60(100)
(10x10 lattice, demand 5-10)	764	113	26955.08(100)	21372.83(100)

(The numbers in parenthesis are the numbers of successful runs.)

It may be interesting to note that for all graphs the number of colors needed was equal to the clique number and hence all the solutions are optimal.

However, it is not easy to find examples of triangular latice graphs with blowups for which the chromatic number is greater than the clique number. In fact, it can be shown that $\omega \leq \chi \leq 8\lceil\frac{\omega}{6}\rceil$ for any triangular lattice graph with blowups, even with arbitrary co-site difference (see [23] and the references there).

4.4 Clique of size 12 with constraints

The last example involves a complete graph of 12 vertices each with demand 3; both frequency separation and forbidden frequencies are added to the graph. The co-site diference is 3 and the differences on edges are given by the matrix [16]:

```
0  4  8  1  5  3  2  6  2  3  7  1
4  0  4  3  1  5  2  2  6  1  3  5
8  4  0  7  3  1  6  2  2  5  1  3
1  3  7  0  4  4  1  5  3  2  6  2
5  1  3  4  0  4  3  1  5  2  2  6
3  5  1  4  4  0  5  3  1  6  2  2
2  2  6  1  3  5  0  4  4  1  5  3
6  2  2  5  1  3  4  0  4  3  1  5
2  6  2  3  5  1  4  4  0  5  3  1
3  1  5  2  2  6  1  3  5  0  4  4
7  3  1  6  2  2  5  1  3  4  0  4
1  5  3  2  6  2  3  5  1  4  4  0
```

The frequency range is: 40 to 99 (40 and 99 inclusive) but the following frequencies are not allowed (for all transmitters):

52 53 54 55 56 57 58 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90

Since the underlying graph of this problem is small it is possible to obtain a solution to this problem by a hand drawing and some reasoning. Such a structured solution for example is:

Vertex index	1	2	3	4	5	6
Frequencies	43,62,91	47,66,95	51,70,99	44,63,92	48,67,96	40,59,71
Vertex index	7	8	9	10	11	12
Frequencies	45,64,93	49,68,97	41,60,72	46,65,94	50,69,98	42,61,73

This problem is far from a coloring problem, since the graph we get after expanding the locations is a complete graph on 36 vertices so there are clearly 36 frequencies needed. The difficulty lies in obeying the weights (i.e. the minimal differences) and forbidden frequencies.

The problem itself seems to be hard in spite of its small size.

Hundred runs (with time limit 12000 for each run) at $T = 0.2$ gave solutions with one up to 5 violated constraint (average 3.220).

A batch of 10000 runs with the same parameters did not find any proper assignment.

This is worse than the results with the span minimization algorithm, with which 4 optimal assignments were found in 1000 runs (with the same parameters as above) [22].

5 Concluding remarks

We tested Petford Welsh type randomized algorithm on frequency assignment problem with fixed set of available frequencies. The preliminary results are promising.

In this section we first discuss the problem of tuning the parameters of the algorithms and continue with remarks on of possible (more or less straightforward) generalizations of the present algorithm.

The main difficulty in practical application of any randomized heuristics is to tune the algorithm parameters for the data we process. In our case, tuning means choosing the right temperature and the maximum number of iterations of each trial. When applying the Petford Welsh type algorithms to graph coloring problems, it was observed that for different types of graphs the performance of the algorithm considerably depends on T [18, 20]. Choosing a good temperature is therefore an interesting open research problem which is not unlike to the well known problem of finding a good cooling schedule for the simulated annealing algorithm [14]. Not surprisingly, temperature is also important parameter of the present algorithm. The fact that it is a single real number gives hope that it can in praxis be tuned by not too complicated and time consuming process. Furthermore, our first experience shows that it can be tuned relatively fast, at least for the types of problem instances had. Therefore, our algorithm seems to be easier to adapt than some other randomized algorithms such as genetic algorithms or simulated annealing.

There are a lot of optimization problems referred to as frequency assignment problems. First, they differ in the level at which they model the interference. Here the simplest example is graph coloring, where all constraints only impose different frequencies at adjacent sites. On the other hand, it is in principle possible to design very complicated constraints modelling different types of interference which are due to various interference mechanisms such as harmonic constraints, adjacent channel constraints, co-site frequency constraints, intermodulation products and spurious emisions and responses. The cost function to optimize can also be defined in many ways, the sum of violated constraints used here being only a simple example. In practical problems some constraints are probably more costly if violated than others [25]. In principle, we do not see any problem to adopt the present algorithm to such more general situations.

Furthermore, there is a whole family of problems, where the frequencies have to be assigned to edges of a graph. This is the case when it is known in advance which pairs of users will need a communication link, although such problems may be presented also in a form of a vertex coloring problem [1]. However, we believe that this problems are probably more naturally solved by generalizations of edge coloring methods. This may be rewarding since it is well known that edge coloring problems in graph theory are usually easier than the corresponding vertex coloring problems.

References

[1] K.I.Aardal, C.A.J.Hurkens, J.K.Lenstra, S.R.Tiourine, *Algorithms for Frequency assignment Problems* CWI Quaterly 9 (1996) 1-8. (see also `http://www.ph.kcl.ac.uk/ hcd/CALMA/index.html`)

[2] S.M.Allen, D.H.Smith, S.Hurley, *Lower Bounding Techniques for Frequency Assignment*, sub-

mitted to Discrete Mathematics. (`http://www.cs.cf.ac.uk/User/Steve.Hurley/`)

[3] S.M.Allen, D.H.Smith, S.Hurley, S.U.Thiel, *Using Lower Bounds for Minimum Span Frequency Assignment*, 2nd International Conference on Metaheuristics MIC97, Sophia-Antipolis, 1997.

[4] D.J.Castelino, S.Hurley and N.M.Stephens, *A Tabu Search Algorithm for Frequency Assignment*, to appear Annals of Operations Research.

[5] B.Chamaret, S.Ubeda, J.Žerovnik, *A Randomized Algorithm for Graph Coloring Applied to Channel Allocation in Mobile Telephone Networks*, in Proceedings of the 6th International Conference on Operational Research KOI'96 (T.Hunjak, Lj.Martić, L.Neralić, eds.), 25-30.

[6] M.Duque-Antn, D.Kunz, B.Rüber, *Channel Assignment for Cellular Radio Using Simulated Annealing*, IEEE Transactions on Vehicular Technology 42 (1993) 14-21.

[7] A.Ferreira, J.Žerovnik, *Bounding the probability of success of stochastic methods for global optimization*, Journal of computers & mathematics with applications 25 (1993) 1-8.

[8] M.R.Garey, D.S.Johnson, *Computers and Intractability*, Freeman, San Francisco 1979.

[9] A.I. Giortzis and L.F. Turner, *A Mathematical Programming Approach to the Channel Assignement Problem in Radio Networks*, IEEE Vehicular Technology Conf,736-740, Atlanta, USA, 1996.

[10] M. Grevel and A. Sachs, *A Graph Theoretical Analysis of Dynamic Channel Assignment Algorithms for Mobile Radiocommunication Systems*, Siemens Forsch.-u Entwickl-Ber Bd 12, number 5 (1983) 298-305.

[11] S.Hurley, D.H.Smith, S.U.Thiel, *FASoft: A Software System for frequency Assignment*, to appear in Radio Science. (`http://www.cs.cf.ac.uk/User/Steve.Hurley/`)

[12] S.Hurley, S.U.Thiel, D.H.Smith, *A Comparison of Local Search Algorithms for Radio Link Frequency Assignment Problems*, 11th Annual ACM Symposium on Applied computing, Philadelphia, USA, 1996, 251-257. (a set of benchmark problems at `http://www.cs.cf.ac.uk/User/Steve.Hurley/`.)

[13] F.J. Jaimes-Romero, D. Munoz-Rodriguez, *Channel Assignment in Cellular Systems Using Genetic Algorithms*, IEEE Vehicular Technology Conf, 741-745, Atlanta, USA, 1996.

[14] P.J.M. Laarhoven and E.H.L. Aarts, *Simulating Annealing, Theory and applications*, D. Reidel Publishing Company, Dordrecht, 1987.

[15] A.Petford, D.Welsh, *A Randomised 3-colouring Algorithm*, Discrete Mathematics 74 (1989) 253-261.

[16] Personal communication from Mauro Manela, Mobile Systems International Plc, 1 Harbour Exchange Square, London E14 9GE, UK.

[17] J.Shawe-Taylor, J.Žerovnik, *Boltzmann Machines with Finite Alphabet*, Artificial Neural Networks 2, vol.1., Brighton, U.K. (1992) 391-394.

[18] J.Shawe-Taylor, J.Žerovnik, *Analysis of the Mean Field Annealing Algorithm for Graph Colouring*, Journal of Artificial Neural Networks 2 (1995) 329-340.

[19] J.Shawe-Taylor, J.Žerovnik, *Adapting temperature for some randomized local search algorithms*, in preparation.

[20] P.Burge, J.Shawe-Taylor, J.Žerovnik, *Graph Coloring by Maximal Evidence Edge Adding*, submitted.

[21] D.H. Smith, S. Hurley, *Bounds for the Frequency Assignment Problem*, Discrete Mathematics 167/168, (1997) 571-582.

[22] S.Ubéda and J.Žerovnik, *A Randomized Algorithm for a Channel Assignment Problem*, to appear in Speedup.

[23] S.Ubéda and J.Žerovnik, *Upper bounds for the span of a frequency assignment in triangular lattice graphs with blowups*, research report.

[24] R.Wilson, *Introduction to Graph Theory*, third edition, Longman Scientific and Technical, Harlow 1985.

[25] D.Youngs, *Frequency Assignment for Cellular Radio Networks*, in the IMA Conference series (number 60) 'Applications of Combinatorial Mathematics', C. Mitchell ed., (1995), 233 - 241.

[26] J.Žerovnik, *A Randomized Algorithm for k–colorability, Discrete Mathematics*, vol.131 (1994) 379–393.