Minimizing Norms to Solve Approximately a Frequency Assignment Problem

Martin J. Gander and Gene H. Golub

Abstract

We show that solving the frequency assignment problem is equivalent to solving a minimization problem involving spectral norms of matrices, if the link gain matrix is symmetric. Often the spectral norm is ideal for minimization problems, but not so in this case. We reformulate the minimization using the one-, infinity- and Frobenius norm. This allows us to deal with non symmetric link gain matrices as well.

We propose two types of algorithms to solve these new minimization problems approximately. The algorithms minimizing the one- and/or infinity-norm are exchanging rows and/or columns based on the weights of the elements therein whereas the algorithm minimizing the Frobenius norm is derived from techniques in spectral bisection of graphs.

Numerical results show that for regular distributions of base stations the algorithm minimizing the Frobenius norm computes good approximations, whereas for irregular distributions of base stations the algorithms minimizing the one- and/or infinity-norm are far superior.

1 Introduction

Portable phone systems can be used almost everywhere, because there is a dense net of base stations, to which the weak portable phone can connect. Each base station covers its neighborhood and when you move with your portable phone from the neighborhood of one station into the neighborhood of a different one while calling, your call is automatically transferred from the old station to the new one. The dense covering of areas with base stations causes problems: If two base stations use the same frequency for a concurrent connection with corresponding portable phones, the signals may interfere and the quality of

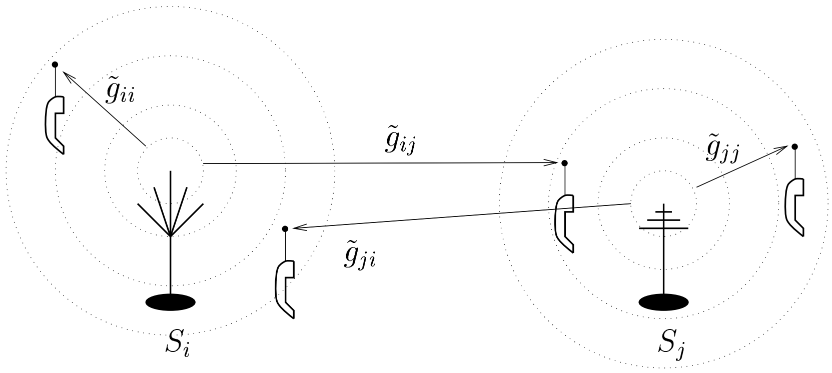

Figure 1: Two base stations and positions of worst reception and interference

the communication decreases. This is the case in particular if the two base stations are physically close to each other. Therefore one would like to assign different frequencies to different base stations. However, in practice, there are less frequencies available than base stations and thus frequencies have to be reused. Naturally the problem arises how to assign frequencies to base stations so that the interference among simultaneous calls is minimized.

This problem is well known under the name *Frequency Assignment Problem* in the literature. It has been shown to be NP-hard by reducing it to the graph coloring problem [3]. Many different algorithms have been proposed to compute an approximate solution. Park and Lee applied a graph coloring algorithm in [7] which is based on a modified Kernighan-Lin algorithm. Baldi and Posner derived lower and upper bounds on the number of frequencies needed using graph theory in [2]. Hao and Dorne applied a genetic algorithm to the frequency assignment problem in [4]. The relation between eigenvalue decompositions and graph coloring has been established in Aspvall and Gilbert [1]. For a precise description of the physical problem the reader is referred to the paper by Knällmann and Quellmalz [5]. Their paper also proposes a simulated annealing algorithm for the approximate solution of the frequency assignment problem.

We derive the frequency assignment problem using the continuous analog of the discrete derivation presented in [5]. Consider n base stations S_i, $i = 1, 2, \ldots, n$. We assume first that each station S_i sends with the same frequency at a power level p_i to describe how the power level affects interference. We look at pairs of base stations S_i and S_j as given in Figure 1. We define the entries of the link gain matrix $\tilde{G} = [\tilde{g}_{ij}]$ as follows:

- $\tilde{g}_{ii}p_i$ is the minimum signal power the phone connected with station S_i (sending with power p_i) receives anywhere in the area of S_i. For practical purposes, Quellmalz proposes to use a discrete set of test points in in the area of S_i [5] to determine this quantity approximately in a real situation.

- $\tilde{g}_{ij} p_j$ is the maximum interference power the phone connected with station S_i receives anywhere in the area of S_i from the sender S_j (sending with power p_j).

In the following we neglect natural noise from other sources. The worst possible signal to noise ratio q_i of a phone connected with station S_i can now be defined [9] as the ratio between the minimum signal power from the station S_i and the sum of the interference powers $\tilde{g}_{ij} p_j$ of the other stations S_j, $j \neq i$,

$$q_i := \frac{\tilde{g}_{ii} p_i}{\sum_{j \neq i} \tilde{g}_{ij} p_j} = \frac{p_i}{\sum_{j \neq i} \frac{\tilde{g}_{ij}}{\tilde{g}_{ii}} p_j} = \frac{p_i}{\left(\sum_j \frac{\tilde{g}_{ij}}{\tilde{g}_{ii}} p_j \right) - p_i} \tag{1.1}$$

Defining the normalized link gain matrix $G = [g_{ij}]$ by

$$g_{ij} := \begin{cases} \frac{\tilde{g}_{ij}}{\tilde{g}_{ii}} & i \neq j \\ 0 & \text{otherwise} \end{cases} \tag{1.2}$$

we can write the worst signal to noise ratio from (1.1) as

$$q_i = \frac{p_i}{\sum_j g_{ij} p_j}. \tag{1.3}$$

In the simple example with one frequency we have only the sending power p_i at each station S_i to maximize the smallest signal to noise ratio q_i over all stations. The optimum is obtained when the signal to noise ratio is the same for all stations. This can be seen as follows: Starting with all the q_i equal and decreasing the sending power p_j of station S_j we would necessarily decrease the signal to noise ratio q_j since $g_{jj} = 0$. On the other hand increasing the power p_j of station S_j would necessarily decrease the signal to noise ratio of coupled neighboring stations, because $g_{jk} \geq 0$.

Since the matrix G is non negative, we have by the Perron Frobenius theory that the largest eigenvalue λ_{max} of G is real and the corresponding eigenvector $\boldsymbol{\phi}_{max}$ is non negative. Using this eigenvector as our power distribution, $\boldsymbol{p} := \boldsymbol{\phi}_{max}$ we obtain for the signal to noise ratio

$$q_i = \frac{p_i}{\sum_j g_{ij} p_j} = \frac{p_i}{\lambda_{max} p_i} = \frac{1}{\lambda_{max}} =: q_{least} \tag{1.4}$$

and thus all the stations can guarantee at least a signal to noise ratio q_{least}, the best possible solution with one frequency. If one has several frequencies, the goal is to collect stations in groups so that for each group q_{least} is maximized, or in other words, the largest eigenvalue of the link gain matrix G for each group is minimized.

In the next section we formulate the subproblem of n base stations and two frequencies mathematically. In Section 3 we show the relation of the frequency assignment problem with norms and propose two strategies to find an approximate solution to the two frequency problem. In Section 4 we show numerical experiments on random matrices, matrices constructed from random base station locations and on matrices which we got from Ericsson. In Section 5 we extend the strategies to several frequencies.

2 The two Frequency Problem

Given a normalized link gain matrix $G \in \mathbb{R}^{n\times n}$ which has zeros on the diagonal and non-negative entries otherwise,

$$G = \begin{bmatrix} 0 & g_{12} & g_{13} & \cdots & g_{1n} \\ g_{21} & 0 & g_{23} & \cdots & g_{2n} \\ \vdots & & \ddots & & \vdots \\ g_{n1} & g_{n2} & \cdots & \cdots & 0 \end{bmatrix}, \quad g_{ij} \geq 0, 1 \leq i, j \leq n,$$

we want to find a decomposition of a permutation Π of the matrix G into four sub blocks,

$$\Pi^T G \Pi = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \quad \begin{array}{ll} A \in \mathbb{R}^{p\times p}, & B \in \mathbb{R}^{p\times(n-p)}, \\ C \in \mathbb{R}^{(n-p)\times p}, & B \in \mathbb{R}^{(n-p)\times(n-p)}, \end{array}$$

such that the maximum of the largest eigenvalues of the sub matrices on the diagonal is minimized,

$$\min_{\Pi,p}(\max(\lambda_{max}(A), \lambda_{max}(D))). \tag{2.1}$$

An exhaustive search to solve the above minimax problem is currently feasible for matrices of size up to 20×20, for which one has to try 2^{20} possibilities. Our brute force code needs $2.117e10$ flops for the complete search in that case. Since the matrices arising in the radio communication problems are of order 100×100, a feasible strategy to compute approximations to the best solution is needed.

3 Approximation Strategies

3.1 Relation with Norms

Suppose the normalized link gain matrix G is symmetric, which would hold in an environment without obstacles. Then G is diagonalizable, $G = Q^T \Lambda Q$ where Q is orthogonal and Λ is diagonal with the eigenvalues of G on the diagonal. Thus

$$||G||_2 = ||Q^T \Lambda Q||_2 = ||\Lambda||_2 = \lambda_{max}(G)$$

and the minimax problem in (2.1) can be written equivalently using the spectral norm,

$$\min_{\Pi,p}(\max(||A||_2, ||D||_2)). \tag{3.1}$$

Often the spectral norm is ideal for minimization problems, but in this case different norms are easier to handle. We investigate algorithms which approximately minimize the one and infinity norm,

$$\begin{aligned} ||A||_1 &:= \max_j \sum_i |a_{ij}| \\ ||A||_\infty &:= \max_i \sum_j |a_{ij}| \end{aligned}$$

and the Frobenius norm,

$$||A||_F := \sqrt{\sum_{ij} |a_{ij}|^2}.$$

The physical intuition behind the use of different norms is the following: If we manage to keep only weak links (small link gain) in the submatrices A and D and put strong links (large link gain) into the off diagonal blocks B and C then the strong links lose their importance, because they link stations with different frequencies. So physically minimizing norms of A and D is appropriate even in the non symmetric case.

3.2 The One and Infinity Norm

We consider the minimization of the one norm,

$$\min_{\Pi,p}(\max(||A||_1, ||D||_1)) \tag{3.2}$$

and the minimization of the infinity norm,

$$\min_{\Pi,p}(\max(||A||_\infty, ||D||_\infty)) \tag{3.3}$$

simultaneously because of their duality. We use the following notation to describe our strategies:

- α_i, β_i, γ_i and δ_i denote the rows of the submatrices A, B, C and D.
- a_j, b_j, c_j and d_j denote the columns of the submatrices A, B, C and D.
- $\overline{\alpha}_i$, $\overline{\beta}_i$, $\overline{\gamma}_i$ and $\overline{\delta}_i$ denote the row sums of the submatrices A, B, C and D.
- $\overline{a}_j$, $\overline{b}_j$, $\overline{c}_j$ and $\overline{d}_j$ denote the column sums of the submatrices A, B, C and D.

There is an additional motivation to minimizing the one and/or infinity norm coming from the Gershgorin theorem. This theorem states that all the eigenvalues of $A \in \mathbb{R}^{p\times p}$ lie in the disk about the origin with the radius $r = \min_{1\leq i,j\leq p}(\overline{\alpha}_i, \overline{a}_j)$. Hence by decreasing the disk radius for A and D the maximum eigenvalue of A and D will be confined to a smaller and smaller disk. To obtain a decrease in the disk radius, we therefore want to perform permutations which move columns or rows with large sum from A and D out into B and C. For a fixed splitting parameter p we have to chose an index $i \in 1, 2, \ldots, p$ and an index $j \in p+1, p+2, \ldots, n$ and swap column i with j and row i with j. We propose the following strategies to choose i and j:

Strategy1 Choose the index i such that

$$\overline{\alpha}_i + \overline{a}_i - (\overline{c}_i + \overline{\beta}_i) = \max_s(\overline{\alpha}_s + \overline{a}_s - (\overline{c}_s + \overline{\beta}_s)),$$

and index j such that

$$\overline{\delta}_j + \overline{d}_j - (\overline{b}_j + \overline{\gamma}_j) = \max_s(\overline{\delta}_s + \overline{d}_s - (\overline{b}_s + \overline{\gamma}_s)),$$

which means simultaneously maximizing the weight loss in the submatrices A and D. There is no freedom in this strategy to choose between different possibilities for the indices i and j and so the algorithm is completely specified.

Strategy2 Compute the index k with the largest row and column sum in A, $\overline{\alpha}_k + \overline{a}_k := \max_s(\overline{\alpha}_s + \overline{a}_s)$, the index l with the largest row and column sum in D, $\overline{\delta}_l + \overline{d}_l := \max_s(\overline{\delta}_s + \overline{d}_s)$, the index m with the smallest row sum in C and smallest column sum in B, $\overline{\gamma}_m + \overline{b}_m := \min_s(\overline{\gamma}_s + \overline{b}_s)$ and the index o with the smallest column sum in C and smallest row sum in B, $\overline{\beta}_o + \overline{c}_o := \min_s(\overline{\beta}_s + \overline{c}_s)$. Now we can choose for i either k or o and for j either l or m, which gives us four different possible moves at each step. We complete the strategy in two different ways:

a) If $\lambda_{max}(A) > \lambda_{max}(D)$ choose $i := k$ and $j := m$, otherwise $i := m$ and $j := l$. This means we choose the maximum weight loss possible for the matrix with the bigger eigenvalue.

b) Choose randomly $i \in \{k, o\}$ and $j \in \{m, l\}$.

Strategy3 Compute the indices with the largest row or column sums for the matrices A and D,

$$\begin{array}{rr} \overline{\alpha}_{k_r} := \max_s(\overline{\alpha}_s), & \overline{a}_{k_c} := \max_s(\overline{a}_s), \\ \overline{\delta}_{l_r} := \max_s(\overline{\delta}_s), & \overline{d}_{l_c} := \max_s(\overline{d}_s) \end{array}$$

and the indices with the smallest row and column sums for the matrices B and C,

$$\begin{array}{rr} \overline{\gamma}_{m_r} := \min_s(\overline{\gamma}_s), & \overline{b}_{m_c} := \min_s(\overline{b}_s), \\ \overline{\beta}_{o_r} := \min_s(\overline{\beta}_s), & \overline{c}_{o_c} := \min_s(\overline{c}_s). \end{array}$$

Now we can choose for i either k_r, k_c, o_r or o_c and for j either l_r, l_c, m_r or m_c, which gives us sixteen different possible moves at each step. We complete the strategy in three different ways:

a) If $\lambda_{max}(A) > \lambda_{max}(D)$ choose $i := k_r$ if $\overline{\alpha}_{k_r} > \overline{a}_{k_c}$ else $i := k_c$ and $j := m_r$ if $\overline{\gamma}_{m_r} < \overline{b}_{m_c}$ else $j := m_c$. Otherwise choose $i := o_r$ if $\overline{\beta}_{o_r} < \overline{c}_{o_c}$ else $i := o_c$ and $j := l_r$ if $\overline{\delta}_{l_r} > \overline{d}_{l_c}$ else $j := l_c$.

b) If $\lambda_{max}(A) > \lambda_{max}(D)$ choose randomly $i \in \{k_r, k_c\}$ and $j \in \{m_r, m_c\}$. Otherwise choose randomly $i \in \{o_r, o_c\}$ and $j \in \{l_r, l_c\}$.

c) Choose randomly $i \in \{k_r, k_c, o_r, o_c\}$ and $j \in \{m_r, m_c, l_r, l_c\}$.

3.3 The Frobenius Norm

Suppose we want to minimize the Frobenius norm of the two sub blocks A and D,

$$\min_{\Pi,p}(\max(||A||_F^2, ||D||_F^2)). \tag{3.4}$$

Let $\boldsymbol{p} \in \mathbb{R}^n$ be a partitioning vector, $p_i \in \{-1, 1\}$. Then the sum of the Frobenius norms of A and D can be written as

$$\begin{aligned} ||A||_F^2 + ||D||_F^2 &= \frac{1}{4}\sum_{ij} g_{ij}^2(p_i + p_j)^2 \\ &= \frac{1}{4}\sum_{ij} g_{ij}^2(p_i^2 + p_j^2) + \frac{1}{2}\sum_{ij} g_{ij}^2 p_i p_j \\ &= \frac{1}{2}(||G||_F^2 + \boldsymbol{p}^T G^2 \boldsymbol{p}) \end{aligned}$$

and hence minimizing the sum of the two Frobenius norms $||A||_F^2 + ||D||_F^2$ is equivalent to minimizing

$$\min_{p_i \in \{-1,1\}} \boldsymbol{p}^T G^2 \boldsymbol{p} \tag{3.5}$$

where the square on G is understood to be element wise. This formulation corresponds to the spectral bisection problem in graph partitioning [8]. It is shown to be NP-hard as well in the literature. Spectral algorithms have been applied to the graph coloring problem in [1] and graph coloring algorithms have in turn been used to find approximate solutions to the frequency assignment problem in [7]. We are however not aware of any direct application of the spectral algorithm to the frequency assignment problem.

To find an approximation to the solution of (3.5) one relaxes the constraint on $\boldsymbol{p}$ to contain only elements $+1$ or -1 and allows arbitrary $\boldsymbol{p}$. Thus the minimum is obtained for $\boldsymbol{p}$ being the eigenvector of G^2 associated with the smallest eigenvalue $\lambda_{min}(G^2)$.

4 Numerical Experiments

We performed three different sets of experiments. The first set of experiments uses completely random matrices with no relation to a physical setting. The second set of experiments uses matrices that come from a random distribution of base stations using that signals decay at the rate one over the distance to the power of three. The last set of experiments is performed on example matrices we received from Ericsson.

4.1 Random Matrices

We show first two experiments with random matrices $G \in \mathbb{R}^{10\times 10}$. Figure 2 shows on the left one experiment for a particular matrix G and on the right the averaged results over 100 random matrices G. The best solution found by the exhaustive search is given by a solid line in the lower part of the graphs. Note that there are no iterations required and the line is provided only to be able to compare with the other strategies. The same applies for the spectral algorithm denoted by stars. The graphs always show the best solution found so far, not the actual solution at each step. Otherwise the curves would not be necessarily monotonically decreasing. After about five iterations the deterministic algorithms stagnate at a local minimum, whereas the randomized algorithms continue to decrease the spectral radius of the submatrices A and D. They tend towards the the optimal solution.

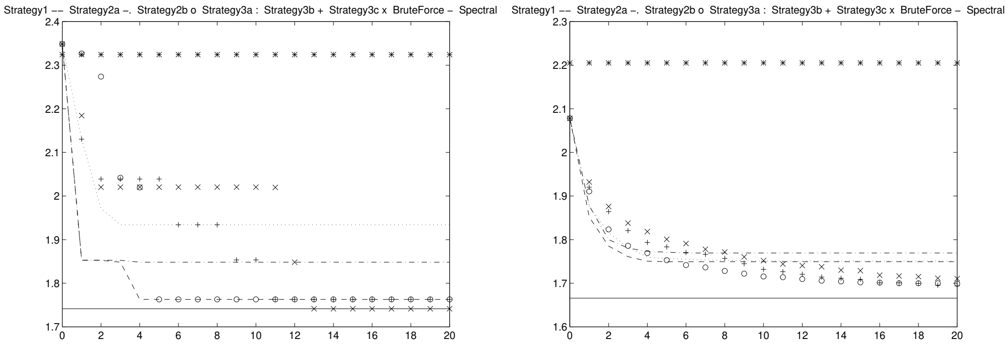

Figure 2: One experiment on a random matrix $G \in \mathbb{R}^{10\times 10}$ on the left and the average over 100 experiments with random matrices of the same size

It is interesting that the spectral algorithm does not perform well at all in the completely random case. This changes dramatically when one considers more physical situations.

Figure 3 shows the averaged results over hundred random matrices $G \in \mathbb{R}^{100\times 100}$. In this case we can not compute the best solution any more, but the randomized algorithms converge as in the case of small matrices to a solution which is probably close to the optimal solution. Note that Strategy1 converges on average after about 30 steps and stagnates from there on, whereas the randomized algorithms continue to improve the solution as the iteration progresses. We did not include the spectral algorithm in those experiments, since it does not perform well in the case of small matrices already.

4.2 Matrices Generated from Random Station Locations

The matrices tested in the previous subsection were completely random and do therefore not represent realistically the class of matrices one would have to deal with in practice. To get a more realistic set of such matrices, we proceed as follows: Generate randomly locations in the xy-plane for base stations and compute the link gain matrix entries by using that the signal strength decays like one over the distance to the power three. Having the n locations of base stations (x_i, y_i), $i = 1, 2, \ldots, n$ we get the link gain matrix

$$g_{ij} := \begin{cases} \frac{1}{((x_i-x_j)^2+(y_i-y_j)^2)^{\frac{3}{2}}} & i \neq j \\ 0 & \text{otherwise} \end{cases} .$$

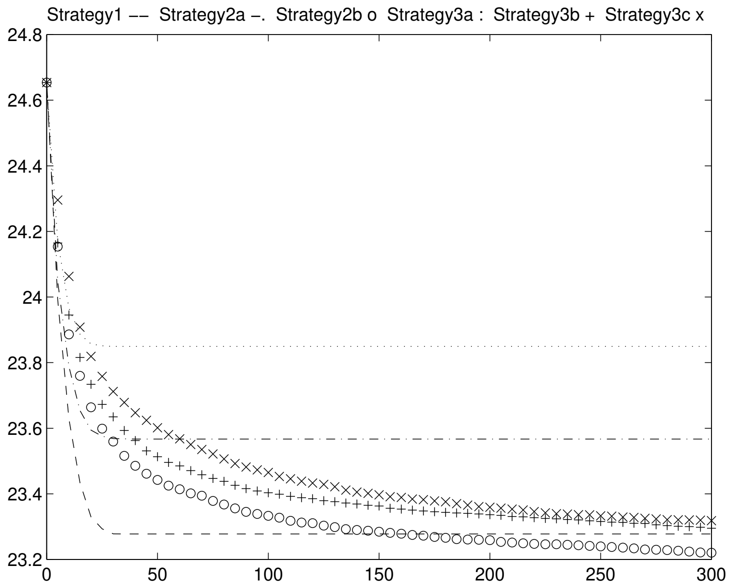

Figure 3: The average over hundred experiments with random matrices $G \in \mathbb{R}^{100\times 100}$

Figure 4 shows the average over 100 experiments using matrices $G \in \mathbb{R}^{10\times 10}$ generated from random base station locations.

The situation becomes even more realistic if one takes into account that one tries to distribute base stations evenly over an area to be covered. We choose to put each base station into a corresponding square of a square grid, but let them deviate randomly from the center of the square by a small amount determined by a parameter δ. More precisely if $\delta = 0$ the stations are located exactly at the center of each square, whereas if $\delta = 1$ they can be anywhere within the square. Two typical situations for different values of the parameter δ are displayed in Figure 5, together with the solution found by the spectral algorithm on the left and Strategy1 on the right. The frequency assignment is displayed by a plus sign for one frequency and a dot for the other one.

For well distributed base stations, the parameter δ being small, the best solution is indicated by physical intuition: Each station has four neighbors which are closest. So all those neighbors should be assigned a different frequency. This corresponds to a red black ordering of the normalized link gain matrix G. Note that the spectral algorithm finds this solution, if the stations are very regularly distributed. For larger values of the parameter δ however, where the stations are less organized, the spectral algorithm fails and Strategy1 computes a better solution.

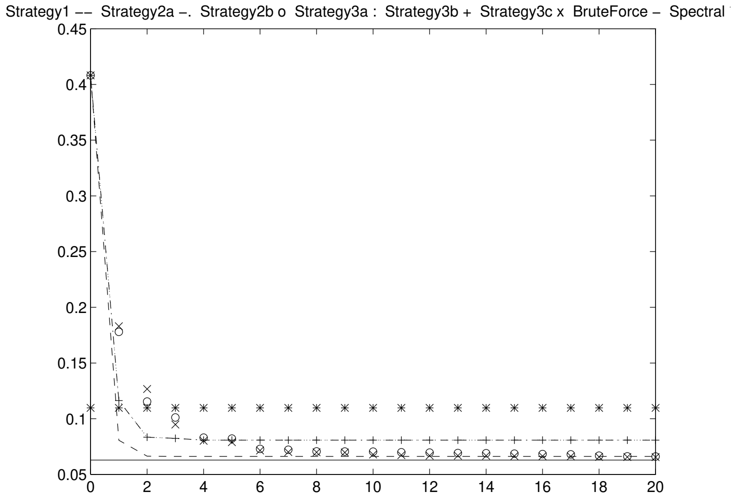

Figure 4: The average over 100 experiments with matrices $G \in \mathbb{R}^{10\times 10}$ generated from random base station locations

Figure 6 shows the average over 100 experiments with matrices generated using the parameter $\delta = \frac{1}{5}$ on the left and $\delta = \frac{4}{5}$ on the right. There are two interesting observations:

1. The deterministic Strategy1 performs very well on the problems which come from well distributed sets of base stations. It finds a close to optimal solution within about 4 iterations. The random strategies take longer and do not improve much on the solution found deterministically with Strategy1.
2. The spectral algorithm performs extremely well on well distributed sets of stations. It finds the optimal solution in every case. However as soon as the distribution is disturbed considerably, the performance of the spectral method decreases.

4.3 Matrices from Ericsson

We received a real test problem and a constructed one from Ericsson in Sweden. The matrix $G_1 \in \mathbb{R}^{81\times 81}$ comes from a real application. Figure 7 shows on the left one experiment on the matrix G_1 and on the right the averaged results over 100 experiments using the randomized algorithms. Clearly the randomized algorithms are superior to the deterministic ones which get stuck in local minima.

The second matrix G_2 is constructed putting base stations on a regular 10×10 grid. Note that in this idealized case periodic boundary conditions are used, so that a station on the right boundary of the rectangular grid is considered as a neighbor of the corresponding

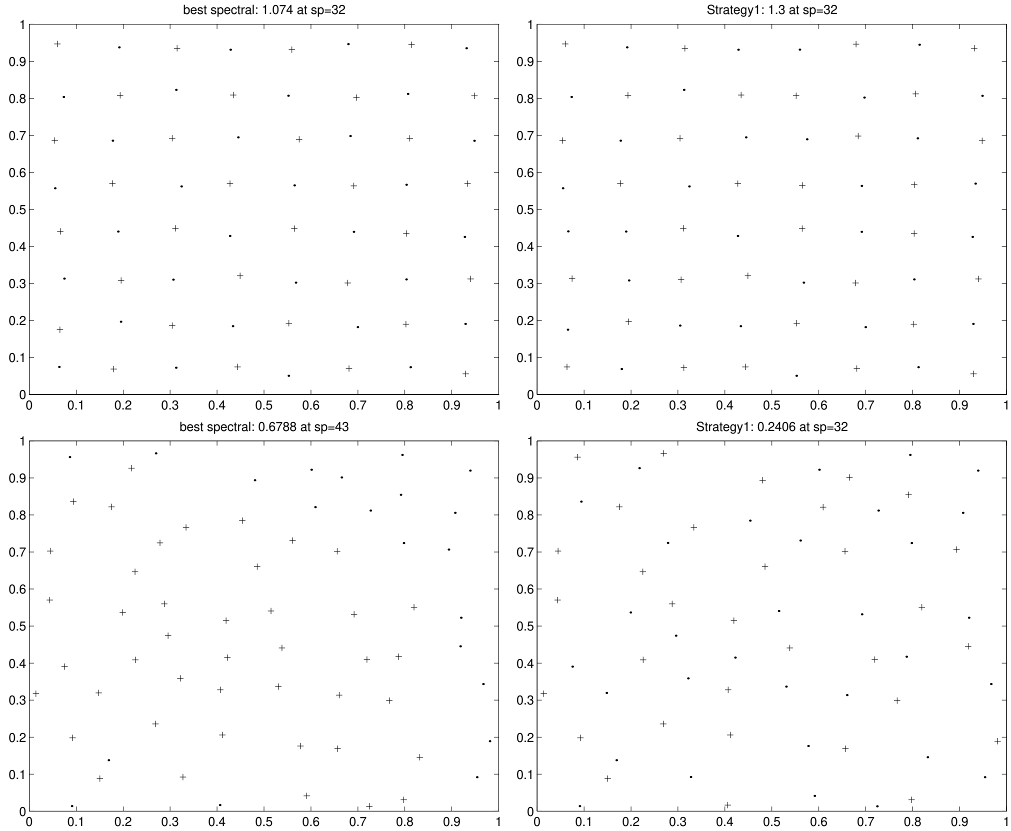

Figure 5: Random locations with $\delta = \frac{1}{5}$ on top and $\delta = \frac{4}{5}$ at the bottom. On the left the solution of the spectral algorithm and on the right the one with Strategy1

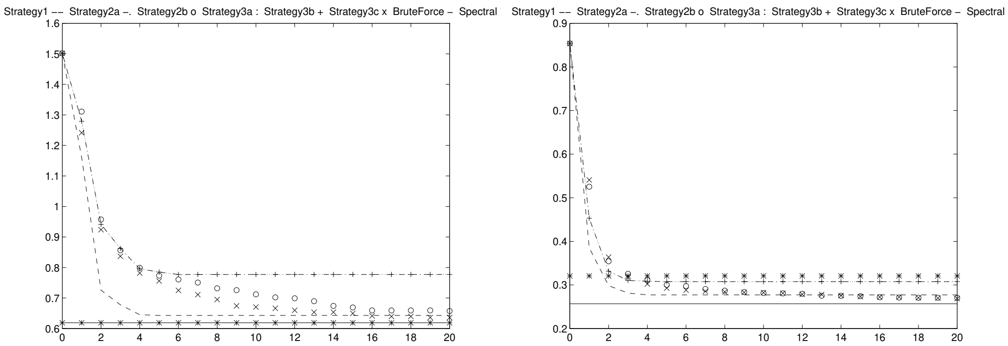

Figure 6: Experiments on a quasi organized distribution of base stations. On the left the average over 100 experiments with well distributed base stations, $\delta = \frac{1}{5}$, and on the right with a more disturbed distribution, $\delta = \frac{4}{5}$

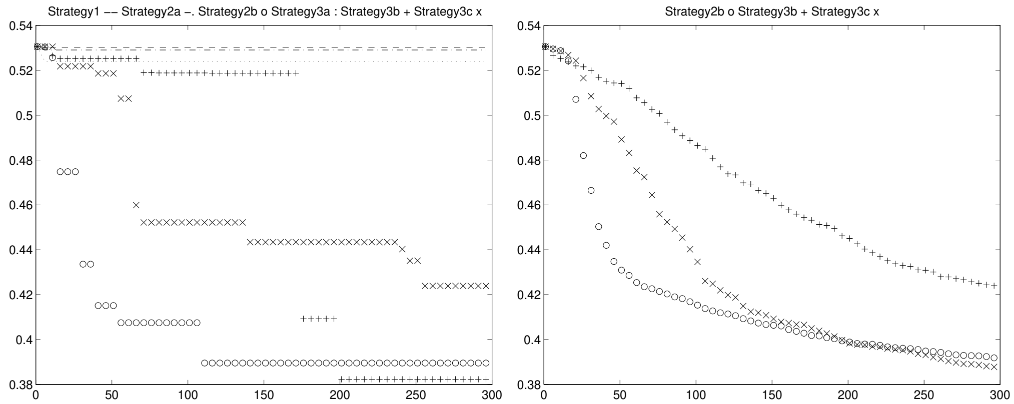

Figure 7: One experiment with the matrix G_1 from Ericsson stemming from a real application on the left and the average over 100 experiments with the randomized algorithms on the same matrix G_1

station on the left boundary. Figure 8 shows in the top row the original matrix G_2 on the left, and on the right the red black ordering indicating the best possible solution. In the bottom row of Figure 8 on the left there is the best solution found by Strategy1 and on the right the solution found by the spectral method. The result with red black ordering is only 6% better than the result found by Strategy1, and identical to the result found by the spectral algorithm which also finds a red black ordering, but the rows in the red respectively black sub matrix are differently ordered.

Figure 9 shows how the algorithms perform on this special symmetric example. It is interesting to note that for this special matrix, all the strategies seem to perform about equally well. None of the algorithms get stuck before they reach a solution close to the best one. Note also the excellent performance of the spectral algorithm.

5 Generalization to Several Frequencies

We generalize the algorithm for several frequencies by recursively applying the algorithm described above. Suppose we are given 2^m frequencies. Then we partition the matrix first into two submatrices assuming that there are two frequencies available. Then we recursively partition the two submatrices again so that after m steps we have a partition of the matrix into 2^m submatrices on the diagonal. Figure 10 shows how the algorithms performs on two examples with

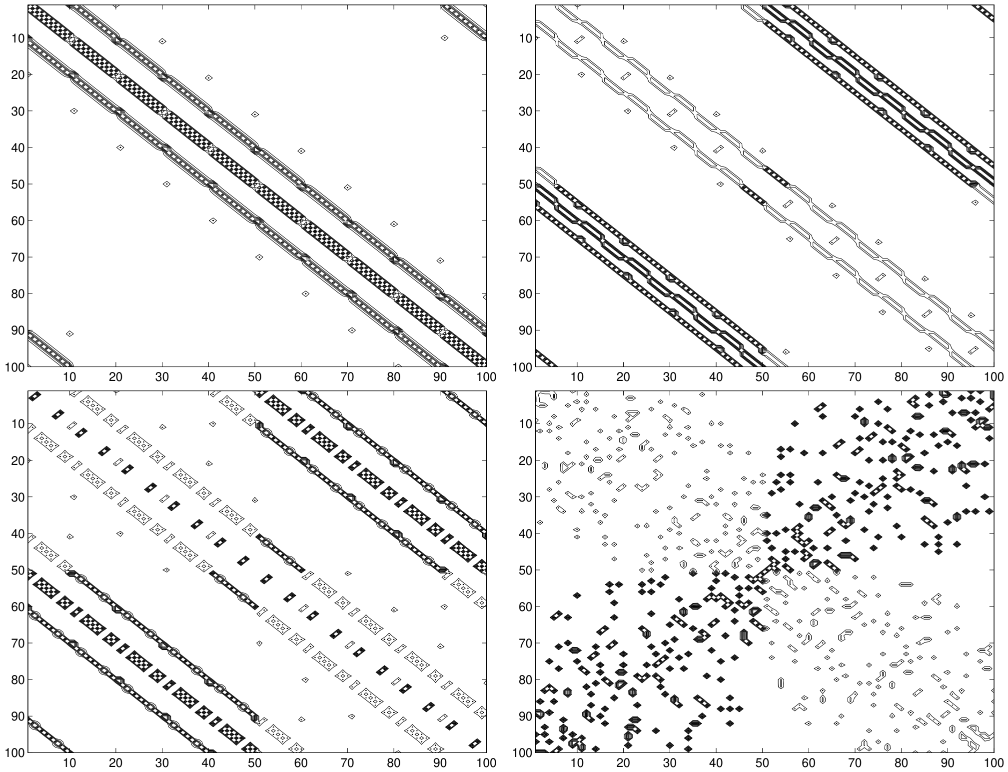

Figure 8: The structure of the original matrix G_2 from Ericsson in the top row on the left, and on the right the reordered using the red black ordering. In the bottom row on the left the best solution found by Strategy1 and on the right the solution found by the spectral method.

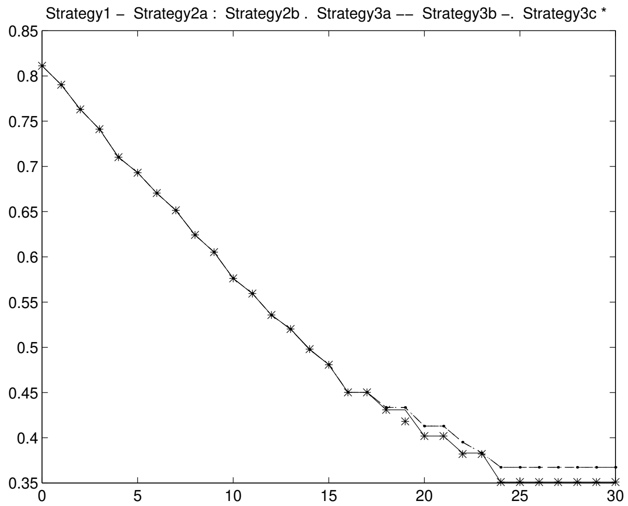

Figure 9: One experiment with the matrix G_2 from Ericsson

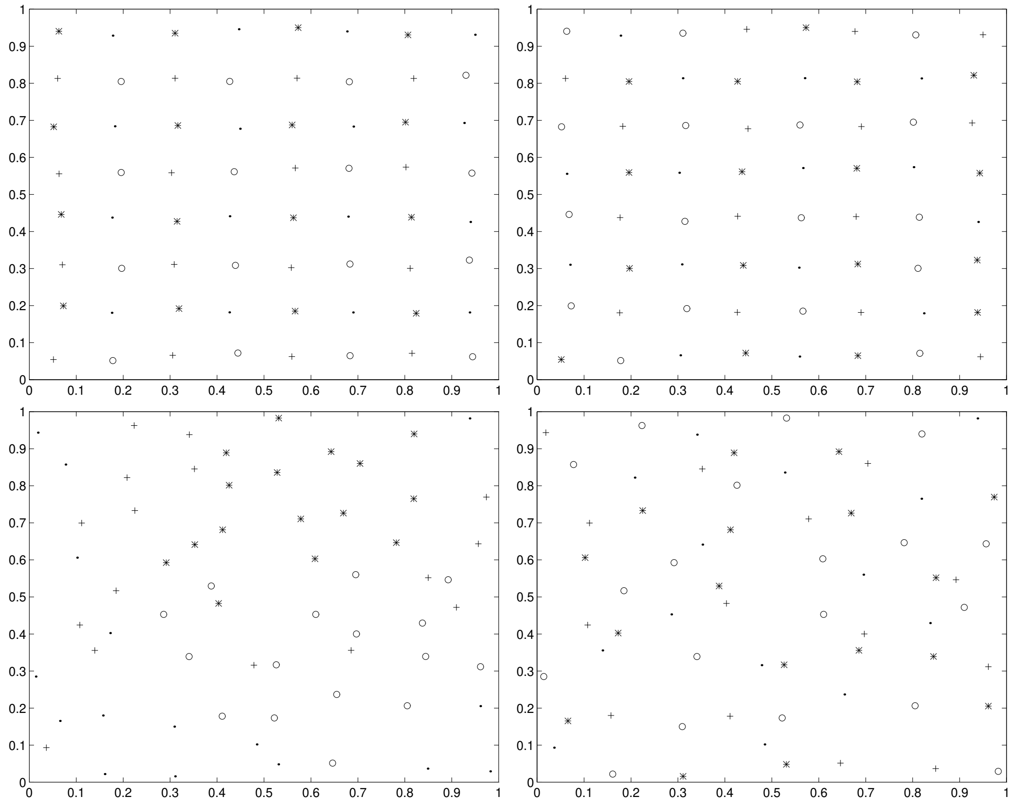

Figure 10: Two examples with four frequencies, on top well ordered with parameter $\delta = \frac{1}{5}$ and at the bottom less ordered with $\delta = \frac{4}{5}$. On the left the spectral algorithm was applied and on the right Strategy1

64 base stations and 4 frequencies.

In the top row the base stations are very regularly distributed. On the left the spectral algorithm was applied recursively. It found the corresponding four color ordering, and the largest eigenvalue of all the diagonal matrices was reduced from 2.4189 in the initial matrix to 0.3422, a factor of 7. On the right Strategy1 was applied. It finds a close by solution to the four color ordering, reducing the largest eigenvalue to 0.4899, a factor of 5.

In the bottom row the base stations are less regularly distributed, with the parameter $\delta = \frac{4}{5}$. On the left the spectral algorithm does not find a good ordering, the largest eigenvalue of all the diagonal matrices is reduced from 1.0440 in the initial matrix to 0.2761, by a factor 4. On the right Strategy1 was applied which works much better; It reduces the largest eigenvalue to 0.0428, by a factor 24.

Table 1 shows the average reduction factors of the methods for ten

Algorithm	δ		
	$\frac{1}{5}$	$\frac{1}{2}$	$\frac{4}{5}$
Strategy1	8.83	16.10	56.00
Strategy2a	7.08	12.58	39.23
Strategy2b	9.79	17.88	55.92
Strategy3a	7.08	12.58	39.23
Strategy3b	7.08	12.58	39.23
Strategy3c	9.96	15.76	55.15
Spectral	14.96	8.00	8.31

Table 1: Improvement factor over initial configuration using four frequencies

example problems $G \in \mathbb{R}^{64\times 64}$ with four frequencies and with different degree of order in the base stations. We used 60 iterations for the iterative methods. It is evident from table 1 that for regular distributions of base stations the spectral algorithm should be used, whereas for irregular distributions of base stations, the other algorithms are far superior. Note that the deterministic algorithm Strategy1 performs as well as the randomized ones in the irregular case.

6 Conclusion

We presented a set of algorithms to solve the frequency assignment problem approximately by minimizing matrix norms. We tested the algorithms extensively on random matrices, matrices derived from random base station locations and test matrices from Ericsson.

The experiments show that minimizing the one and/or infinity norm provides good approximate solutions to the frequency assignment problem for all types of matrices. For random matrices, the randomized algorithms perform better on average than the deterministic ones, whereas for matrices derived from random base station locations the deterministic ones perform as well as the randomized ones.

Minimizing the Frobenius norm leads to excellent solutions in the case where the base stations are regularly distributed. If this is not the case, then minimizing the Frobenius norm does not provide useful solutions.

References

[1] Bengt Aspvall and John R. Gilbert, *Graph Coloring Using Eigenvalue Decomposition*, SIAM J. ALG. DISC. METH. Vol 5, No. 4,

December 1984.

[2] Pierre Baldi and Edward Posner, *Graph Coloring Bounds for Cellular Radio*, Computers Math. Applic. Vol. 19, No. 10, pp 91-97, 1990.

[3] A. Gamst and W. Rave, *On the Frequency Assignment in Mobile Automatic Telephone Systems*, in Proc. of GLOBECOM 82, pp. 309-315, 1982.

[4] Jin-Kao Hao and Raphaël Dorne, *Study of Genetic Search for the Frequency Assignment Problem*, Artificial Evolution European Conference, Springer-Verlag, pp. 333-44, 1996.

[5] A. Knällmann and A. Quellmalz, *Solving the Frequency Assignment Problem with Simulated Annealing*, Electromagnetic Compatibility, Conference Publication No 396, IEE.

[6] A. Quellmalz, A. Knällmann, B. Müller, *Efficient Frequency Assignment with Simulated Annealing*, Antennas and Propagation, Conference Publication No 407, IEE 1995.

[7] Teahoon Park and Chae Y. Lee, *Application of the Graph Coloring Algorithm to the Frequency Assignment Problem*, Journal of the Operations Research Society of Japan, Vol. 39, No. 2, June 1996.

[8] H. D. Simon, *Partitioning of Unstructured Problems for Parallel Processing*, Computing Systems in Engineering, vol 2, no 2/3, pp 135-148, 1991.

[9] J. Zander, *Performance of Optimum Transmitter Power Control in Cellular Radio Systems*, IEEE Transactions on Vehicular Technology, Vol. 41, No. 1, February 1992.