A tabu search algorithm with a dynamic tabu list for the frequency assignment problem

R. Montemanni and D.H. Smith

Division of Mathematics and Statistics
School of Technology
University of Glamorgan
Pontypridd, CF37 1DL, Wales, U.K.

29 October 2001

Abstract

The frequency assignment problem involves the assignment of discrete channels (or frequencies) to the transmitters of a radio network, such as a mobile telephone network. Frequency separation is necessary to avoid interference by other transmitters to the signal received from the wanted transmitter at the reception points. Unnecessary separation causes an excess requirement for spectrum. Good assignments minimise interference and the spectrum required.

In this paper we study the fixed spectrum frequency assignment problem, where the frequency spectrum available is given and the target is to minimise (a measure of) the interference in the network. In particular a tabu search algorithm with some non-standard features, together with implementation details, is described for this problem.

Computational results confirm the quality of the algorithm we propose, in comparison with methods of other authors.

1 Introduction

The frequency assignment problem occurs when a wireless communication network has to be established. Wireless communication networks have become an important reality, especially in the last decade, during which there has been a large scale development of cellular telephone networks. As the available radio spectrum is a limited resource, the growth of demand has increased the importance of good network planning, aiming to use as few frequencies as possible. On the other hand, using fewer frequencies in a network generally leads to higher interference and thus a lower service quality.

The aim of frequency assignment problems is to balance the economies achieved by frequency reuse with the consequent loss of quality in the network. In this paper we study the fixed spectrum frequency assignment problem, which is a combinatorial problem arising when a network has to be established and there is a given amount of spectrum available, and whose target is to minimise (a measure of) the interference of the network. For this problem we present an adaptation, with particular features not present in the general scheme, of a classic metaheuristic algorithm.

Many heuristic algorithms have been presented in the literature for the fixed spectrum frequency assignment problem, or for slightly more complex problems which generalise it. In this section we briefly describe some of these approaches, without attempting to be comprehensive. More detailed reviews can be found in Koster [16], Smith et al. [22], Eisenblätter et al. [8] and, for the algorithms developed within the EUCLID CALMA project, in Hurkens and Tiourine [13], Tiourine et al. [24] and Aardal et al. [1].

Borndörfer et al. [3], [4] and Eisenblätter [9] present some *constructive algorithms* (i.e. algorithms which construct a solution step by step starting from an empty one) together with some basic *local searches* (i.e. algorithms which move from solution to solution, searching for the best possible one). Other notable local search approaches are presented in Koster [16] and Montemanni [20], where a set of constructive algorithms is also proposed. An interior point method applied to a quadratic formulation of *FS-FAP* is described in Warners [28] and in Warners et al. [29].

In the field of *metaheuristic algorithms* (i.e. iterative methods which, using particular strategies, drive a subordinate heuristic algorithm to explore the search space in an intelligent way), an Approximate Non deterministic Tree Search (ANTS) algorithm is presented in Maniezzo and Carbonaro [18], Maniezzo et al. [19] and Montemanni [20]. Some methods based on the genetic algorithm paradigm are proposed in Kolen [15] and in Lau and Tsang [17]. In [5], Boyce et al. present a GEneral NETwork (GENET) algorithm together with a tabu search algorithm. A Guided Local Search (GLS) algorithm is presented in Voudouris [26] and Voudouris and Tsang [27].

A complete collection of heuristic methods is provided by FASoft (Hurley et al. [14]), where assorted constructive algorithms, local searches and metaheuristic algorithms are implemented.

Finally, in Whitaker et al. [31] a tabu search algorithm developed to deal with binary and non binary constraints (i.e. constraints involving more than two transmitters at a time), called NBS (non binary solver), is described.

The paper is organised as follows. In Section 2 the fixed spectrum frequency assignment problem is formally defined. In Section 3 the general schema of the tabu search algorithm and its adaptation to the fixed spectrum frequency assignment problem are described. Some implementation details are also given. In Section 4 some computational results are proposed. Finally, in Section 5 our conclusions are presented.

2 The fixed spectrum frequency assignment problem

The fixed spectrum frequency assignment problem can be represented through a weighted undirected graph. Formally it is a 5-tuple $FS\text{-}FAP = \{V, E, D, P, F\}$ with:

- V: vertex set of an undirected graph G. Every vertex represents a transmitter of the original frequency assignment problem;

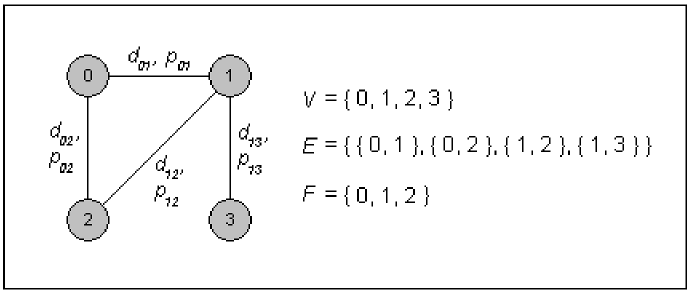

Figure 1: Fixed spectrum frequency assignment problem. Example.

- E: set of edges of the undirected graph G. Edges represent those transmitters that are constrained, i.e. pairs of potentially interfering transmitters;
- D: set of labels for which $\forall\{v, w\} \in E \;\; \exists\, d_{vw} \in \mathcal{N}_0^+$. d_{vw} is the highest separation between the frequency assigned to the transmitter v and the one assigned to w that generate unacceptable interference. If we indicate with $f(v)$ the frequency assigned to transmitter v, then if $|f(v) - f(w)| > d_{vw}$, the interference involving the two transmitters is acceptable;
- P: set of labels for which $\forall\{v, w\} \in E \;\; \exists\, p_{vw} \in \mathcal{N}^+$. p_{vw} is a cost to be paid if the separation between the frequencies of transmitters v and w is less than or equal to d_{vw};
- F: set of consecutive frequencies available for every vertex (transmitter) in V (assumed the same $\forall v \in V$).

The objective of the fixed spectrum frequency assignment problem is to find an assignment which minimises the sum of p_{vw} over all pairs $\{v, w\} \in E$ for which $|f(v) - f(w)| \leq d_{vw}$.

An example of a graph associated with a problem is given in Figure 1.

3 The tabu search algorithm

In this section we describe the general ideas of the tabu search algorithm and an implementation of it we have developed for the *FS-FAP*. This implementation presents some particular features, which are not present in the general schema. These features seem to improve the performances of the tabu search algorithm on the *FS-FAP*s.

3.1 General description

Tabu search is a metaheuristic algorithm. It was first suggested in Glover [10] (see also Glover et al. [11]).

The basic idea of the method is to partially explore the search space of all feasible solutions by a sequence of moves. At each iteration, the move carried out is the most promising among those available. A mechanism which forbids a set of moves at each iteration is present, aiming to help the algorithm to escape from local (but not global) minima.

Formally, the main elements of the algorithm are:

- Solution representation: each feasible solution of the optimisation problem must have an unique representation within the search space;
- Cost function: a function *Cost* mapping each feasible solution into a value representing its optimisation cost. The goal of the algorithm is to find a solution which minimises this value;
- Neighbourhood: a function mapping each feasible solution S into a set of other solutions. Each time the algorithm has to consider a new solution, it is chosen from the neighbourhood of the current solution;
- Tabu list: a list containing the last T moves carried out, which for this reason are forbidden. A solution obtained from the current solution S with a move contained in the tabu list, cannot (in general) be a member of the neighbourhood of S;
- Aspiration criterion: if a tabu move (a move which is contained in the tabu list) satisfies this criterion, then the solution obtained by applying it to the current solution S can be considered to be in the neighbourhood of S. The usual criterion is that the move produces the best solution obtained so far.
- Termination criterion: the algorithm stops when the termination criterion is satisfied.

At each iteration the algorithm calculates the neighbourhood of the current assignment. Solutions generated by using a move contained in the tabu list cannot be in the neighbourhood set, unless the respective move satisfies the aspiration criterion. The solution with the minimum cost among those in the neighbourhood becomes the new current solution.

The tabu list, whose dimension strictly depends on the neighbourhood selected, has been inserted to prevent the search becoming trapped in a local minimum, while the aspiration criterion has been introduced to give more flexibility to the algorithm: it makes a move contained in the tabu list feasible in case it would produce a very promising new solution.

In Figure 2 the pseudocode of the general tabu search algorithm is presented.

3.2 A tabu search algorithm for the *FS-FAP*

In this section we present the elements of an adaptation, characterised by non-standard features, of the tabu search algorithm for the *FS-FAP*.

3.2.1 Solution representation

The representation of a frequency assignment S is obtained by using a list $\{f_S(0), f_S(1), \ldots, f_S(|V|-1)\}$ where the v^{th} element ($f_S(v)$) contains the frequency assigned to transmitter v.

TabuSearch(*Pr*) // *Pr* is an optimisation problem

S := randomly generated solution of Pr;
$Best$:= S;
While(termination criterion not met)
    S := best solution in the neighbourhood of S^*;
    If($Cost(S) < Cost(Best)$)
        $Best$:= S;
    EndIf
    update tabu list;
EndWhile
Return $Best$;

*the neighbourhood of S does not include solutions obtained using those moves which are contained in the tabu list and do not satisfy the aspiration criterion.

Figure 2: Tabu search algorithm.

3.2.2 Cost function

The cost function *Cost* maps an assignment into the sum of the penalties paid in it. Formally we have:

$$Cost(S) = \sum_{\substack{\{v,w\} \in E; \\ |f_S(v) - f_S(w)| \leq d_{vw}}} p_{vw} \quad (1)$$

3.2.3 Neighbourhood

An assignment S_N is in the neighbourhood of the current solution S_O if S_N differs from S_O in the frequency assigned to exactly one violating transmitter[1] and the move which produces S_N from S_O is not in the tabu list (no aspiration criterion is used, see Section 3.2.5). Defining $V^V_{S_O}$ as the set of violating transmitters in the assignment S_O, S_N is a neighbour of S_O if $\exists\, v \in V^V_{S_O} \mid f_{S_O}(v) \neq f_{S_N}(v)$ and $f_{S_O}(w) = f_{S_N}(w)$ $\forall w \in V, w \neq v$ and the move $(v, f_{S_N}(v))$ is not in the tabu list.

3.2.4 Tabu list

The tabu list of our algorithm contains pairs $(v,\ f)$, where v is a transmitter and f a frequency. Each time a move involving the assignment of frequency f to transmitter v is carried out, we insert (v,f) into the tabu list, where it will remain for approximately T iterations.

Instead of using a tabu list with a fixed length T, as in the original schema, we have decided to dynamically vary T during the running of the algorithm. This choice has been suggested by some tests which indicated the superiority of

[1] A violating transmitter is defined as a transmitter involved in at least one constraint violated in S_O.

the dynamic tabu list over the static one (see Section 4.3). In particular we have noticed that the best results were achieved by reducing the length of the tabu list in the same way as it is done for the temperature parameter in a simulated annealing algorithm. Every I_{ts} iterations we reduce the length T of the tabu list using the following assignment:

$$T := \beta T \tag{2}$$

where $0 < \beta < 1$ is a user defined parameter. When T is reduced, the oldest moves which exceed the new length of the list become feasible. The initial value of T, which we will refer to as T_{init}, is defined by the user.

3.2.5 Aspiration criterion

We do not use any aspiration criterion in our tabu search algorithm. Some preliminary tests suggested that the use of an aspiration criterion slows down our implementation of the algorithm (because of the extra data structures and their updating) without improving the results. This situation is quite uncommon in the field of combinatorial optimisation, and could be an indication of the peculiarity of the *FS-FAP*.

3.2.6 Termination criterion

The algorithm stops when T, the length of the tabu list, becomes smaller then a threshold value $T_{\min}$, which is specified by the user. This termination criterion is quite uncommon for a tabu search algorithm, but here its use is indicated by our strategy of dynamically modifying the parameter T.

3.2.7 Implementation details

The implementation technique we adopt is inspired by the one described in Hao et al. [12]. We maintain a table of dimension $|V||F|$, called the *cost change table*, where position (v, f) contains the cost of the solution obtained by changing to f the frequency currently assigned to transmitter v (if f is the frequency currently assigned to v, then the value contained in the entry (v, f) is the cost of the current solution). Each time a move is carried out, the elements of the table affected by the move are updated accordingly. For each transmitter v we also adopt a list containing its adjacent transmitters (transmitters involved in at least one constraint with v). This list is used to speed up the table updating process. Only the transmitters which are adjacent to the one modified are involved in the updating process. Only the positions of the table corresponding to the frequencies which interfere with the old or the new frequency of the reassigned transmitter are modified. An example of the use of the cost change table is given in Figure 3. An interval of columns of the rows corresponding to four transmitters, u, v, w and z, appears in the figure. In the graph representing the problem, u and v are not connected, u and w are connected by an edge with separation $d_{uw} = 2$ and u and z are connected by an edge with separation $d_{uz} = 0$. In the figure we depict the effects which derive from the modification of the frequency assigned to transmitter u from j (Old) to i (New). In the figure we have highlighted the table entries the values of which are modified because of the frequency reassignment. In blue we have

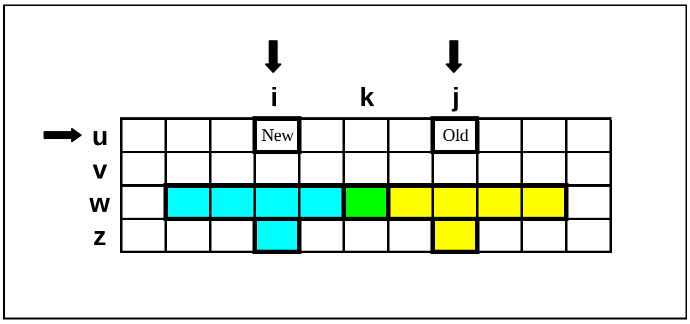

Figure 3: Updating the cost change table.

indicated the entries which increase their values (of p_{uw} and p_{uz} for w and z respectively) and in yellow the entries which have their values decreased. Notice that the entry (w, k) (green) is not modified. As transmitter v is not connected with transmitter u, its row is not affected by the frequency reassignment.

There is also a second table of dimension $|V|\,|F|$ which contains in each position (v, f) the last iteration number in which the respective move (assignment of frequency f to transmitter v) has been carried out. This table is used to check in a fast way whether or not a candidate move is tabu: if the iteration number stored in the cell is greater than or equal to the current iteration number minus T, then the move is tabu. We also maintain a list which indicates for each transmitter, the frequency (different from the one currently assigned) that, without generating a tabu move, would produce the lowest cost if assigned to it. This list is used at each iteration to select quickly the best move in the active neighbourhood. After each iteration this list is updated efficiently, by modifying only the entries affected by the last move. We also have a one-dimensional array of length $|T_{init}|$ which contains the conventional tabu list. It is used at each iteration to identify quickly the move which exits from the list (it is used to update efficiently other data structures).

Initialising the structures has a computational complexity of $O(|V|^2|F|)$. After each move, the tables can be updated with a theoretical complexity of $O(|V||F|)$. In practise the updating process is extremely fast.

4 Computational results

In this section, after a brief description of the benchmarks adopted, we present the results obtained by the algorithm described in Section 3.2. In particular a study on the effectiveness of the dynamic length tabu list within the tabu search algorithm is presented. Finally, a comparison of our tabu search algorithm with methods proposed by other authors is proposed. All the tests of this thesis have been carried out on a computer with an Intel Pentium II 400MHz processor, equipped with 128MB of memory.

4.1 Description of the benchmarks adopted

In this section we describe the characteristics of the graphs on which the benchmarks we adopt are based. It is interesting to notice that we will often undertake different tests on the same graph, changing the size of the spectrum available. For this reason there are more benchmarks than the number of graphs we describe in the following.

We have three different sets of graphs.

The graphs of the first set were originally minimum span problems. We have converted them into the *FS-FAP* format by fixing $p_{vw} = 1 \ \forall \{v, w\}$.

Specifically, the scenarios of this first set are created from different families of minimum span problems:

- AC-x-y: scenarios derived from a binary constraint representation of area coverage problems (see Watkins et al. [30]). x is the number of transmitters and y the required *SIR* (signal to interference ratio);
- GSM-x: realistic *GSM* scenarios. x is the number of transmitters in the network;
- Test$x$: graphs generated by Cardiff University (see Castellino et al. [6] and Smith et al. [23]). Again x is the number of transmitters in the network;
- P06-z: subproblems of the well-known *Philadelphia* problem, originally proposed in Anderson [2] (see also Smith et al. [23]). The generic graph P06-z is obtained by considering for every cell i of the problem a demand of $\left\lceil \frac{m(i)}{z} \right\rceil$, where $m(i)$ is the original demand for cell i.
- P06b-z: graphs obtained from the *Philadelphia* problem with the same method described for *P06-z*, but with a co-cell separation of 3 instead of the original 5. This has been done to more closely match the characteristics of realistic modern frequency assignment problems.

The second family of scenarios is formed by only one type of graph:

- GSM2-x: adaptation to our model of realistic *GSM* scenarios. x is the number of transmitters in the network.

The third family is composed of random scenarios we have generated using a basic graph generator. Our generator randomly places some sites into a rectangular region and assigns to each site a random number of transmitters. Given a required edge density for the graph, we fix the Euclidean distance ϵ below which there will be a constraint (edge). The highest separation δ and highest penalty ϕ which will appear in the problem are specified by the user. These values will be used for co-sited constraints. For each non co-site constraint, if we call σ the Euclidean distance between the two transmitters v and w involved, the required separation d_{vw} is a random integer in the following interval:

$$0 \leq d_{vw} \leq \left\lceil \frac{(\epsilon - \sigma)\delta}{\epsilon} \right\rceil \tag{3}$$

Each p_{vw} is generated in a similar way, and it is a random integer in the following interval:

$$1 \leq p_{vw} \leq \left\lceil \frac{(\epsilon - \sigma + 1)\phi}{\epsilon + 1} \right\rceil \tag{4}$$

We do not expect the scenarios created using our generator to be very realistic because of the very basic model of reality adopted, but we think they are adequate for our purpose. For the description of better, and more complex, frequency assignment problems generators, we refer the interested reader to Dunkin and Allen [7] and van Benthem [25].

Specifically, the third set of scenarios is composed of the following type of graph:

- $r1$-$r2$-s-x-w-δ-ϕ: *r1* is a random seed used to place sites on the rectangular area; *r2* is a second random seed adopted to calculate separations and penalties; s is the number of sites of the network; x is the number of transmitters to distribute among the s sites; w ($0 < w \leq 1$) is an approximation of the edge density of the graph (number of edges of the graph $\approx \left\lfloor \frac{wt(t-1)}{2} \right\rfloor$); δ is the maximum separation value in the scenario; ϕ is the maximum penalty value in the scenario.

4.2 Parameter settings

The algorithms are naturally fast because of the efficient implementation, so we had the opportunity to choose quite conservative values for the parameters, which give priority to a careful search instead of the convergence speed. Execution times are anyway under 45 minutes for all of the problems, and under 10 minutes for most of them.

In detail we have the following settings. For all of the problems we have fixed $T_{\min} = 10$, $\beta = 0.96$ and $I_{ts} = 5 * 10^4$. $T_{\text{init}} = 500$ for the problems based on *AC-45-17*, *AC-45-25*, *AC-95-9* and on the graphs of the third set; $T_{\text{init}} = 1000$ for the problems based on *AC-95-17*, *GSM-93*, *GSM-246*, *Test95* and on the graphs of the second set. Finally $T_{\text{init}} = 2000$ for the remaining problems (based on *Test282*, *P06-5*, *P06-3*, *P06b-5* and *P06b-3*).

4.3 Effectiveness of the dynamic length tabu list in the tabu search algorithm

In this section we compare the results obtained by our tabu search algorithm (which incorporates a dynamic length tabu list) with the results obtained by a tabu search algorithm with a fixed length tabu list (i.e. $\beta = 1$), which we will refer to as the *conventional tabu search algorithm*.

The implementation of the conventional algorithm is the same as described in Section 3.2.7 except for the exit criterion, which in this case is a maximum computation time of 45 minutes. For each problem considered, this time is longer than the time require by the tabu search algorithm with the dynamic length tabu list, which in the remainder of the paper we will refer to as the *dynamic tabu search algorithm*.

In Table 1 the results achieved by the conventional tabu search are compared with those obtained by the dynamic tabu search. Three different values for the length of the conventional tabu list have been considered for each problem, and for each one of these values the best result achieved in five runs is presented in the table. The columns of the table have the following meaning:

- Problem: names of the problems. Each name is composed of the following two elements:

  - Graph: name of the graph on which the problem is based;
  - $|F|$: number of channels available;

- DTS: best results obtained by the dynamic tabu search algorithm;

- CTS: best results obtained by the conventional tabu search algorithm. Subcolumns have the following meaning:

  - T_i: length of the tabu list;
  - Val$_i$: best upper bound obtained with a tabu list of length T_i.

The advantage arising from the use of the dynamic length tabu list is clear from Table 1. The dynamic tabu search algorithm obtains a worse result than the conventional tabu search algorithm in only two cases (problem *GSM2-184* with $|F| = 49$ and problem *1-1-50-75-30-2-50* with $|F| = 15$).

It is also important to observe that the parameter tuning of the dynamic tabu search algorithm is easier because even a greatly overestimated choice for parameters T_{init}, β and I_{ts} does not compromise the quality of the estimates (although the convergence speed of the algorithm may be affected). On the contrary, the choice of parameter T is crucial for the conventional tabu search algorithm.

4.4 Comparison with algorithms of other authors

In this section we compare our tabu search algorithm with two programs developed by other authors.

The first program considered is FASoft (Hurley et al. [14]). It treats only problems where the target is to minimise the number of constraint violations, so we cannot run it on the second and, except for one problem, the third families of benchmarks. As stated in Section 1, FASoft contains more than one algorithm. For the tests reported in this thesis we have adopted the tabu search algorithm, which seems to be the best one of the collection.

The second algorithm we consider is the tabu search algorithm NBS, developed by Whitaker et al. [31]. The algorithm has been originally developed to deal with binary and non binary constraints, but we do not have this last type of constraint. Thus we do not use all of its functionality. Unfortunately the method is not able to manage the problems of the second benchmarks family because their penalties are too high. Consequently we have run NBS only on the problems of the first and the third families.

The methods have been tuned in such a way as to have computation times similar to those of the algorithms developed by us. Notwithstanding our efforts in the tuning phase, we believe that the developers of the algorithms could have found better parameter configurations, which probably would have produced better results.

In Table 2 we compare the upper bounds produced by our algorithms with those provided by FASoft and NBS. The results presented are, for each method, the best obtained in five or more runs. The symbol " - ", which appears in some entries, means that no result is available (method cannot manage this type of problem). The columns of the tables have the following meaning:

Table 1: Effectiveness of the dynamic length tabu list.

Problem Graph	$\|F\|$	DTS	CTS T_1	Val_1	T_2	Val_2	T_3	Val_3
AC-45-17	7	32	450	36	100	32	30	32
AC-45-17	9	15	450	20	100	15	30	15
AC-45-25	11	33	450	33	140	33	30	34
AC-45-25	19	8	450	8	140	8	30	8
AC-95-9	6	31	450	36	140	31	30	31
AC-95-9	10	3	450	3	140	3	30	3
AC-95-17	15	33	450	40	100	34	30	34
AC-95-17	21	10	450	13	100	10	30	10
GSM-93	9	32	900	44	100	33	50	33
GSM-93	13	7	900	13	100	7	50	7
GSM-246	21	79	900	94	500	87	100	100
GSM-246	31	25	900	36	500	33	100	34
Test95	31	12	900	15	300	12	100	12
Test95	36	8	900	10	300	8	100	8
Test282	61	51	1800	83	400	63	100	68
Test282	71	27	1800	52	400	36	100	40
P06-5	11	133	1800	161	400	133	100	161
P06-5	41	15	1800	15	400	15	100	21
P06-3	31	115	1800	121	400	123	100	144
P06-3	71	26	1800	26	400	29	100	31
P06b-5	21	52	1800	52	400	52	100	61
P06b-5	31	25	1800	25	400	25	100	27
P06b-3	31	112	1800	117	300	117	100	117
P06b-3	71	26	1800	26	300	26	100	26
GSM2-184	39	5521	900	6806	300	5881	100	5896
GSM2-184	49	999	900	1054	300	874	100	999
GSM2-227	39	10979	900	13325	500	11561	100	12996
GSM2-227	49	2459	900	3322	300	2517	100	2961
GSM2-272	39	27416	900	30775	500	29481	100	39506
GSM2-272	49	7785	900	8877	500	8776	100	10165
1-1-50-75-30-2-50	5	1242	450	1347	140	1255	30	1352
1-1-50-75-30-2-50	10	101	450	163	140	105	30	165
1-1-50-75-30-2-50	11	68	450	101	140	73	30	102
1-1-50-75-30-2-50	15	12	450	18	140	11	30	15
1-2-50-75-30-4-50	11	323	450	444	140	327	30	441
1-3-50-75-30-0-50	11	36	450	48	140	39	30	44
1-4-50-75-30-2-1	10	19	450	24	140	20	30	20
1-5-50-75-30-2-100	10	186	450	336	140	198	30	349
1-6-50-75-30-0-10^3	10	6942	450	8982	140	7176	30	9100

- Problem: names of the problems. Each name is composed of the following two elements:
  - Graph: name of the graph on which the problem is based;
  - $|F|$: number of channels available;
- Best: best estimates available for the problems (see Montemanni [21]);
- Tabu search: best results obtained by the tabu search algorithm described in Section 3.2;
- FASoft: best results obtained by the tabu search algorithm contained in the system FASoft;
- NBS: best results obtained by the tabu search algorithm NBS.

On the benchmarks analysed in Table 2, both FASoft and NBS are clearly dominated by our algorithm. On some problems (in particular on those of the third benchmarks set) there is a great difference among the upper bounds provided by the different algorithms. This may be seen as an indication of the difficulty of these problems.

We believe that the better performance of our method can be explained partly by a better implementation. The two tabu search algorithms of FASoft and NBS are implemented in a classic way, and do not use the special structures adopted in the implementation of our tabu search algorithm (see Section 3.2.7). A consequence of the adoption of these special structures is that our tabu search is able to use a full neighbourhood, which we believe to be a really important factor in the quality of our results. FASoft and NBS have to use a random (partial) neighbourhood for computational reason.

Considering the pure speed of the algorithms, we can observe that, notwithstanding that we consider a full neighbourhood instead of a partial one as the other methods, our tabu search is, in terms of number of iterations carried out in a given interval of time, at least 5 times faster than the tabu search of FASoft and at least 20 times faster than the one developed within NBS. It is anyway important to remind the reader that NBS is the only method, among those compared, which is able to deal with non binary constraints. This extra feature contributes to make the algorithm slower.

To understand the role of the speed of our methods in the better results obtained by them, we tested FASoft and NBS on longer runs. Practically no improvement was found, and this suggests that the superiority of our method does not depend only on its speed, but probably on the different neighbourhood adopted. However it must be observed that the use of a better neighbourhood in our methods is a direct consequence of the use of an efficient implementation.

Another factor which could make our tabu search algorithm more effective than those implemented in FASoft and NBS is, we believe, the dynamic length tabu list (as observed in Section 4.3).

It is also important to notice that the dynamic tabu search algorithm matches the best known results for all of the problems except *GSM2-184* with $|F| = 49$ and *1-1-50-75-30-2-50* with $|F| = 15$. In Table 1 these two problems were the only for which the conventional algorithm outperformed the dynamic one. This may be an indication that, for these problems, the five runs considered for the dynamic tabu search algorithm were particularly unlucky.

Table 2: Computational results.

Problem Graph	$\|F\|$	Best	Tabu search	FASOFT	NBS
AC-45-17	7	32	32	33	32
AC-45-17	9	15	15	16	16
AC-45-25	11	33	33	34	33
AC-45-25	19	8	8	9	8
AC-95-9	6	31	31	31	31
AC-95-9	10	3	3	3	3
AC-95-17	15	33	33	36	36
AC-95-17	21	10	10	11	10
GSM-93	9	32	32	34	39
GSM-93	13	7	7	9	9
GSM-246	21	79	79	89	93
GSM-246	31	25	25	35	36
Test95	31	12	12	15	13
Test95	36	8	8	10	9
Test282	61	51	51	75	70
Test282	71	133	133	137	144
P06-5	41	15	15	16	15
P06-3	31	115	115	123	132
P06-3	71	26	26	28	29
P06b-5	21	52	52	52	53
P06b-5	31	25	25	25	25
P06b-3	31	112	112	113	116
P06b-3	71	26	26	26	26
GSM2-184	39	5521	5521	-	-
GSM2-184	49	874	999	-	-
GSM2-227	39	10979	10979	-	-
GSM2-227	49	2459	2459	-	-
GSM2-272	39	27416	27416	-	-
GSM2-272	49	7785	7785	-	-
1-1-50-75-30-2-50	5	1242	1242	-	1296
1-1-50-75-30-2-50	10	101	101	-	131
1-1-50-75-30-2-50	11	59	68	-	112
1-1-50-75-30-2-50	15	11	12	-	17
1-2-50-75-30-4-50	11	323	323	-	371
1-3-50-75-30-0-50	11	36	36	-	44
1-4-50-75-30-2-1	10	19	19	19	21
1-5-50-75-30-2-100	10	186	186	-	251
1-6-50-75-30-0-10^3	10	6942	6942	-	18034

5 Conclusion

A tabu search algorithm with some non-standard features (i.e. a dynamic length tabu list) has been developed and implemented for the fixed spectrum frequency assignment problem.

The benefit arising from the adoption of the dynamic length tabu list has been studied. Some computational tests, in which our tabu search algorithm outperforms two methods proposed by other authors, have been presented.

References

[1] K.I. Aardal, C.A.J. Hurkens, J.K. Lenstra, and S.R. Tiourine. Algorithms for the radio link frequency assignment problem. Technical Report UU-CS-1999-36, Utrecht University, November 1999.

[2] L.G. Anderson. A simulation study of some dynamic channel assignment algorithms in a high capacity mobile telecommunications system. *IEEE Transactions on Communications*, 21:1294–1301, 1973.

[3] R. Borndörfer, A. Einsenblätter, M. Grötschel, and A. Martin. Stable set and other techniques for frequency assignment problems. Preprint, Konrad-Zuse-Zentrum für Informationstechnik Berlin.

[4] R. Borndörfer, A. Einsenblätter, M. Grötschel, and A. Martin. Frequency assignment in cellular phone networks. *Annals of Operations Research*, 76:73–93, 1998.

[5] J.F. Boyce, C.H.D. Dimitropoulos, G. vom Scheidt, and J.G. Taylor. GENET and tabu search for combinatorial optimization problems. In *WC-NN'95, World Congress on Neural Networks*, 1995.

[6] D.J. Castellino, S. Hurley, and N.M. Stephens. A tabu search algorithm for frequency assignment. *Annals of Operations Research*, 63:301–319, 1996.

[7] N. Dunkin and S.M. Allen. Frequency assignment problems: Representations and solutions. Technical Report CSD-TR-97-14, Royal Holloway, University of London, 1997.

[8] A. Einsenblätter, M. Grötschel, and A.M.C.A. Koster. Frequency assignment and ramifications of coloring. Technical Report ZIB-Report 00-47, Konrad-Zuse-Zentrum für Informationstechnik Berlin, December 2000.

[9] A. Eisenblätter. A frequency assignment problem in cellular phone networks (extended abstract). Technical Report SC 97-27, Konrad-Zuse-Zentrum für Informationstechnik Berlin, July 1997.

[10] F. Glover. Heuristics for integer programming using surrogate constraints. *Decision Sci.*, 8:156–166, 1977.

[11] F. Glover, E. Taillard, and D. de Werra. A user's guide to tabu search. *Annals of operations research*, 41:3–28, 1993.

[12] J.-K. Hao, R. Dorne, and P. Galinier. Tabu search for frequency assignment in mobile radio networks. *Journal of Heuristics*, 4:47–62, 1998.

[13] C.A.J. Hurkens and S.R. Tiourine. Upper and lower bounding techniques for frequency assignment problems. Technical Report COSOR 95-34, Eindhoven University of Technology, 1995.

[14] S. Hurley, D.H. Smith, and S.U. Thiel. FASoft: A system for discrete channel frequency assignment. *Radio Science*, 32(5):1921–1939, 1997.

[15] A.W.J. Kolen. A genetic algorithm for frequency assignment. Technical report, Maastricht University, 1999.

[16] A.M.C.A. Koster. *Frequency Assignment. Models and Algorithms.* PhD thesis, Maastricht University, November 1999.

[17] T.L. Lau and E.P.K. Tsang. Solving the radio link frequency assignment problem with the guided genetic algorithm. In *Proceedings of the NATO RTA SET/ISET Symposium on Frequency Assignment, Sharing and Conservation in Systems (Aerospace),* Aalborg, Denmark, October 1998, NATO RTO-MP13, pages 14b.1–14b.27, 1999.

[18] V. Maniezzo and A. Carbonaro. An ANTS heuristic for the frequency assignment problem. *Future Generation Computer Systems*, 16:927–935, 2000.

[19] V. Maniezzo, A. Carbonaro, and R. Montemanni. An approach to frequency assignment problem based on an ANTS heuristic. In *Proceedings of the Third Metaheuristic International Conference (MIC'99)*, pages 311–316, 1999.

[20] R. Montemanni. *Algoritmi Esatti ed Euristici per il Problema dell'Assegnamento di Frequenze*. Tesi di Laurea, Universitá degli Studi di Bologna, March 1999. In Italian.

[21] R. Montemanni, D.H. Smith, and S.M. Allen. An improved lower bound for the fixed spectrum frequency assignment problem. Submitted for publication, October 2001.

[22] D.H. Smith, S.M. Allen, and S. Hurley. Characteristics of good metaheuristic algorithms for the frequency assignment problem. *Annals of Operations Research*, to appear.

[23] D.H. Smith, S. Hurley, and S.U. Thiel. Improving heuristics for the frequency assignment problem. *European Journal of Operational Research*, 107:76–86, 1998.

[24] S. Tiourine, C. Hurkens, and J.K. Lenstra. An overview of algorithmic approaches to frequency assignment problems. In *CALMA Symposium on Combinatorial ALgorithms for Military Applications*, pages 53–62, 1995.

[25] H.P. van Benthem. GRAPH. Generating Radio link frequency Assignment Problem Heuristically. Master's thesis, Delft University of Technology, May 1995.

[26] C. Voudouris. *Guided Local Search for Combinatorial Optimisation Problems.* PhD thesis, University of Essex, 1997.

[27] C. Voudouris and E. Tsang. Solving the radio link frequency assignment problem using guided local search. In *Proceedings of the NATO RTA SET/ISET Symposium on Frequency Assignment, Sharing and Conservation in Systems (Aerospace),* Aalborg, Denmark, October 1998, NATO RTO-MP13, pages 14a.1–14a.9, 1999.

[28] J.P. Warners. A potential reduction approach to the radio link frequency assignment problem. Master's thesis, Delft University of Technology, April 1995.

[29] J.P. Warners, T. Terlaky, C. Roos, and B. Jansen. A potential reduction approach to the frequency assignment problem. Technical Report 95-98, Technische Universiteit Delft, 1995.

[30] W.J. Watkins, S. Hurley, and D.H. Smith. Evaluation of models for area coverage. Technical Report No. 98003, Cardiff University, December 1998.

[31] R.M. Whitaker, S. Hurley, and D.H. Smith. Frequency assignment heuristics for area coverage problems. Technical report, Cardiff University, September 2000.