Reactive GRASP with Path Relinking for Channel Assignment in Mobile Phone Networks

Fernando C. Gomes*
Dep. de Ciência da Computação
Univ. Federal do Ceará, Fortaleza, CE, Brazil
carvalho@lia.ufc.br

Panos Pardalos and Carlos S. Oliveira
Dept. of Industrial and Systems Engineering
University of Florida, Gainesville, FL, USA
pardalos,oliveira@ufl.edu

Mauricio G. C. Resende
Shannon Laboratory
AT&T Labs Research, Florham Park, NJ, USA
mgcr@research.att.com

ABSTRACT

The Frequency Assignment Problem (FAP) arises in wireless networks when the number of available frequency channels is smaller than the number of users. FAP is NP-hard and plays an important role in the network planning. Usually, the number of available channels is much smaller than the number of users accessing the wireless network. In this case, the reuse of frequency channels is mandatory. Consequently, this may cause interference. Nowadays, cellular phone operators use various techniques designed to cope with channel shortage and, as a consequence, to avoid interference. For instance, frequency division by time or code, and local frequency clustering models have been used. These techniques are bounded by the number of users, *i.e.* as the number of users increases, they tend to become obsolete. In this work, we propose to minimize the total interference of the system, using a metaheuristic based on GRASP (Greedy Randomized Adaptive Search Procedure). A reactive heuristic has been used in order to automatically balance GRASP parameters. Furthermore, Path Relinking, which consists of an intensification strategy, has been applied. We report experimental results given by our proposed approach.

1. INTRODUCTION

With the success of mobile telephony services, some problems related to them have become research challenges. These problems can be identified in phases such as design, implementation, and management of mobile systems. A number of Problems occurring in mobile telephony are difficult and most are NP-Hard. There is an increasing demand for resources needed to implement mobile systems. An important example of such a resource is the electromagnetic frequency spectrum. As the number of users of mobile telephone systems increases, more frequencies are needed. However, due to physical constraints, the electromagnetic frequency spectrum is a limited resource. Furthermore, the number of frequencies available to mobile phone operators is restricted by regulatory agencies.

*Sponsored in part by CNPq

DIAL M 2001 7/01 Rome, Italy
© 2001 ACM ISBN 1-58113-421-5/01/07...$5.00

The search for better solutions to larger instances of mobile telephony problems has lead to improved search methods, such as metaheuristics. In this work, a metaheuristic approach is proposed to minimize the total interference in a mobile phone system. The algorithm is based on a Greedy Randomized Adaptive Search Procedure (GRASP), originally proposed by Feo and Resende [13].

This paper is organized as follows: in Section 2, concepts of the mobile cellular system and the definition of the Frequency Assignment Problem (FAP) are presented. In Section 3, previous work using different approaches to solve the FAP are reviewed. The methodologies proposed in this paper are shown in Section 4. In Section 5, computational results are given. Finally, in Section 6 concluding remarks are discussed.

2. PROBLEM DEFINITION

Cellular mobile networks have its structure defined by fixed antennas and mobile phones which can both transmit and receive electromagnetic signals. The area covered by the mobile system is divided into regions, called *cells*. A *base-station* is responsible for calls coming from and going to mobile phones located in the area covered by one or more nearby cells. One base-station holds one or more antennas. The communication between two base-stations is propagated into a cable-based network.

The availabe frequency band is divided into fixed sized slots, called frequency channels. Every antenna is assigned to a channel. The number of channels on each covered cell depends on the demand. As the number of antennas increases and the number of available channels keeps constant, channels must be reused. Reusing channels may lead to intolerable interference. Different antennas using the same channel should be separated in order to keep interference in an acceptable level.

Interference can be classified as of two main types: co-site and far-site. *Co-site* interference occurs between antennas located at the same base-station. When co-site interference occurs in antennas which point to the same cell, it is called *co-cell*. *Far-site* interference, on the other hand, occurs between antennas located in different base-stations. Far site interference can be further divided into *co-channel* and *adjacent-channel*. Interference between antennas using the same channel is known as co-channel. In the same way, but in a lesser degree, antennas using adjacent channels in the frequency spectrum cause adjacent-channel interference.

In order to limit the maximum interference occurring in a mobile system, a minimum separation in the frequency spectrum between channels used by any two antennas should be enforced. Given V, the set of antennas, let $d : V^2 \to \mathbb{Z}_+$ be a function that assigns for each pair (u, v), $u, v \in V$, a minimum separation in the frequency spectrum. If $d(u, v) = 1$, then u and v cannot use the same channel. If $d(u, v) = 2$, they cannot use the same channel or adjacent channels and so on. Generally, in order to avoid co-site and co-cell interferences, for antennas located in the same base-station, $d > 2$. It should be mentioned that, for two antennas far off from each other, $d = 0$, *i.e.* the same channel may be used. The reuse distance, $d(u, v)$ is given by the cellular service provider for each pair of antennas. Feasible solutions must comply with the given reuse distance.

The objective of the cellular FAP considered in this paper is to minimize both the system co-channel and adjacent-channel interferences. Let the *interference graph* $G = (V, E)$ be defined by the set $V = \{v_1, \ldots, v_n\}$ of antennas and by the set $E = \{(u_{i_1}, v_{j_1}), \ldots, (u_{i_m}, v_{j_m}) : d(u, v) > 0\}$ of arcs. The set of vertices adjacent to a vertex v is the *neighborhood* of v, and is denoted by $N(v)$. The intensity of co-channel and adjacent-channel interferences are represented respectively by the functions $c^{co} : E \to [0, 1]$ and $c^{ad} : E \to [0, 1]$, such that $c^{co}(u, v) \geq c^{ad}(u, v)$, $\forall\, u, v \in V$.

Let C be the set of available channels. An static assignment for the FAP is a mapping y from V to C. As the wireless network changes, static assignments should be reset.

The cellular FAP can be stated as the minimization of the function

$$f(y) = \sum_{\substack{y(v)=y(w)\\ v,w\in V}} c^{co}(v, w) + \sum_{\substack{|y(v)-y(w)|=1\\ v,w\in V}} c^{ad}(v, w) \quad (1)$$

subject to

$$|y(v) - y(w)| \geq d(v, w), \quad \forall\, v, w \in V,\ v \neq w \qquad (2)$$

3. PREVIOUS WORK

The FAP may arise in any radio network communication system. In general, the goals associated with the FAP can be seen as the minimization of three resources, namely, number of channels (order) [7], range of spectrum (span), or the total interference. A survey of different frequency assignment problems and proposed solutions can be found in [27].

Hale [19] proposed a common terminology for such problems, which is adopted in this paper. Hale also demonstrated that FAP related problems are NP-Hard, by reduction from a generalized *graph coloring* problem. Since then, connections of the FAP with other well-known problems, such as List Coloring [32] and T-Coloring [8] were found and explored.

The main distinctive feature of cellular systems is the limited number of available channels. The goal is to minimize the interference associated with the assignment. Furthermore, the channel shortage has also been tackled by frequency division either by time or code.

The frequency assignment problem has been approached in several different ways. The first generation of assignment algorithms was based on sequential assignment methods of graph coloring [6, 29]. However, these methods do not yield suitable results in the presence of adjacent-channel interference constraints [27]. Another area where there has been considerable research is that of solution methods using integer programming. Some of the most promising results use the method of column generation [22, 23]. These approaches, however, are not computationally efficient, and therefore are limited in practice to small instances of the problem.

The CALMA project (Combinatorial ALgorithms for Military Applications) was settled to study general frequency assignment problems, not restricted to mobile systems. Several alternative solutions to the problem were presented. Among them, exact, approximate and heuristic methods can be cited [1, 2, 9, 21]. For example, in Aardal *et al.* [3], the method used to solve the problem was a branch-and-cut algorithm while Hurqens and Tiourine [21] derive upper and lower bounds for the problem.

Six simple greedy heuristics were used in [5] to minimize the interference. The results reported were satisfying from a practitioner's point of view.

Most metaheuristics to date have been designed for the span or order minimization problems. For example, Genetic Algorithms and Tabu Search have been used, respectively, in [25] and [9]. Recently, Liu *et al.* [26] described a GRASP for the FAP, in which the goal is to minimize the number of violated restrictions in an assignment. The GRASP presented by Liu *et al.*uses a simulated annealing metaheuristic in the local search phase, creating a hybrid GRASP metaheuristic. Simulated Annealing (SA) has been applied to FAP of various forms (see [11, 31]).

Summing up, in recent years much work has been done in the area of frequency assignment, using many different approaches. However, with respect to the interference minimization for cellular systems, as the number of users increases research work has to be done in order to reach optimal results. This is a complex problem and only very large instances are of practical interest. In this work, we are concerned with large instances, where the number of antennas equals or surpasses that featured in real instances.

4. METHODOLOGY

Finding feasible FAP solutions is itself NP-Hard [12] . Accordingly, an objective function for the FAP that considers the number of violated constraints in the solution is used in this work. The objective function f to be minimized consists of the sum of system interferences and the number of violated constraints times a weight ρ. Therefore a greater cost is associated with a solution with more violated constraints. The objective function to be minimized can be restated as:

$$f(y) = \rho \times |\mathcal{V}| + \sum_{y(v)=y(w)} c^{co}(v,w) + \sum_{|y(v)-y(w)|=1} c^{ad}(v,w)$$

where

$$\mathcal{V} = \{(v,w) : |y(v) - y(w)| < d(y,w),\ v < w,\ v,w \in V\}.$$

4.1 GRASP

GRASP (Greedy Randomized Adaptive Search Procedure) [13, 14] is a metaheuristic based on two main processes: a solution constructor, which generates randomized solutions, and a local search heuristic which improves the solutions generated by the constructor. GRASP has been used in a large number of problems, such as scheduling problems [16], routing [20], facility planning [10], maximum independent set [15]. See [17] for an annotated bibliography of GRASP.

A general description of GRASP is given in Algorithm 1. GRASP carries on a loop while a stopping criterion is not satisfied. The criterion used can be processing time or number of iterations performed. Each iteration of GRASP consists of construction (line 2), local search (line 3), and potential solution updating (line 4).

Algorithm 1 GRASP Algorithm

1: **while** Stopping criterion not satisfied **do**
2:    Greedy randomized constructor
3:    Local search
4:    Update best solution
5: **end while**

In the construction phase, the desired solution is created step by step. The available candidates for the next element in solution are stored in a sorted list, called *Restricted Candidate List* (RCL), using a greedy ordering function. In each iteration, an element of the RCL is randomly selected. This element, however, must satisfy an acceptance criterion. For example, the element could be chosen among a percentage α of the best candidates available. After selection, the RCL must be updated according to the ordering function, so we may say that the constructor is adaptive.

For the cellular FAP, we use as constructor an algorithm which, in each step, chooses the next antenna to which a frequency will be assigned, as shown in Algorithm 2. The RCL is made out of antennas sorted by increasing order of the number of channels available for assignment. Initially, since there is no channel assigned, all antennas have the same number of available channels.

Figure 1 shows an example in which antennas 1 and 2 should be assigned a channel. Antenna 1 has three assigned neighbors, while antenna 2 has only two assigned neighbors. The constructor tends to select antenna 1, since it has less available channels for use (three channels are already assigned to its neighbors). The goal of the constructor algorithm is to ensure that antennas with fewer options are assigned first. The algorithm randomly chooses among the α elements at the top of the list, where α is automatically set by a reactive heuristic (see section 4.2). Whenever the antenna is selected, the less used channel is assigned to it.

Algorithm 2 GRASP constructor

for all $v \in V$ **do**
   $y(v) \leftarrow$ **nil**
end for
for all $f \in C$ **do**
   usage$(f) \leftarrow 0$
end for
RCL $\leftarrow$ list of $v \in V$
while $\exists v \in V, y(v) =$ **nil do**
   Randomly select w from α% first elements of the RCL
   $f \leftarrow x$ such that usage(x) is minimum
   $y(w) \leftarrow f$;
   usage$(f) \leftarrow$ usage$(f) + 1$
   Update RCL(w)
end while

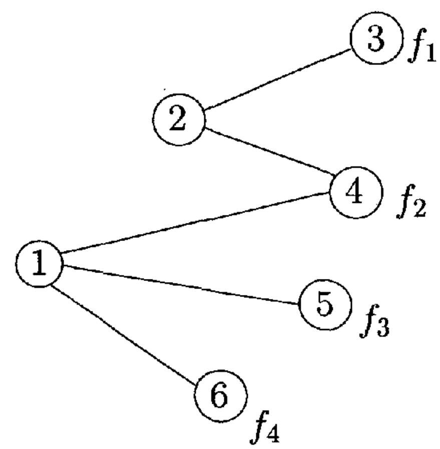

Figure 1: Example for the GRASP constructor. Antenna 1 has three assigned neighbors, while antenna 2 has only two.

Finally, the RCL must be updated to reflect the assignment. This is done by updating the set of blocked channels $B(v)$ of each antenna v adjacent to a, as shown in Algorithm 3. The changes are due to the channels blocked by the choice of $y(a)$, and represented by $B_a(v)$. The last step consists in reordering the RCL.

In order to determine the time complexity of the constructor, it should be noticed that the selection of an antenna and a channel can be done in constant time. The most costly part of Algorithm 2 is the update of the RCL. Since each antenna has a small number of neighbors, it can be considered as a fixed number. For each of them, at most four channels must be blocked, due to separation requirements, which is also

Algorithm 3 Update RCL

```
input: Antenna a ∈ V
for all v ∈ N(a) do
    Let B_a(v) = {f : |f − y(a)| < d(a, v), f ∈ C}
    B(v) ← B(v) ∪ B_a(v)
end for
Sort RCL in nondecreasing order of |B(v)|, ∀v ∈ V
```

a fixed number. Sorting of the RCL then can be done in $O(|V| \log |V|)$ if it is kept in a heap. This yields a total running time of $O(|V|^2 \log |V|)$ for the GRASP constructor.

The second important GRASP component, shown in line 3 of Algorithm 1, is the local search procedure. This procedure allows the improvement of the randomized greedy solution created by the generator through a search of the instance's local neighborhood. The local search phase of GRASP for the FAP is implemented using a *Down Hill* algorithm.

The Down Hill algorithm is a local search method that tries to improve the current solution s using a perturbation, such as changing the channel assigned to one of the antennas. If this returns a better solution s^*, the current solution s is updated to s^*. Else, if the maximum number of tries was not exceeded, it continues with a new iteration. The stopping criterion used for local search is a fixed number of iterations without improvements to the best solution. Algorithm 4 describe the steps of a generic Down Hill.

Algorithm 4 Local search

```
input: Current solution s
numtries ← 0
while numtries < MAX_TRIES do
    s* ← Perturbation(s)
    if f(s*) < f(s) then
        s ← s*
        numtries ← 0
    else
        numtries ← numtries + 1
    end if
end while
```

Tipically, for the FAP, the perturbation scheme consists simply in changing the channel assigned to one antenna. This can lead to co-site interference, resulting, therefore, in poor quality solutions. The perturbation method adopted in this paper consists in changing, at each iteration, the channels assigned to all antennas in a base-station, respecting the co-site distance separation. Algorithm 5 shows the steps of the perturbation method. First, the algorithm selects a base-station at random. Then, a starting point is selected in the frequency spectrum. Each of the antennas in the base-station sequentially receives one channel from the starting point. The channels are separated by a suitable value, MAX_SEP, sufficient to avoid co-site interference. Each iteration of the local search algorithm can be done in time proportional to $O(|V|)$, the number of antennas in a base station, leading to a very competitive performance.

Algorithm 5 Perturbation

```
input: Current solution s
MAX_FREQ ← |C|
B ← random base station
f ← random integer mod MAX_FREQ
for all v ∈ B do
    y(v) ← f
    f ← (f + MIN_SEP) mod MAX_FREQ
end for
```

4.2 Reactive GRASP

The α parameter defines the size of the Restricted Candidate List, and therefore determines diversity of the solutions found by the GRASP constructor. The traditional method used in tuning this value is through extensive experimentation. To reduce the effort need to design a GRASP, one can think of a method to determine α automatically.

In this paper we use a method proposed by Prais and Ribeiro [30], called Reactive GRASP (RG). In the RG approach, information is gathered about the quality of solutions generated with different values of α.

Initially, let $\mathcal{A}$ denote a set of candidate values for parameter α, i.e. $\mathcal{A} = \{\alpha_1, \ldots, \alpha_n\}$. For example, $\mathcal{A}$ can be set to $\{0.1, 0.2, \ldots, 1.0\}$. In each iteration, Reactive GRASP randomly selects an element from $\mathcal{A}$ and uses it as the α parameter during that iteration. Denote by p_i the probability of value α_i being chosen, for $i = 1, \ldots, n$. Initially, $p_i = 1/n$, for all $i = 1, \ldots, n$.

In order to update the values of p_i, a set $\overline{A} = \{a_1, \ldots, a_n\}$ is used. Each element of $\overline{A}$ stores the mean value a_i of solutions found using the parameter α_i. After a fixed number of iterations, the values $q_1, \ldots, q_n$ are computed:

$$q_i = \frac{f(s^*)}{a_i}, \quad 1 \leq i \leq n$$

where s^* is the best solution found up to that point. Finally, the probabilities $p_1, \ldots, p_n$ are updated:

$$p_i = q_i / \sum_{j=1}^{n} q_j, \quad 1 \leq i \leq n.$$

The RG method presents some advantages over standard GRASP. Reactive GRASP is able to find improved solutions through the use of different values for α_i, since the possibilities for exploration of search space are increased.

4.3 Path Relinking

Path Relinking (PR) is a local search method applied to metaheuristics. It was first proposed by Glover and Laguna [18]. Laguna and Martí [24] have applied this method to GRASP.

Path Relinking implements a local search guided by a high quality solution. The central idea is to find a new path, through the search space, linking the current solution to another one, previously found by GRASP, with known good solution value. By incorporating elements of the high quality solution into the current solution, an improvement may be found somewhere else in this process.

Path Relinking is used in this paper as a technique to improve solution quality. Initially, PR creates a set of guiding solutions, denoted as S_g. This is done, as shown in Algorithm 6, within PR_SIZE first iterations of GRASP, where PR_SIZE is the maximum size of the set S_g, which is a small constant value, fixed to 5 in the tests of Section 5. Then, S_g is updated to hold the best solutions found, up to that point.

Algorithm 6 Metaheuristic GRASP with Path Relinking

```
i ← 1
while Criterion not satisfied do
    Greedy Randomized Constructor
    Local Search
    if i > PR_SIZE then
        Path Relinking
    else
        Add current solution to S_g
    end if
    Update best solution
    s̄ ← worst solution in S_g
    if f(s) < f(s̄) then
        S_g ← S_g \ {s̄}
        S_g ← S_g ∪ {s}
    end if
    i ← i + 1
end while
```

PR improvement phase occurs after GRASP local search. Initially, one solution s' of S_g is selected and used as the guiding solution. At each step one antenna $a \in V$ is randomly selected and its assigned channel $y_s(a)$ in s changed to the correspondent channel $y_{s'}(a)$ in s'. In this way the current solution at each step resembles more and more s'. For each such step, a rapid local search can be done in order to improve the quality of the resulting solution. The process continues until solution s is equal to s'. The complete description of the path relinking method can be seen in Algorithm 7.

Algorithm 7 Path Relinking

```
1: input: Current solution s
2: Randomly select s' ∈ S_g
3: while s ≠ s' do
4:     Randomly select an antenna a ∈ s
5:     y_s(a) ← y_s'(a)
6:     Local Search(s)
7:     Update best solution
8: end while
```

Recent trends in Path Relinking were also incorporated, such as Reverse Path Relinking and Final Path Relinking [4]. Path Relinking provides a way to explore the path from an initial solution s to a known good solution $s' \in S_g$. It is possible though to do a search in the reverse direction in the following way: after line 3 (Algorithm 7) exchange solution s for s'.

Another improvement over PR is Final Path Relinking. This is an additional step, run at the end of GRASP execution, in which PR is executed for each pair of solutions $s_i, s_j \in S_g$, yielding k^2 executions of PR, where $k = |S_g|$. This can lead to improvements of the final solution, as the best solutions found by GRASP are stored in S_g.

5. COMPUTATIONAL RESULTS

In order to verify the results of the proposed GRASP for the FAP, two different types of instances were used:

1. Instances with known global optimum;
2. Randomly generated instances;

Each instance was executed 10 times, each with a different random number generator seed. The number of available channels in each test was fixed to 80. Execution time was limited to 1 hour for all instances. The tests were run on an IBM RISC/6000 with 128MB RAM, using AIX 4.2 operating system. The methods described were implemented using C programming language. The compiler used was gcc 2.95.1 with optimization parameter -O2.

The instances with known global optimum were used to verify the convergence of the algorithms towards a known optimum. This set of instances consists of lattice graphs that can be colored with 2, 3 or 4 colors, which global optimum is equal to zero (see [28]). Table 1 presents results found for these instances. It should be observed that the Local Search method finds the desired solutions only for small instances of the problem, i.e, up to 70 base stations and 350 antennas. GRASP, Reactive GRASP and Path Relinking always find the optimum (zero), however, there is a computational time increase due to extra processing made by these methods. The columns of Table 1 correspond respectively to the number of base stations (BS), number of antennas, mean objective function value (f), and mean time to find this value. The last two columns are repeated for each of the tested methods: local search (LS), GRASP, Reactive GRASP (RG), and Reactive GRASP with Path Relinking (PR).

instances		LS		GRASP		RG		PR	
BS	ant.	f	time	f	time	f	time	f	time
20	100	0	18	0	6	0	6	0	10
30	150	0	49	0	33	0	20	0	31
40	200	0	59	0	49	0	29	0	42
50	250	0	94	0	68	0	79	0	84
60	300	0	138	0	116	0	100	0	128
70	350	0	223	0	146	0	92	0	195
80	400	3	237	0	234	0	38	0	254
90	450	2	306	0	303	0	232	0	312
100	500	5	458	0	285	0	231	0	327
110	550	7	557	0	451	0	288	0	508
120	600	7	592	0	465	0	293	0	561
200	1000	14	713	0	517	0	326	0	682

Table 1: Results to graphs with known optima. LS (Local Search), RG (Reactive GRASP), PR (Path Relinking), BS (Base Stations), f objective function

Randomly generated instances were also used, with sizes ranging from 100 to 600 antennas, in steps of 25, resulting in 20 random instances. We used these instances to test the behavior of the algorithms, considering processing time, and the increase of instances' complexity. The parameters use for the generation of these instances are based on real world data (see [28]). The following methods were tested: 1) GRASP; 2) The local search procedure – the initial solution was created by the GRASP constructor – and 3) T-Coloring, a greedy algorithm explained in Borndörfer *et al.* [5].

In a second step, the following methods were added to the initial GRASP, in order to improve its results: 1) Reactive GRASP; 2) Path Relinking; 3) Reverse Path Relinking; and 4) Final Path Relinking.

The experiments used the same neighborhood and parameters for all tested instances.

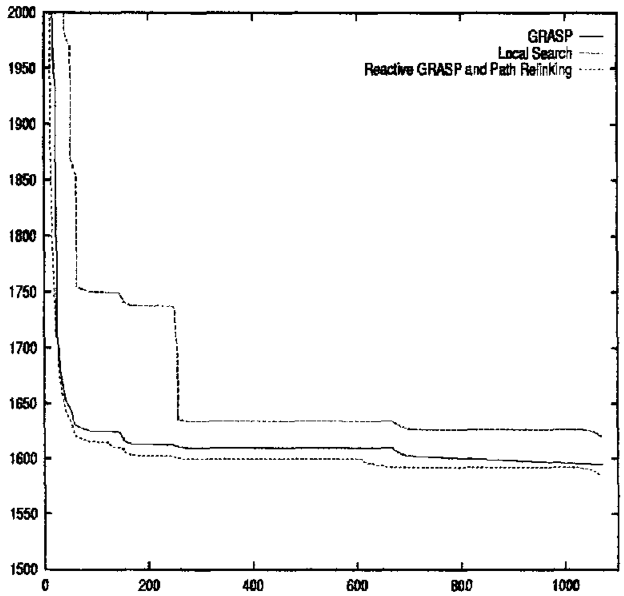

Figure 2: Comparison of objective function values for one instance with 600 antennas using different methods *versus* time.

Figure 2 shows a comparison of Local Search, GRASP and Reactive GRASP with Path Relinking for one randomly generated instance with 600 antennas. In this plot the value of objective function *versus* time is presented. It can be seen that GRASP converges faster than simple Local Search. Moreover, the combination of Reactive GRASP and Path Relinking yields better results than simple GRASP. Figure 3 presents the evolution of the objective function for some of the test instances, with 50 to 120 antennas. The data on this graph was output by the Reactive GRASP with Path Relinking method. It can be seen that there are only small variations in the time necessary to find good solutions (around 30 seconds) and that good solutions are found early in the process (first 100 seconds). After this, only small improvements are recorded.

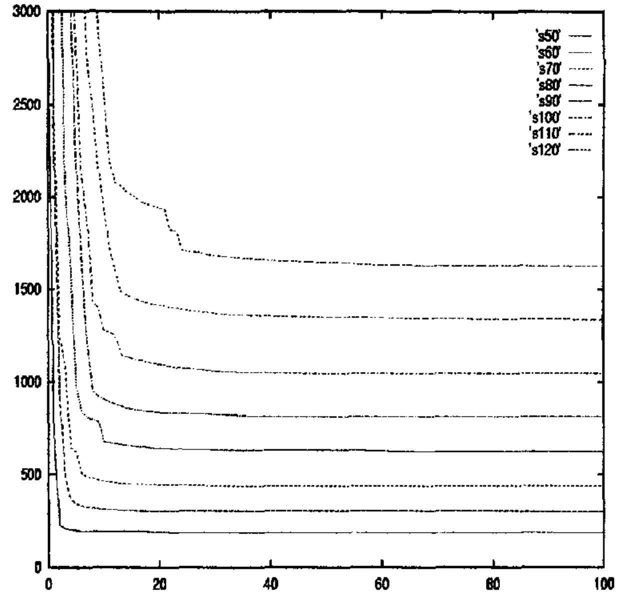

Figure 3: Comparison of objective function values *versus* time for instances with 50, 60, 70, 80, 90, 100, 110 and 120 antennas. The solution method used was Reactive GRASP with Path Relinking.

Table 2 shows results of tests performed with random generated instances using the methods investigated in this paper. The columns correspond to the number of antennas in the instance, the method used (GRASP, T-coloring heuristic (TC), Local Search (LS), Reactive GRASP (RG), Path Relinking (PR), Reverse Path Relinking (RPR) and Final Path Relinking (FPR)), the average value of objective function, the best value found in all executions, time (seconds), number of iterations and the value for α when the best solution was found (this is shown only with Reactive GRASP results).

The performance of GRASP was tested against the heuristic T-Coloring [5] and the plain local search method. It can be noted that T-Coloring gives inferior results over the GRASP and Local Search. This happens because T-Coloring does not consider the whole base-station, as the other methods do. These results are shown in the first three lines for each instance.

The next five lines show the result of modifications proposed to the initial GRASP: Reactive GRASP, Path Relinking, Reverse PR and Final PR. It can be seen that Reactive GRASP gives better results than those found by the simple GRASP, since it is able to get an appropriate value for the α parameter. This parameter is most of the time 0.1 or 0.2, according to the results. The list of candidate values for α used was $\{\alpha_1 = 0.1, \alpha_2 = 0.2, \ldots, \alpha_{10} = 1\}$. The combination of GRASP Reactive and Path Relinking gives an improved solution for most of the instances. However, computational time for this combination is worse, due to

inst.	method	average	best	time(s)	# iter.	α
450	GRASP	781	778	1752	43	
	T-Col	18842	793	31		
	LS	795	793	2812		
	RG	780	775	1838	54	0.2
	PR	778	773	2723	71	
	RPR	769	766	3521	28	
	FPR	770	765	3556	29	
475	GRASP	903	902	2085	39	
	T-Col	29487	913	39		
	LS	915	913	2163		
	RG	903	899	1953	48	0.2
	PR	903	899	2530	53	
	RPR	889	889	3364	25	
	FPR	896	889	3616	23	
500	GRASP	1028	1025	3238	41	
	T-Col	32322	2303	46		
	LS	1036	1035	2956		
	RG	1026	1020	2597	51	0.2
	PR	1025	1017	2317	49	
	RPR	1011	1001	3492	22	
	FPR	1004	1001	3521	21	
525	GRASP	1169	1162	2851	38	
	T-Col	49320	1177	58		
	LS	1177	1177	1825		
	RG	1167	1158	2263	44	0.3
	PR	1165	1156	2941	47	
	RPR	1148	1145	3537	22	
	FPR	1147	1145	3554	20	
550	GRASP	1308	1298	3129	36	
	T-Col	61302	1319	69		
	LS	1321	1319	2531		
	RG	1305	1301	3015	43	0.2
	PR	1303	1297	3384	34	
	RPR	1287	1278	3371	20	
	FPR	1280	1278	3433	22	
600	GRASP	1594	1589	2071	19	
	T-Col	11651	1606	81		
	LS	1612	1606	3287		
	RG	1593	1584	2834	35	0.2
	PR	1594	1580	3186	28	
	RPR	1562	1560	2572	19	
	FPR	1563	1555	3562	23	

Table 2: Experiments with GRASP

greater effort made by Path Relinking. The last two lines for each instance shows the value found by adding Reverse Path Relinking and Final Path Relinking to the GRASP, as explained in Section 4.3. The two methods gave noticeable contribution to most of the solutions due to an intensification of the search phase.

6. CONCLUDING REMARKS

This paper gives a metaheuristic solution to the Frequency Assignment Problem for mobile telephony systems, using the GRASP methodology. This problem is of practical interest, since the increasing number of users of mobile systems demands better management of the frequency spectrum. We proposed a GRASP, with constructor and local search methods, which was tested against greedy heuristics, showing good results for the tested instances. Furthermore, Reactive GRASP and Path Relinking were added to GRASP. The integration of these methods allowed them to find improved solutions over those found by GRASP. The proposed methods proved reliable and are applicable to large instances of the FAP.

7. REFERENCES

[1] K. Aardal, C. A. Hurkens, J. K. Lenstra, and S. R. Tiourine. Algorithms for frequency assignment problems. *CWI Quarterly*, 9, 1996.

[2] K. Aardal, C. A. J. Hurkens, J. K. Lenstra, and S. R. Tiourine. An overview of algorithmic approaches to frequency assignment problems, CALMA project. Technical report, Dep. of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, Holanda, 1995.

[3] K. I. Aardal, A. Hipolito, C. P. M. Van Hoesel, B. Jansen, C. Roos, and T. Terlaky. A branch-and-cut algorithm for the frequency assignment problem. Technical report, Eindhoven University, 1998.

[4] R.M. Aiex, M.G.C. Resende, P.M. Pardalos, and G. Toraldo. GRASP and path relinking for the three-index assignment problem. Technical report, AT&T Labs Research, Florham Park, NJ, 2000.

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

[6] D. Brélaz. New methods to color the vertices of a graph. *Communications of ACM*, 22(4):251–256, April 1979.

[7] A. Capone and M. Trubian. Channel assignment problem in cellular systems: A new model and a tabu search algorithm. *IEEE Transactions on Vehicular Technology*, 48(4):1252–1260, July 1999.

[8] D. Costa. On the use of some known methods for T-colorings of graphs. *Annals of Operations Research*, 41:343–358, 1993.

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

[10] H. Delmaire, J. A. Díaz, E. Fernández, and M. Ortega. Comparing new heuristics for the pure integer capacitated plant location problem. Technical Report DR97/10, Department of Statistics and Operations Research, Universitat Politecnica de Catalunya, Barcelona, Spain, 1997.

[11] M. Duque-Antón, D. Kunz, and B. Rüber. Channel assignment for cellular radio using simulated annealing. *IEEE Transactions on Vehicular Technology*, 42(1):14–21, 1993.

[12] A. Eisenblätter. A frequency assignment problem in cellular phone networks (extended abstract). In *Network Design, Connectivity and Facility Location*, volume 35 of *Dimacs Series in Discrete Mathematics and Theoretical Computer Science*. American Mathematical Society, 1997.

[13] T. A. Feo and M. G. C. Resende. A probabilistic heuristic for a computationally difficult set covering problem. *Operations Research Letters*, 8:67–71, 1989.

[14] T. A. Feo and M. G. C. Resende. Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6:109–133, 1995.

[15] T. A. Feo, M. G. C. Resende, and S. H. Smith. A greedy randomized adaptive search procedure for maximum independent set. *Operations Research*, 42:860–878, 1994.

[16] T. A. Feo, K. Venkatraman, and J. F. Bard. A GRASP for a difficult single machine scheduling problem. *Computers & Operations Research*, 18:635–643, 1991.

[17] P. Festa and M. G. C. Resende. GRASP: An annotated bibliography. To appear in "Essays and Surveys on Metaheuristics," P. Hansen and C.C. Ribeiro, eds., Kluwer Academic Publishers, 2001.

[18] F. Glover and M. Laguna. *Tabu Search*. Kluwer Academic Publishers, Boston, 1997.

[19] W. K. Hale. Frequency assignment: Theory and applications. *Proceedings of the IEEE*, 68(12):1497–1513, December 1980.

[20] C. A. Hjorring. *The vehicle routing problem and local search metaheuristics*. PhD thesis, University of Auckland, Auckland, New Zealand, 1995.

[21] C. Hurkens and S. Tiourine. Upper and lower bounding techniques for frequency assignment problems. Technical report, Eindhoven University, 1995.

[22] B. Jumard, O. Marcotte, and C. Meyer. Mathematical models and exact methods for channel assignment in cellular networks. *Les Cahiers du GERAD*, March 1998.

[23] B. Jumard and T. Vovor. A column generation approach for the exact solution of channel assignment problems. *Les Cahiers du GERAD*, July 1998.

[24] M. Laguna and R. Martí. GRASP and path relinking for 2-layer straight line crossing minimization. *INFORMS Journal on Computing*, 11:44–52, 1999.

[25] W. K. Lai and G. G. Coghill. Channel assignment through evolutionary optimization. *IEEE Transactions on Vehicular Technology*, 45(1), February 1996.

[26] X. Liu, P. M. Pardalos, S. Rajasekaran, and M. G. C. Resende. A GRASP for frequency assignment in mobile radio networks. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 52:195–201, 2000.

[27] R. A. Murphey, P. M. Pardalos, and M. G. C. Resende. Frequency assignment problems. In D.-Z. Du and P. M. Pardalos, editors, *Handbook of Combinatorial Optimization*, volume 4, pages 295–377. Kluwer Academic Publishers, 1999.

[28] C. A. S. Oliveira. Alocação de freqüências para telefonia celular: Uma solução através do procedimento de busca guloso adaptativo aleatório. Master's thesis, Universidade Federal do Ceará, 2000.

[29] J. Peemöller. A correction to Brelaz's modification of Brown's coloring algorithm. *Communications of ACM*, 26(8):595–599, August 1983.

[30] M. Prais and C. C. Ribeiro. Reactive GRASP: An application to a matrix decomposition problem in TDMA traffic assignment. *INFORMS Journal on Computing*, 12(3):164–176, 2000.

[31] A. Quellmalz, A. Knälmann, and B. Müller. Efficient frequency assignment with simulated annealing. In *IEEE ninth international conference on antennas and propagation*, volume 2, Eindhoven, The Netherlands, 1995.

[32] Z. Tuza. Graph colorings with local constraints a survey. *Discussiones Mathematicae Graph Theory*, 17(2):161–228, 1997.