An Evolutionary Approach for Frequency Assignment in Cellular Radio Networks*

Raphaël Dorne and Jin-Kao Hao
LGI2P
EMA-EERIE
Parc Scientifique Georges Besse
F-30000 Nîmes
France
email: {dorne, hao}@eerie.fr

ABSTRACT

This paper presents a study of Evolutionary Algorithms (EAs) for a real application: the Frequency Assignment Problem (FAP) in Cellular Radio Networks. This problem is of great importance both in practice and in theory. In practice, solving this problem efficiently will allow the telecommunications operator to manage larger and larger cellular networks. In theory, the simplification of FAP is reduced to the graph coloring problem which is NP-complete. In our work, we take a progressive approach: first, we study separately the different components of EAs in order to understand the interest of each of them for our application; then, we design hybrid EAs which integrate efficient techniques (local search, constraint programming, etc.) into evolutionary operators. Experiments using our approach on real-size FAP instances (up to 300 cells and 13000 constraints) give very encouraging results. Direct comparison of our approach with simulated annealing (SA), constraint programming (CP) and graph coloring algorithms on the same set of tests shows the strong interest of our hybrid evolutionary approach for this application.

1. Introduction

The Frequency Assignment Problem (FAP) in Cellular Radio Networks is a very complex application in the field of telecommunications. The main goal is to serve the maximal number of network users with limited transmission resources. The transmission resource is an available radio spectrum which consists of a limited number of frequencies (or channels). FAP consists in assigning frequencies to each radio cell in such a way that a set of constraints is satisfied. These constraints can be classified into three categories defined as follows:

1. The limited number of available frequencies in the radio spectrum.
2. The traffic constraints corresponding to the minimum number of necessary frequencies indispensable for covering communications between mobiles moving on this cell.
3. The interference constraints among cells classified into two categories:
   Adjacent channel constraints: the frequencies assigned to two adjacent cells must be sufficiently separated in the frequency domain. Two cells are adjacent if they emit within a common area even if they are not geographically adjacent.
   Co-cell constraints: any pair of frequencies assigned to a cell must have a certain distance between them.

The global technique comes down to finding a good frequency assignment to favor the same frequency reuse by sufficiently distant cells which allows the number of communications over the network to be maximized with a limited number of frequencies. Different optimization versions of FAP could be developed such as maximizing the total traffic, minimizing the number of frequencies used and minimizing the interferences over the network.

The satisfiability of this problem can be shown to be NP-complete because it is reduced to the graph k-coloring problem [10]. Many methods have been proposed to solve FAP, including 1) classic methods: graph coloring algorithms [10, 9] and integer programming; 2) heuristic methods: neural networks [17, 8], genetic algorithms (GAs) [4, 16], local search such as simulated annealing (SA) [6, 1] and Tabu search [13], and constraint programming

*Supported by the CNET (French National Research Center for Telecommunications) under the grant No.940B006-01.

(CP) [3].

In this paper, we present a hybrid evolutionary approach combined with local search. Our hybrid methods have been tested on real-size FAP instances (up to 300 cells and 13000 constraints) and give very encouraging results. Direct comparison of our approach with SA, CP and graph coloring algorithms on the same set of tests shows the strong interest of our hybrid evolutionary approach for this application.

The paper is organized as follows. In Section 2, FAP is modelled as an optimization problem. In Section 3, we deal with the representation issues and the evolution mechanisms. In Section 4, experimental results are presented and compared with SA and CP. Future work and perspectives are discussed in the last section.

2. Frequency Assignment Problem Modelling

Given N the number of cells, *NBfreq* the number of available frequencies in the spectrum, *NBinter* the total number of interference constraints defined for each pair of adjacent cells, FAP can be modelled with a quadruple <X,D,C,F> representing a constrained optimization problem (COP) with:

X = {Ci | Ci is a cell of the network, i $\in$ [1..N]}.

D = {Fi | Fi is an available frequency of the spectrum, i $\in$ [1..NBfreq]}.

C = T $\cup$ I

- T = {Ti | Ti minimal number of frequencies necessary for Ci, i $\in$ [1..N]}.
- I = {Ii | Ii interference constraints between two cells, i $\in$ [1..NBinter]}.

F = fitness function of a frequency assignment of the network.

2.1. Constraints

The traffic constraint of each cell is represented by Ti which is an integer coding the minimal number of frequencies necessary for Ci to cover its maximum traffic communications. In reality, this maximum traffic value is defined by an estimation of the maximum number of mobiles which can move at the same time within this cell.

The interference constraints are represented by a matrix M of N*N and with fi,n corresponding to the n^{th} frequencies of Ci, each element of M defines the set of constraints as follows:

- M[i,j] with i $\neq$ j represents the minimal number of channel separations required to satisfy the adjacent channel constraints between the cells Ci and Cj.
  $\forall$ n$\in$[1..Ti], $\forall$ m$\in$[1..Tj], |fi,n - fj,m| $\geq$M[i,j]
- M[i,i] represents the channel separations necessary to satisfy the co-cell constraints:
  $\forall$ n,m $\in$ [1..Ti], n $\neq$ m, | fi,n - fi,m | $\geq$ M[i,i]
- M[i,j]=0 means there is no constraint between the cells Ci and Cj.

3. Evolutionary Approach for FAP

EAs include algorithms based on a population of individuals and on a set of evolution operators [15, 12]. This class contains, among others, Genetic Algorithms (GAs) [14, 11], Evolution Strategies (SE) [2], and Evolutionary Programming (EP) [7]. Although EAs are general search methods applicable to many domains, it is generally recognized that it is both necessary and beneficial to combine EAs with other efficient search methods and integrate special knowledge into EAs whenever possible [5]. For our application, we take a progressive approach: first, we study separately the different components of EAs in order to understand the interest of each of them for our application; then, we design hybrid EAs which integrate efficient techniques into evolutionary operators.

3.1. Chromosome Representation

One chromosome corresponds to a complete frequency assignment and one gene to one frequency of a cell. The length of a chromosome is thus equal to the total traffic of all the cells in the network (see fig. 1).

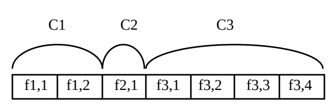

Fig. 1: Chromosome representation

Other representations are possible. For instance, one may use a NBfreq*N boolean matrix, each element of the matrix indicating if a frequency is assigned to a cell. One may also use a heterogeneous 2 dimensional representation where to each cell corresponds a number of frequency values equal to its traffic.

The chosen representation (see fig. 1) has some desirable properties compared with other representations. First, the number of genes in a chromosome is minimized. Second mutation operators can be directly applied. Third, crossover operators can be applied with a minimum constraint of choosing crossover points at the beginning of the genes of each cell.

3.2. Evaluation of Frequency Assignments

The fitness function F: P $\rightarrow$ [0...NBinter] associates with each chromosome of a population P (the set of frequency assignments), an integer corresponding to the number of *unsatisfied* interference constraints. For any p of P, F(p) is the total number of unsatisfied interference constraints for the assignment p. Therefore, a chromosome p is a solution (a frequency assignment without interference) if and only if F(p) = 0. Different assignments could be a solution of the problem and the telecommunications operator can choose one of them with other criteria such as the real interference cost, or the maximum reuse of each frequency value, and so on.

3.3. Evolution Mechanisms

The complexity in finding an efficient evolutionary algorithm lies in the difficulty of combining its different components well. Indeed, the simultaneous application of the operators of selection, crossover and mutation to a population becomes so complex that we cannot determine which operator is really efficient or not. For this reason, we take a step-by-step approach. We begin with a very simple EA which does not use all the genetic operators. We then slowly add another evolutionary component to the EA, keeping in mind the objective of understanding the interactions between these different elements.

3.3.1. Mutation & Selection With a Population of One Chromosome

Our first mechanism is a largely simplified one with no crossover and it manipulates a population of only one chromosome. This algorithm allows us to study the different possibilities of the mutation operator and to select the most efficient one. Its generation cycle is described as follows:

```
generate(P); // random population having 1 individual
evaluate(P);
while ( not(stop condition) ) do {
   choose(I) in P; // I is unique
   I'= Mutation(I);
   if ((I' better than I) or (probability(q) verified))
      // q = probability of deterioration
   then P=I';
   else P=I; }
```

The first mechanism uses a selection operator which is controlled by a deterioration probability. The mutation is traditionaly considered as a random operator of local change with a low rate [14]. Our objective is to include in this operator some special knowledge about the application and some controls. Our mutation operator is composed of three steps: the selection of a cell, the selection of a frequency (gene) for the cell, then the selection of a value (allele) for the frequency of the cell (see fig. 2). For each step, several heuristics have been developed and are described below:

Selection of a cell: one cell chosen randomly among all the cells; one cell chosen randomly among the cells in conflict. A cell is in conflict if at least one of its frequencies does not verify all interference constraints.

Selection of a frequency: randomly among all the frequencies of the cell; randomly among all the frequencies in conflict of the cell.

Selection of a frequency value: randomly; best value; best value different from the current one.

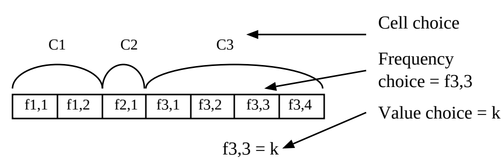

Fig. 2: Decomposition of mutation operator

We use special knowledge about FAP to determine a cell in conflict, and then a value for this cell from its frequency domain.

All the possible combinations have been tested and results will be reported elsewhere. Suffice it to say that the combination "conflict cell/conflict frequency/best different value" outperforms other ones. And it is this combination that is used in our experiments.

Note that this single chromosome algorithm resembles a stochastic hill climber such as SA with a fixed schedule and a different neighbourhood function. However they are different in that our algorithm uses a controlled three steps selection strategy which is *a priori* more intelligent than pure random choices used by SA.

3.3.2. Mutation & Selection With a Population of N Chromosomes

The notion of population is now added to the previous mechanism and the selection operator plays a more important role. The selection is composed of the choice of chromosomes and the acceptance conditions of these chromosomes in the next population.

```
generate(P); // random population having N individuals
evaluate(P);
while ( not(stop condition) ) do {
  while ( not(P' filled) ) do {
    choose(I) in P;
    I' = mutation(I);
    if ((I' better than I) or (probability(q) verified))
    then add I' to P' }
  P=P'; }
```

The use of a population is justified by the improvements we may get when a set of individuals is running in competition. Indeed by selecting well-fitted individuals, the population will be directed to local or global optima faster than a set of individuals running independently. Furthermore, to escape from a local optimum, one individual uses only one exploring path whereas a population with several paths has a greater chance of continuing its search towards a global optimum. The selection mechanism favors the individuals which still have a fitness increase compared with those which are already in a local optimum. This helps the EA to converge more rapidly. Note however that the selection may prevent the population from escaping from local optima.

4. Experimental Results

Our approach has been tested with a set of 18 FAP instances provided by the French National Research Center on Telecommunications. The traffic for all the cells is limited to one frequency and the minimum distance between two frequencies of a pair of adjacent cells is fixed to one. It is easy to see that these tests are isomorphic to the graph coloring problem.

The names of the instances consist of three numbers nf.nc.d which are respectively the minimum number of frequencies needed to find an optimal solution (an assignment without interference), the number of cells and the density of interference constraints over the network. For example, 8.150.30 defines a problem composed of 150 cells with 8 available frequencies in the spectrum and 30 % of 150*(150-1)/2 the total constraints. For large problems having a high density we obtain up to 13000 constraints.

Table 1 gives comparative results of our approach with SA [1] and CP [3] which both are shown to be more efficient than classic graph coloring methods [1]. Two comparative criteria are used and explained below: the number of frequency excess and timing. Other criteria such as the evaluation number of the fitness function would have been more interesting, however such data are not available for SA and CP.

Excess (of frequencies): the number of frequencies added to the minimum of the optimal solution. For instance, for the problem 8.150.20, an excess of 2 frequencies of a method means that the method can only find a satisfying assignment without interference by adding 2 extra frequencies. This criterion is essential because adding even one frequency may make the initial problem much easier.

Time: the average time in seconds for one successful run. Our tests have been carried out on SUN SPARC10 stations, whereas those of SA [1] and CP [3] have been run on computers considered to be at least 3 times faster.

% success: success rate over 10 runs.

The characteristics of our two algorithms are given below.

• Mutation & Selection with 1 individual (M&S (1))

Selection: With a rate of deterioration determined empirically
Mutation:
Selection of a cell: Randomly among the cells in conflict
Selection of frequency: The only possible frequency (traffic = 1)
Selection of value: Best value different from the current one
Population: 1 individual
Stop cond.: A limited number of generations

• Mutation & Selection with 20 individuals (M&S (20))

Selection: The best and a rate of deterioration determined empirically
Mutation: Idem above
Population: 20 individuals
Stop cond: A limited number of generations

Among the 18 FAP instances, four are very difficult (in bold) for all the methods. The results of CP presented are the synthesis of a set of complete constraint methods using classic heuristic techniques to choose variables and values [3]. For each problem we reported the best result obtained by at least one of these heuristics. CP never finds the global optimum for the four hard instances of FAP and needs an excess of up to 10 frequencies to solve 15.300.30. To better understand the influence of the excess of frequencies on the complexity of the tests, it will be sufficient to make a parallel with graph coloring problems. Indeed, coloring a graph with n+k (k$\geq$1) colors is really much easier than coloring one with n colors. For example, with 4 extra frequencies for

problems	CP Excess /time	SA Excess /time	M&S(1) Excess /time	M&S(20) Excess /time
4.75.10	+0/1	+0/7	+0/1.3	+0/4.65
4.75.20	+0/1	+0/7	+0/2.6	+0/14.31
4.75.30	+0/1	+0/4	+0/1.9	+0/11.75
8.75.10	+0/1	+0/4	+0/0.08	+0/1.06
8.75.20	+0/1	+0/8	+0/0.27	+0/3.2
8.75.30	+0/1	+0/20	+0/3.7	+0/15.84
8.150.10	+0/1	+0/15	+0/0.96	+0/10.24
8.150.20	+2/13	+1/7237	+0/330	+0/7707
8.150.30	+6/1	+0/33	+0/21.85	+0/143.24
15.150.10	+0/1	+0/7	+0/0.35	+0/5.49
15.150.20	+0/1	+0/23	+0/97	+0/14.33
15.150.30	+0/1	+0/40	+0/18.87	+0/51.36
15.300.10	+0/1	+0/61	+0/2.4	+0/39.81
15.300.20	+3/1	+1/9190	+0/920	+0/12277
15.300.30	+10/1560	+4/10945	+0/1243	+0/12562
30.300.10	+0/1	+0/68	+0/2	+0/31
30.300.20	+0/1	+0/66	+0/6.33	+0/84
30.300.30	-	+0/840	+0/70	+0/223

Table 1: Comparison with SA and CP

problems	M&S(1) Nb eval. per ind. / % success	M&S(20) Nb eval. per ind. / % success
8.150.10	1526 / 100%	949 / 100%
8.150.20	299180 / 100%	173124 / 10%
8.150.30	25466 / 100%	9427 / 100%
15.150.10	868 / 100%	742 / 100%
15.150.20	1862 / 100%	1479 / 100%
15.150.30	28000 / 100%	4227 / 100%
15.300.10	3332 / 100%	2961 / 100%
15.300.20	957894 / 100%	275793 / 10%
15.300.30	1011724 / 60%	275563 / 100%
30.300.10	3538 / 100%	3229 / 100%
30.300.20	6728 / 100%	5697 / 100%
30.300.30	64728 / 100%	11387 / 100%

Table 2: Comparison of the two evolutive mechanisms

15.300.30, our algorithm M&S(1) find an optimal solution in 483 sec. with a success rate of 100%. CP shows its limits when the number of constraints begins to be high (density = 30%) and then needs an excess of 6 frequencies to solve the 8.150.30 whereas all the other methods easily find a global optimum without excesses. This fact proves the interest of having an incomplete approach for FAP.

SA cannot solve 3 of 4 hard instances without excesses[1]. This method has better results than CP with a lower number of excesses but still needs 4 extra frequencies for 15.300.30, which remains very high.

Our methods show their efficiency on all the tests and find optimal solutions without any excess of frequency even for the hard instances 8.150.20, 15.300.20, 15.300.30. Our results are clearly better than those of SA which are nevertheless close to our M&S(1). We think this is partially due to our techniques in choosing cells in conflict and values of frequency, which are more intelligent than random choices as used in SA.

Running times are close to CP for small instances and remain clearly faster than those of SA even for M&S(20) which manipulates nevertheless a population.

To compare the efficiency of M&S(1) and M&S(20), we use as the main criterion the number of evaluations needed per individual to obtain an optimal solution over 10 runs. This criterion allows the performances of a population of N individuals to be compared with those of N independent runs of only one individual without punishing the population for its higher running time. Table 2 shows the effects of populations on the performances.

As can be seen from table 2, the EA with a real population seems more efficient in terms of the number of evaluations needed to solve all the problems. This is true in particular for the 15.300.30 for which it obtains a success rate of 100 % with only 275563 evaluations compared with a success rate of 60 % and over 1 000 000 evaluations using M&S(1). However, M&S(1) has a better success rate than M&S(20) for 8.150.20 and 15.300.20. The exact reason for this is not clear, but the following observations may help explain this point.

The competition among individuals may be a double-side knife. On the one hand, a good competition mechanism helps the EA to converge more efficiently and more precisely. On the other hand, it may constitute an obstacle for an EA to escape from local optima. Indeed, all the individuals which are trying to escape from a local optimum are in competition with those close to a local optimum and having thus a better fitness. By consequence, the selection mechanism may abandon the individuals escaping from local optima and therefore may force the population to stay at the same set of local optima found so far. This may be what happens with M&S(20) on the instances 8.150.20 and 15.300.20. We think that there are some deep local optima in these 2 difficult instances which need many large oscillations. In the case of M&S(1), there is no individual competition, and the single individual can deteriorate and therefore realize the required oscillations to reach global optima.

It is interesting to determine the real influence of the population on the performance of our EAs. Limited tests were carried out with populations of 50 and 100 individuals using different selection operators. We noticed that in terms of evaluation number, bigger populations do give better results for easy instances. But for difficult instances, no improvement is observed. However, more work is needed to confirm or disconfirm these observations.

5. Conclusions

In this paper, we have presented two classes of evolutionary mechanisms without crossover for the frequency assignment problem. In particular, we have studied different controlled mutation and selection operators for our application. We have shown once

[1] According to a personal communication, a recently improved SA managed to solve these instances.

again that the integration of local search and special knowledge about the given application into these operators can largely speed up the performance of EAs. Experiments with real-size test instances give very encouraging results and show clearly the interest of a hybrid evolutionary approach for this application. Also we think the techniques developed here can be easily adapted to many optimization applications.

6. Future Work

Currently, we are working on several points. First, we are investigating other selection operators which will permit to escape more easily from local optima. Also, we are studying the influence of population using various selection mechanisms for solving difficult problems. Second, we are testing different classic crossovers such as monopoint, uniform and developing specialized crossovers to our problem. At this time we don't have enough interesting results but we think that crossover, which carries out the information exchange among individuals, may have some advantages over the other operators. In particular, for highly constrained optimization problems like FAP, crossover may help to escape from local optima and introduce a better exploration of the search space. This may lead to better results both in quality and in solving speed. Third, we are working on more difficult FAP instances which have the following features: the traffic of a cell may be greater than one and the distance between two assigned frequencies is no longer limited to only one frequency. The results on these instances will be reported in the near future.

Finally, we plan to use the methods described in this paper to tackle other difficult constrainted optimisation problems.

Acknowledgements

We would like to thank the three referees for their useful comments which helped to improve this paper and the CNET which supported this work. Special thanks go to A. Caminada and R. Mignone from the CNET for their various help about FAP.

References

[1] A. Akrout. Problèmes d'affectation de fréquences: méthodes basées sur le recuit simulé. Technical Report RP/PAB/SRM/RRM/4123, CNET, 1994.

[2] T. Bäck, F. Hoffmeister, and H.P. Schwefel. A survey of evolution strategies. In *4th International Conference on Genetic Algorithms*, La Jolla, CA, 1991.

[3] A. Caminada. Résolution du problème de l'affectation des fréquences par programmation par contraintes. Technical Report FT.CNET/CNET BEL/POH/CDI/71-95/CA, CNET, 1995.

[4] W. Crompton, S. Hurley, and N.M. Stephen. A parallel genetic algorithms for frequency assignment problems. In *IMACS SPRANN94*, pages 81–84, 1994.

[5] L. Davis. *Handbook of Genetic Algorithms.* Van Nostrand Reinhold, New York, 1991.

[6] M. Duque-Anton, D. Kunz, and B. Rüber. Channel assignment for cellular radio using simulated annealing. In *IEEE Trans. Veh. Technol.*, volume 42, pages 14–21, 1993.

[7] D.B. Fogel. Evolutionary programming: An introduction and some current directions. *Statistics and Computing*, 1:11–129, 1994.

[8] N. Funakini and Y. Takefuji. A neural network parallel algorithm for channel assignment problems in cellular radio network. In *IEEE Trans. Veh. Technol.*, volume 41, pages 430–437, 1992.

[9] A. Gamst. Some lower bounds for a class of frequency assignment problems. In *IEEE Trans. Veh. Technol.*, volume 35, pages 8–14, 1986.

[10] A. Gamst and W. Rave. On the frequency assignment in mobile automatic telephone systems. In *GLOBECOM 82*, pages 309–315, 1982.

[11] D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, 1989.

[12] D.E. Goldberg. Genetic algorithms and evolutionary algorithms. In *Communications of the ACM*, volume 37, 1994.

[13] J.K. Hao and L. Perrier. Tabu search for channel assignment problems. submitted, 1995.

[14] J. Holland. *Adaptation in Natural and Artificial Systems.* University of Michigan Press, Ann Arbor, 1975.

[15] K. De Jong and W. Spears. On the state of evolutionary computation. In *International Conference on Genetic Algorithms*, pages 618–623, Urbana-Champaign, 1993.

[16] A. Kapsalis, V.J. Rayward-Smith, and G.D. Smith. Using genetic algorithms to solve the radio link frequency assignment problem. In *International Conference on ANN and GAs*, pages 37–40, Alès, France, 1995.

[17] D. Kunz. Channel assignment for cellular radio using neural networks. In *IEEE Trans. Veh. Technol.*, volume 40, pages 188–193, 1991.