Designing cellular networks using a parallel hybrid metaheuristic on the computational grid

E.-G. Talbi *, S. Cahon, N. Melab

CNRS/LIFL and INRIA, University of Lille, 59655 Villeneuve d’Ascq Cedex, France

Abstract

Cellular network design is a major issue in mobile telecommunication systems. In this paper, a model of the problem in its full practical complexity, based on multiobjective constrained combinatorial optimization, has been investigated. We adopted the Pareto approach at resolution in order to compute a set of diversified non-dominated networks, thus removing the need for the designer to rank or weight objectives a priori. We designed and implemented a “ready-to-use” platform for radio network optimization that is flexible regarding both the modeling of the problem (adding, removing, updating new antagonist objectives and constraints) and the solution methods. It extends the “white-box” ParadisEO framework for metaheuristics applied to the resolution of mono/multi-objective Combinatorial Optimization Problems requiring both the use of advanced optimization methods and the exploitation of large-scale parallel and distributed environments. Specific coding scheme and genetic and neighborhood operators have been designed and embedded. On the other side, we make use of many generic features related to advanced intensification and diversification search techniques, hybridization of metaheuristics and grid computing for the distribution of the computations. They aim at improving the quality of networks and robustness, to speed-up the search and obtain results in a tractable time, hence efficiently solving large instances of the problem. Using three realistic benchmarks, the computed networks and speed-ups on different parallel and/or distributed architectures show the efficiency and the scalability of hierarchical models of hybridization and parallelization used in conjunction.

© 2006 Published by Elsevier B.V.

1. Introduction

The design of large cellular networks is a complex task with a great impact on the quality of service and the cost of the network. Engineering of mobile telecommunication networks involves two major problems: the design of the network and the frequency planning. The design consists in positioning base stations (BS) on potential sites, in order to fulfill some objectives and constraints [20]. The frequency planning sets up frequencies used by BS with criteria of reusing. In this paper, we address the first problem. Network design is an NP-hard combinatorial optimization problem [21]. The BS positioning problem deals with finding a set of sites for antennas from a set of pre-defined candidate sites, determining the type and the number of antennas, and setting up the configuration of different parameters of the antennas (tilt, azimuth, power, ...). A new formulation of the problem of BS positioning is given as a multiobjective constrained combinatorial optimization problem. The model deals with specific objectives and constraints due to the engineering of cellular radio network. Reducing costs without sacrificing the quality of service are issues of concern.

Most of the proposed models in the literature are mono-objective, where only one objective is optimized (coverage, cost, linear aggregation of objectives, etc.). In [7,10,29], only the objective related to the coverage of a relatively small area is optimized. In [26], the minimization of interferences is considered. In [2,39] a linear aggregation of objectives is used for dealing with the problem. Here, the cellular network designer has to specify the weights for the different objectives. Most of the existing studies are oriented toward small-scale micro-cellular or indoor systems involving a small number of antennas [17,27,33]. Moreover, other works use non-realistic simplified models of
The problem. In [1,40], a cell is supposed to have a specific shape (hexagonal topology) and then a propagation model is not used. Many search algorithms have been used for solving multiobjective combinatorial optimization problems [34,35]. Exact algorithms such as branch and bound [32] and dynamic programming [9] have been used to solve small instances of bi-objective problems. Population based metaheuristics such as Genetic Algorithms (GAs) have turned out to be of great efficiency to deal with multi-criteria optimization problems [11,14]. An important issue in designing efficient heuristics is related to the balance between the exploration of the search space and the exploitation of the obtained Pareto frontier. The design problem is a complex combinatorial problem, where a heuristic approach is required. Some metaheuristics have been suggested to deal with this problem [2,39]. They use a mono-objective model of the problem. In fact, they transform the multiobjective problem into a mono-objective one by combining the objectives in a linear aggregation way, or by using a goal programming approach or $\epsilon$-constraint approach. These approaches transform the structure of the problem and then its eventual properties may be lost. The problem is optimized with a model which is different from the initial one. The other drawback of those methods is that they need the specification of some parameters in the optimized model. For example, the linear aggregation model needs to specify the weights associated to each objective. Moreover, it generates only supported solutions, i.e., solutions in the convex hull of the Pareto front. For the $\epsilon$-constraint approach, we have to specify the coefficients associated to the constraints. For a detailed analysis of those methods, the reader can refer to [11,14,35].

In this paper, we present the framework DEMARNO for the approached Pareto multi-objective resolution by use of metaheuristics. It implements the specific part of the problem: specific coding scheme and genetic operators have been designed for the problem for which problem-specific knowledge is embedded. It is based on the generic platform ParadisEO for the flexible and easy design of metaheuristics. Basically, we designed a Steady-State Evolutionary Algorithm (EA), to approximate the Pareto frontier of the problem, by generating a set of non-dominated solutions. EAs are well suited to multiobjective optimization problems [12,16,13]. In order to represent an interesting set of solutions for the decision maker, solutions produced by the EA need to satisfy two conditions. They have to be good approximations of Pareto optimal solutions and uniformly distributed on the Pareto front. Solutions have to be good enough and well scattered. Advanced generic intensification and diversification search techniques for the efficient Pareto approach at resolution are provided by ParadisEO and have been investigated.

We exploit the models of parallelization and hybridization supported by ParadisEO too. Indeed, due to the complexity of the problem in terms of the number of solutions in the search space, and the time/space required to evaluate the objectives and the constraints, a parallel design of the proposed algorithm has been considered. On the other side, hybridization features allow the design of robust algorithms that cooperate, and so to significantly improve the quality of results.

The paper is organized as follows. In Section 2, the mobile network design problem is formulated as a multiobjective constrained Combinatorial Optimization Problem (COP). An analysis of the complexity of the problem is also given. In Section 3, we present the architecture and main features/components of a black-box framework for the problem, flexible both in the modeling of criterion and constraints and the solution methods. In Section 4, we present the solution method designed for the resolution. It basically consists in a Steady-State Evolutionary Algorithm. Several layers of models of parallelism and hybridization have been tested with the framework. For every one of them, the performance speed-up and/or the contribution in the improvement of robustness have been evaluated and quantified. Then, Section 5 presents and analyzes the results computed for the three tackled instances of networks. Finally, Section 6 contains some concluding remarks and perspectives of this work.

2. The network design

2.1. Modeling

In this part, the main concepts related to radio network design (e.g., cell, antenna, propagation model, etc) will be presented. Multiple antagonist criteria and associated constraints will be formulated. Then, an analysis of the complexity of the problem is given leading to an approximation of the number of candidate configurations on different instances. The modeling is based on second generation technologies GSM 900 and DCS 1800 but could yet be applied to one of the third-generation (3G) mobile phone technologies (e.g., UMTS). Indeed, the optimization techniques that are proposed, implemented and tested in this work are independent of the specification of objectives, constraints and engineering data. The reader is advised [28,31] for a more extensive analytical and technical description of the tackled problem.

2.1.1. Environmental and engineering data

The working area $\mathcal{P}$ is a geographical area, discretized in testing points. It defines the instance of the problem. It is a set of geographical information required for the design. A working area is described by a Digital Map Database. Four sets of points are identified in $\mathcal{P}$:

- A set of sites which are candidates for the positioning of BSs, $\mathcal{L} = \{L_i | i \in [1, \ldots, m]\}$, where $m$ represents the number of sites. Each site is defined by its coordinates $(x,y)$ and the height above the sea level $z$. 

Please cite this article as: E.-G. Talbi et al., Designing cellular networks using a parallel hybrid metaheuristic ..., Computer Communications (2006), doi:10.1016/j.comcom.2006.08.017
A set of Reception Test Points (RTP) in which the radio signal will be tested, \( R = \{ R_i \mid i \in [1, \ldots, l] \} \), where \( l \) represents the number of reception test points. Every RTP may be used as a signal test point to compute the coverage of the network [19].

A set of Service Test Points (STP) in which the expected service is tested, \( ST = \{ ST_i \mid i \in [1, \ldots, k] \} \), where \( k \) represents the number of service test points. Each STP defines the set of STP where the network must overcome a signal quality threshold to ensure a given Quality of Service (QoS). This threshold, \( S_q \), depends on the mobile terminal type. The STP points are used for defining the cell notion.

A set of Traffic Test Points (TTP) in which the expected amount of traffic will be tested, \( T = \{ TT_i \mid i \in [1, \ldots, n] \} \), where \( n \) represents the number of traffic test points. Each TTP is associated with the amount of traffic on this point \( e_i \) given in Erlang (unit of traffic measure). Depending on the global amount of traffic on its TTP, each BS will support a given number of Transmitters (TRX).

To ensure a good quality of signal of the area where the traffic is located, each TT is associated to a ST on the same coordinates. Moreover, it is necessary to know the radio signal on each ST, that is the reason why each ST, associated to a \( R_i \) the following inclusion condition, which is an additional characterization of the three sets of test points is always satisfied:

\[
T \subseteq ST \subseteq R
\]

Fig. 1 shows an example of a real working area and the inclusion between STPs and TTPs.

Let us define some engineering data and concepts necessary to the problem definition:

1. **Base station (BS):** this is a radio transmitter and receiver used for transmitting and receiving calls to or from mobile telephones in a particular cell. Antennas differ in their loss diagram which influences the cover area of the associated cell. We distinguish three types of antennas: omni-directional, small directive and large directive which differ by their loss diagram and gain (amplification).

The dimensioning of each antenna function of the supported traffic depends on the number of embedded radio transmitters/receivers (TRXs). Each BS can host at most 7 TRXs which corresponds to 43 Erlang of Effective Isotropic Radiated Power (EIRP) transmitted towards the receiver.

In this work, a site may be equipped with either one omni-directional antenna or one to three sectoral base stations. Main features of the antenna are: their type, the transmitting power, tilt and azimuth.

- **Mobile station:** it is a hand-held mobile radiotelephone for use in an area divided into small sections, each with its own short-range transmitter/receiver. Its main features are its receiving sensitivity and transmitting power.
- **Cell:** the logical cell associated to each BS is the set of STPs having this BS as the best server. Formally, the cell \( C_{jk} \) associated to the \( k^\text{th} \) antenna of the site \( L_j \) may be defined as follows:

\[
C_{jk} = \{ ST_i \mid C_{dijk} \geq S_q \} \quad \text{and} \quad C_{dijk} \geq C_{duv} \; \forall u \geq 0 \quad \text{and} \quad \forall v \geq 0
\]

where \( C_{dijk} \) denotes the field strength measured at point \( ST_i \) transmitted by the \( k^\text{th} \) BS of site \( L_j \), and \( S_q \) represents the service threshold of \( ST_i \). In practice, the shapes of cells are not homogeneous. They depend on both the topology of the geographic zone and the used wave propagation model which is defined below.

- **Wave propagation model:** obstacles and reflecting surfaces in the vicinity of the antenna have a substantial influence on the characteristics of the propagation path [25]. Moreover, the propagation characteristics change from place to place and, if the mobile unit moves, from time to time. Thus, the transmission path between the transmitter and the receiver can vary from simple direct line of sight to one that is severely obstructed by buildings, foliage and the terrain.

Many propagation models may be used such as free-space, Hocumura-Hata, and Walish-Ikegami (COST231) [24]. They differ in the information used for computing the field strength. The more interesting is the precision of a model, the more important its computing complexity [38,23]. The presented results in this paper are based on the free-space propagation model, which can be considered as a skeleton for other complex propagation models. Changing the propagation model will not affect the optimization model and the resolution techniques presented in this paper. Given that the other propagation models are more complex, using them will improve our results in term of speed-ups, because the ration between computing cost and communication cost (granularity) will be larger. Indeed, the communication cost is constant while the computing cost is more important.
2.1.2. Objectives and constraints

We are now going to define the objectives and constraints involved in the process of evaluation of a candidate network. Three main antagonist criterion have been identified in the efficient design of networks: cost, held traffic and quality.

- Cost: it deals with minimizing the number of sites, thus reducing the cost of the whole network. The cost associated to the installation of a site is relatively independent of the number of hosted base stations. Moreover, the cost of an equipped site is not related to its geographical position.

\[ \text{Min. } \sum_{j \in |\mathcal{J}|} y_j \]

with,

\[ y_j = \begin{cases} 1, & \text{if site } L_j \text{ is used,} \\ 0, & \text{else.} \end{cases} \]

- Held traffic: considering the set of TTPs, we are known an estimated amount of traffic to hold for every one of these points. In order a network to be efficient, held traffic should be maximized. Every cell is in charge of a subset of Service Test Points. Yet, the equipment considered in this work is limited regarding the number of transmitters a BS has (7 TRXs for 43 Erlang). This amount of traffic must be settled by a BS. At the network level, it consists in adapting the capacity of the network to the traffic demand. This objective may be expressed as follows.

\[ \text{Max. } \sum_{j,k} \left( \min \left( \sum_{i \in C_{jk}} e_i \right), 43 \right) \]

where \( e_i \) represents the traffic held at point \( TT_i \).

- Interferences: partially overlapping cells involve interferences decreasing the Quality of Service. The more interferences will be high, the more difficult will be the later frequency assignment.

Total interference is the sum of interferences computed at each STP of the geographical area. At each point, the BSs of the network producing the \( h^* \) most powerful fields are considered as potential servers and so useful. They are discarded in the computation of interferences. We admit that each STP does not require more than \( h \) BS considered as potential servers: the current Base Station and \( h - 1 \) handover BSs. The fields less that the \( h \) first ones are considered as useless and disrupting (Fig. 2). The third objective consists in minimizing the sum of interferences (field strengths) associated to non-handover BSs for every STP but greater than the threshold sensibility of the mobile station. For each point \( ST_i \), radio fields \( C_d \) can be sorted according to their strength.

\[ C_{d_{i1}} \geq C_{d_{i2}} \geq \cdots \geq C_{d_{ih}} \geq S_m \]

where \( S_m \) denotes the threshold sensibility of a receiver, \( h \) defines the number of potentially useful fields from handover servers.

The interference objective is formulated as follows:

\[ \text{Min. } \sum_{i \in |\mathcal{I}|} \sum_{j,k} (C_d - S_m) \]

Two constraints have to be satisfied to insure the validity of any candidate network: coverage and handover.

- Coverage: In order to insure the coverage of the geographical area, all the Service Test Points must receive at least one radio signal greater than the receiver sensibility threshold of the mobile terminal. Hence, the union of all cells must be equal to the set of STPs.

\[ \bigcup_{j,k} C_{jk} = \mathcal{S} \mathcal{F} \]

- Handover: the cellular network must be able to ensure the communication continuity from the starting cell to the target cell, when a mobile is moving toward a new cell (cell switching). Handover is a mechanism supplying this continuity. When the mobile is moving from one cell to another one, the starting cell drops out its communications with the mobile as soon as the target cell is able to ensure the communication with the incoming mobile device. This mechanism requires to manage overlap areas between cells. Every cell needs a non-empty handover area with neighboring cells in which the difference between the two most powerful received signals must be less than a certain threshold (fixed in this paper at 7 dB m).

For each point \( ST_i \), radio fields \( C_d \) can be sorted according to their strength.

\[ C_{d_{i1}} \geq C_{d_{i2}} \geq \cdots \geq C_{d_{ih}} \]

\[ \text{Threshold margin} \]

\[ C_{d_{i1}} \geq C_{d_{i2}} \geq \cdots \geq S_m \]

\[ \text{Handover fields} \]

\[ \text{Best server field} \]

Fig. 2. Signals are considered differently at each reception point: best signal, handover signals and interference.
Each cell must then satisfy:

\[ v_i < |\mathcal{F}|, \quad \| C_d_{\text{ref}1} - C_d_{\text{ref}2} \| \leq 7 \]

### 2.2. Decision space and complexity

#### 2.2.1. Instances

The realistic instances tackled in this work have been provided by France Telecom R&D: a highway area and two urban zones (Table 1). Every instance embeds the loss diagrams, the set of geographical positions for equipped sites, the locations of STPs, and the expected amount of traffic for TTPs.

First highway instance is characterized by a lower traffic, 250 candidate sites and 29,954 STP points. The first urban instance has a lower surface, less number of STPs, but is composed of a high number of candidate sites. Base stations should deal with a huge traffic in the town center and wide areas to cover in the outskirts. Third instance is a hard urban zone too, it stands out from the first ones by its size regarding both the number of STPs and TTPs.

#### 2.2.2. A practical hard Combinatorial Optimization Problem

The network design problem is NP-hard. A polynomial reduction of the problem to a NP-hard problem has been proposed in [30,39]. The main decision variable deals with the transmitting power, azimuth and tilt denoting the horizontal and vertical discretized engineering parameters (Table 2): the transmitting power from any STP to any site, it is independent of the configuration of the emitting BS. It may be huge, and so it could not be loaded entirely on a single workstation, but it significantly reduces the CPU time to evaluate a network. Moreover, the computing of the different objectives and constraints is also time-consuming. The evaluation of a candidate network depends on both the number of activated base stations and the static number of STP points.

Table 3 summarizes the mean CPU cost to process a candidate network according to the tackled instance and the chosen scenario. The evaluations have been performed on an AMD Opteron of 2.2 GHz.

### 3. A framework for radio network optimization

#### 3.1. Motivations

The work described here continues the thesis of Meunier [28]. A first software prototype has been implemented enabling not only the multi-criterion resolution of the problem but the evaluation of the formulated modeling of the network design problem too. The expert’s report proposed the extension of the modeling to new objectives and constraints. This brought the problem of extensibility of this first implementation regarding both the modeling of the problem and the solution methods applied to its resolution. On the other side, experimentations lead with the first prototype were limited as regarding the harnessed computational resources and the deployment at the same time of hierarchical parallel and hybrid models of metaheuristics that cooperate to give results of better quality.

---

Table 1

<table>
<thead>
<tr>
<th>Instance</th>
<th>Arno 1.0</th>
<th>Arno 3.0</th>
<th>Arno 3.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total traffic</td>
<td>3210.91</td>
<td>2988.12</td>
<td>8089.78</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Power</th>
<th>Azimuth</th>
<th>Tilt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notation</td>
<td>( P )</td>
<td>( \beta )</td>
<td>( \delta )</td>
</tr>
<tr>
<td>Domain</td>
<td>([26, 55])</td>
<td>([0^\circ, 360^\circ])</td>
<td>([-15^\circ, 0^\circ])</td>
</tr>
<tr>
<td>Discretization</td>
<td>2 dB m</td>
<td>10(^\circ)</td>
<td>3(^\circ)</td>
</tr>
<tr>
<td>Size</td>
<td>15</td>
<td>36</td>
<td>6</td>
</tr>
</tbody>
</table>

Therefore, the wish of the industrial partner was the integration of all the previous results in a generic framework for the parallel approached resolution of multi-objective problems: the ParadisEO framework was applied to the modeling and the resolution of the design of cellular networks. The result of this implementation is the ‘black-box’ DEMARNO software. The goal is the application and evaluation of different modeling schema and heuristics according to the needs and the impact of previously computed results, on benefits on the GSM and UMTS technologies.

In this section, we first present ParadisEO for the flexible design of metaheuristics applied to real and hard Combinatorial Optimization Problems (COPs) in several parallel and distributed contexts. Then, we describe the organization and architecture of the DEMARNO environment applied to radio network optimization. We justify some choices at the modeling step in order to ensure flexibility.

3.2. ParadisEO for the approached resolution of COPs

ParadisEO is a framework for the DEsign of Multi-Objective Algorithms applied to radio network optimization. We justify some choices at the modeling step in order to ensure flexibility.

In practice, combinatorial optimization problems are often NP-hard, CPU time-consuming, and evolve over time. Unlike exact methods, metaheuristics are general-purpose heuristics allowing to tackle large-size problems instances by delivering satisfactory solutions in a tractable time.

Metaheuristics can be classified in different ways, depending on some characteristics differentiating them [3]: nature-inspired vs. non-nature inspired, memory usage vs. memory-less methods, dynamic vs. static objective function, population-based vs. single point search, etc. Regards this last criterion, population-based metaheuristics are exploration oriented meaning they are efficient to find ‘good’ solutions in the search space. On the other side, solution-based metaheuristics have the power to intensify the search in promising regions.

Hybridization of metaheuristics balances between intensification and diversification. Best results for many high-dimensional instances of COPs have been found by hybrid algorithms [36].

Although metaheuristics allow to reduce the temporal complexity of their resolution, they are unsatisfactory to tackle large-scale COPs or industrial problems in which the processing of a single solution is costly. Parallel computing on computational grids has recently been revealed to be a powerful way to deal with very large time-intensive problems [18].

ParadisEO has been achieved in the context of the DOC-G French national project. It aims at building and evaluating the efficiency of frameworks for the resolution of hard and large instances of COPs that are either aca-
demic or taken from the real world. Such frameworks should harness, in an efficient and transparent manner, the huge computational power of grids.

ParadisEO is dedicated to the reusable design of parallel and hybrid metaheuristics. It provides a broad range of new features including local searches (Hill Climbing, Simulated Annealing and Tabu Search), the most common parallel models [37] (based on the walk, the solution and the objective function) and some hybridization mechanisms. ParadisEO is based on a clear conceptual separation of the solution methods from the problems they are intended to solve. This separation confers to the user a maximum code and design reuse.

Fig. 3 sketches the architecture of the project at the level of design and execution. We distinguish several complementary modules: Evolving Objects (EO) and Moving Objects (MO) that are, respectively, dedicated to the design of EAs and solution-based metaheuristics. Multi-objective Evolving Objects (MOEO) is related to multi-criterion optimization and embeds some features and techniques for the Pareto approach at resolution.

At the level of execution, ParadisEO relies on some underlying middlewares for the deployment on dedicated parallel and/or distributed architectures. ParadisEO has been recently extended to allow the efficient exploitation of large-scale Metacomputing grids of non-dedicated and heterogeneous workstations geographically distributed and networked in various ways [5].

The experiments lead on the modeling and the parallel resolution of real and hard problems from telecommunications, genomics, physical and chemical sciences. They give significant results, regarding both the quality of the computed solutions and the parallel efficiency on different architectures. The high content, utility and maintenance of ParadisEO encourages its use at worldwide level.

3.3. DEMARNO for multi-objective cellular network design

The aim was to design and implement a framework that is flexible regarding both the modeling of the problem (adding, removing, designing new objectives and constraints) and the design of advanced optimization methods applied to its resolution. This lead to the ‘black-box’ DEMARNO framework for cellular network design. It is ready to use and embeds some environmental and engineering features, the coding of solutions and some associated specific search operators, variation or neighborhood ones, that are, respectively, used by population of solution-based metaheuristics. DEMARNO bases on the ParadisEO framework (Fig. 4) which implements the invariant part of metaheuristics, hybridization and parallelization techniques.

3 Challenges in Combinatorial Optimization on Grids http://www.pris-m.upsq.fr/DOC-G/.

4 A framework for the DEsign of Multi-Objective Algorithms applied to Radio Network Optimization.
Below, we will present the specific encoding used, the genetic and neighborhood operators. The set of metrics for the multi-objective constrained evaluation of a given network have been embedded in the framework too.

- Encoding: a candidate solution of the network design needs to be encoded. It consists in a set of equipped sites hosting from one to three configured antennas. We use a multi-level hierarchical encoding: first level decides the activation of sites, second level the number and type of antennas. Then, third level handles the parameters of one base station.

- Variation operators: it deals with recombination and mutation operators. The classical genetic operators have been adapted to suit with the problem. We design the swapping crossover and the multi-level mutation. The swapping crossover works at the first level of the encoding hierarchy, exchanging sites that are located within a given radius around a randomly chosen site. Such a recombination operator is non-destructive: the offspring inherit good properties of the parents. The mutation acts at one random level of the hierarchy. It either acts on the activation toggling or power, azimuth and tilt tuning of BS.

- Neighborhood operators: we aim at designing local search techniques to be coupled with EAs in order to improve their robustness. We designed new operators to the exploration of the neighborhood of a given network. They are based either on the withdrawal of one base station, or the increase/decrease of one engineering parameter (azimuth, tilt, power). Exploring the neighborhood of one network involves the tuning of only one decision variable.

Such a minor change should not involve a full evaluation process of the new solution. So, we design new incremental operators of evaluation that can efficiently compute new objective and constraint values for the neighboring network. Efficiency of such operators will be measured and presented later.

4. A parallel hybrid metaheuristic

4.1. A multi-objective steady state genetic algorithm

Evolutionary algorithms have been widely used to solve multiobjective problems (MOP), as they are working on a population of solutions. The books edited by Deb [14] and Coello et al. [11] discuss general issues concerning design of EAs for MOP. A MOP consists in optimizing a vector of \( n \) objectives \( f(x) = (f_1(x), f_2(x), \ldots, f_d(x)) \) where \( x \) is a decision vector from the set \( C \) of feasible solutions. Two main issues have to be taken into account while solving MOP:

- Converge toward the Pareto frontier: most of the research about the application of genetic algorithms to MOP concentrates on the selection stage. At this stage, ranking methods are applied in order to assign a fitness to individuals. Our approach is Pareto in the
sense that the fitness depends only on the dominance notion, and thus, directly depends on the Pareto optimality.

A solution $u^* \in C$ is Pareto optimal if a solution $v \in C$ such that $F(v)$ dominates $F(u^*)$ does not exist. In maximizing $n$ objectives $f_i$, solution $u$ is said to be dominated by $v$ iff $\forall i \in [1, \ldots, n]$, $f_i(u) \leq f_i(v)$ and $\exists j \in [1, \ldots, n]$ such that $f_j(u) < f_j(v)$.

Find diversified solutions on the Pareto frontier: a sharing method able to maintain the diversity, using ecological niche induction, have been used to stabilize multiple subpopulations along the Pareto frontier. Unlike non-Pareto approaches where multiple executions are necessary to approximate the Pareto front, in a Pareto approach a single execution is sufficient.

The basic line of the algorithm is derived from a steady state genetic algorithm, where only one replacement occurs per generation. The steady state model of GAs is well adapted to an interactive decision making and a parallel asynchronous evaluation of the offspring. Initial solutions are randomly generated using a maximum number of activated sites according to expertise of engineers. Moreover, power of embedded antennas is initially low. The other decision variables (type of antenna, number of antennas, tilt, azimuth, etc.) are also generated in a uniform random manner. As a result, the initial population is well spread along the search space. The crossover and mutation operators are then applied. The crossover is applied to two selected individuals, generating two children. The resulting individual is integrated into the population if it is not Pareto dominated. The replacement phase replaces the worst individual of the current population in terms of a ranking procedure.

4.2. Ranking, sharing and elitism

Standard GA mechanisms need a revision in order to take into account several objectives. The multiobjective evaluation function becomes mono-objective, using ranking methods to sort the population according to the definition of Pareto dominance. The GA handles diversity using sharing, and elitism is used to speed-up the convergence process.

We use the ranking function NDS that is proposed by Fonseca and Fleming in [15]. An individual $i$ of the population, dominated by $k$ individuals, obtains the rank $k + 1$.

The ranking is based on the three objective functions.

The elitism method used consists in maintaining an archive population that will contain the Pareto solutions encountered during the search. This population will participate to the selection phase. To make a couple of networks to cross, the selection operator alternatively chooses one solution from the evolving population by applying the ranking and another random solution taken from the archive. At each generation, the archive is updated with the generated offspring. The non-dominated solutions from the union of the archive and offspring constitute the new archive.

To maintain diversity along the Pareto frontier and to avoid the archive to grow too much, we use a clustering/sharing technique that aims at spreading the population of individuals along this Pareto frontier by discarding the individuals that are strongly represented into the population. Sharing can be used in the decision space or in the objective space. We use the sharing method in the objective space to enforce diversity in the Pareto frontier. The reader is referred to [22] for a detailed presentation of the application of such diversification techniques in practice.

Table 4 summarizes all the features and strategies applied to the modeling of the basic Steady-State Evolutionary Algorithm.

4.3. Hybrid models

We propose two independent but complementary models of hybridization of metaheuristics: the cooperative island model of EAs and the coupling with Local Search techniques that aim, respectively, at diversifying the search along the Pareto frontier and so delaying convergence and aim at locally improving sets of non-dominated solutions in isolated regions.

4.3.1. The cooperative island model of EAs

The cooperative island model bases on the concurrent evolution of populations distributed in space. EAs exchange individuals between them, involving a delay in the global convergence and also the improvement of robustness and the obtaining of better results. It has been adapted to multi-criterion optimization problems. A part of the Pareto archive, which participates in the selection phase, is periodically communicated to the neighbors during the search.

Migrations are characterized by several parameters such as size, frequency and migration policy [8]. Many experiments and research works aimed at analyzing their mutual relationships for better efficiency. They depend on the landscape of the tackled problem, the size of the search space, the coding, etc. Due to the huge computation time required for one experiment, we did not tune those parameters and chose the most common parameters and values that generally permit the best improvement of performance. Islands are organized in a ring topology. The cooperation between islands is asynchronous and occurs after each cycle of 10
generations. A set of 10 randomly chosen networks (10% of the size of the pop.) are emigrated to the neighboring EA. When receiving some solutions, an island EA updates its local archive. Parameters are summarized in Table 5.

In this work, the quality of an approximative Pareto frontier is given by the Hyper-volume Metric calculation, also known as the standard S-metric, that estimates the volume of the non-dominated space [41]. To measure the impact of cooperation, we compare the obtained Pareto frontiers with or without migrations between the EAs. Fig. 5 applies the S-metric to the union of the Pareto archives of independent EAs and the union of those of cooperative EAs. Convergence is delayed in the cooperative EAs. Convergence is delayed in the cooperative EA, the size of the local archive dramatically increases as an exploration of the neighborhood is iteratively applied to any new PO solution found (Fig. 6). For a given network, the neighborhood reaches up to \(3 \times |\mathcal{D}| \times |\mathcal{P}_j| + |\mathcal{I}| + |\mathcal{P}| + 1\) solutions.

Faced to the first difficulty, we design and implement an incremental evaluation functor that efficiently computes the new fitness of a network as only one antenna has been added, removed or updated. It only requires to be known for the old network: the two best servers, the set of handover fields and the most powerful interference for any STP. Such information is not memory costly. Table 6 summarizes the mean CPU cost and complexity of the LS for a given solution regarding the three tackled instances. Then, to avoid the size of the local archive to grow too much, we fixed a maximum depth for the iterative search. Regarding the hybridization with the EA, we preferred the co-evolutionary mode: the set of EAs and LSs concurrently run and share the same elites of PO solutions. At the beginning of the optimization process, the set of EAs are powerful to quickly improve the archive of PO solutions than the LSs.

### Parameters setting the flow of migrations

<table>
<thead>
<tr>
<th>Criterion decision</th>
<th>Cyclic (10 gen.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection strategy</td>
<td>Random</td>
</tr>
<tr>
<td>Number of emigrants</td>
<td>10</td>
</tr>
<tr>
<td>Replacement strategy</td>
<td>Archive updating</td>
</tr>
<tr>
<td>Number of coop. EAs</td>
<td>4</td>
</tr>
<tr>
<td>Topology</td>
<td>Ring</td>
</tr>
</tbody>
</table>

4.3.2. Hybridization with Local Searches

EAs are exploration oriented: they are efficient to find good solutions scattered along the Pareto frontier. They are weaker to exploit regions including those solutions. We were interested in the hybridization between the Evolutionary Algorithm and a Local Search based on the exploration of the neighborhood of a solution. It is intended to carefully tune the configuration of base stations of a given network.

The Multi-objective LS we designed is iterative. It starts from a local archive of few networks selected in the Pareto frontier. For each activated base station of any network, it successively tests its removal, the updating of the power, azimuth and tilt with any of the values allowed in the domain. Then, the local archive is updated with the set of non-dominated neighbors and the LS restarts from any new solution being inserted in the local archive and so on.

In the deployment of such optimization process, we are faced to two main issues. First, the full evaluation of given solution regarding objectives and constraints is costly (Table 3). Then, the size of the local archive dramatically increases as an exploration of the neighborhood is iteratively applied to any new PO solution found (Fig. 6). For a given network, the neighborhood reaches up to \(3 \times |\mathcal{D}| \times |\mathcal{P}_j| + |\mathcal{I}| + |\mathcal{P}| + 1\) solutions.

![Fig. 5. Application of S-metric to the Pareto frontier found by cooperative/independent EAs (instance Arno 1.0).](image)

![Fig. 6. Iterative application of the LS starting from three random solutions (instance Arno 1.0).](image)

### Table 6
Some information characterizing the Local Search optimization for the three tackled instances (Opteron 2.2 GHz)

<table>
<thead>
<tr>
<th>Instance</th>
<th>Average number of neighboring solutions processed</th>
<th>Average number of PO solutions</th>
<th>Mean CPU time for the LS (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arno 1.0</td>
<td>10661.8</td>
<td>11.2976</td>
<td>511.763</td>
</tr>
<tr>
<td>Arno 3.0</td>
<td>16467.5</td>
<td>14.5591</td>
<td>461.44</td>
</tr>
<tr>
<td>Arno 3.1</td>
<td>33232.5</td>
<td>17.2353</td>
<td>2497.51</td>
</tr>
</tbody>
</table>
PO solutions found by LS, which require more CPU time, are likely be dominated and so withdrawn by new solutions computed by the EAs. Hence, the LSs restart from few new solutions taken from the evolving archive and so on. But the EAs will gradually be less and efficient as they converge to a set of good diversified solutions. Conversely, the LSs will be more and more powerful to exploit such already good solutions and compute new ones better (Fig. 7). Such a strategy enables a dynamic and automatic balance between exploration and intensification during the search.

Fig. 8 shows the cumulative CPU time allocated to the EA and the LS during the search. The more the optimization process progresses, the more the LS is applied instead of the EA. However, the EA remains active and so combines new solutions found the LS.

Table 7 summarizes miscellaneous parameters setting the design of the of the LS and its hybridization with the basic EA.

### 4.4. Parallel models

One of the main issues of the network design problem is the high computational cost consecutive to the processing of networks at different stages of the optimization process (Tables 3 and 6).

To get the results in a tractable time, we harnessed parallel and/or distributed platforms and deployed three parallel models:

- The parallel multi-start model of LSs: it consists in the concurrent execution of independent walks from starting solutions. Parallelism is said embarrassing.
- A parallel (a)synchronous evaluation model: the evaluation step of the EA is done in parallel. Those two first parallel models are independent of the network design problem.
- A parallel synchronous decomposition model: the evaluation of a single solution is carried out in parallel by partitioning the geographical area. This model is specific to the mobile network design problem.

In this paper, we are dealing with the relative speed-up where the parallel program was run on one processor. In this section, the results have been obtained on a Cluster Of Workstations (COW). The hardware platform is composed up to one hundred of AMD Opteron 2.2 GHz. with 4 GB RAM. The communication network is Gigabit Ethernet.

#### 4.4.1. The parallel multi-start model of LSs

The multi-start parallelization is based on the concurrent execution of independent Local Searches applied to starting solutions. No information is exchanged during the search. The same results would be obtained by sequentially running the walks one after the other.

#### 4.4.2. The parallel (a)synchronous evaluation model

The evaluation step of the offspring occurs at the end of each iteration of the EA. It succeeds to the transformation and consists in evaluating the fitness of every new solution. This computation is independent of the old generation. The parallelization is so obvious and bases on the distribution of the population.

We test the synchronous and asynchronous models. In the first implementation, all the solutions are scattered between evaluating nodes. Then, the computed fitness values are gathered by the EA which resumes its evolution.
Conversely, in the parallel Steady-State EA, the evaluation step is desynchronized from the other steps of the EA: the selection, transformation and replacement. New crossed solutions are enqueued. The EA does not wait for the return back of computed fitnesses but runs while the size of the queue does not reach a given threshold. Hence, a computing node never waits a task to do.

Measures of efficiency have been done for the instance Arno 1.0. For the synchronous model, we compute the speed-up at one iteration from the CPU time to evaluate a whole population, given a variable number of workers. Regards the asynchronous Steady-State EA, the speed-up is computed from the CPU time to reach a fixed number of performed evaluations. The speed-ups (Fig. 9) show a weak scalability of the synchronous model which is first explained by the variable cost of the evaluation. Indeed, the computation cost is dependent of the number of activated sites. During the search, the evolving population is composed of networks that are heterogeneous regarding the values of objectives. Hence, the parallel processing of such solutions is always irregular.

On the other part, this decreasing of performance can be understood as the number of individuals is constant and thus limited (e.g., 100). Therefore, as the number of computing nodes increases, there are less and less tasks scheduled to every one of them between two synchronization steps. On the contrary, the efficiency of the asynchronous model is high up to 100 workers.

4.4.3. The parallel synchronous decomposition model

In this model, the evaluation of a single solution is done in parallel. Its behavior follows the Master/Worker paradigm too. A same solution is shared between evaluating nodes that compute partial fitness values which are gathered and aggregated. The parallelization of the objective function is relevant as the processing of only one solution is CPU costly. It is generally used in conjunction with the previous model in order to improve the factor or scalability on large-scale platforms. The way to split the evaluation is specific to the tackled COP. For the network design, it is based on the partitioning of the geographical domain (operational space).

The efficiency of the parallel decomposition model depends on both the tackled instance and the number of cooperative partial evaluators (Fig. 10). Anyway, the measures of efficiency follow the logarithmic asymptote. We notice this fine-grained parallel model is efficient for a small number of decompositions. This model is not scalable but should be used with other hierarchical parallel models.
5. Results

For the optimization of cellular networks, we basically designed in DEMARNO a multi-objective Evolutionary Algorithm in which we embedded some generic techniques for the Pareto approach that aims at both intensifying and diversifying the search in the space of criterion. We designed different heuristics for the exploration of the neighborhood of a given network. Those algorithms aim at intensifying the search in a promising region. They are applied in a high-level coevolutionary hybridization with the EA from a set of diversified solutions taken from the elite.

We made use of parallelization techniques in order to distribute the computations and so reach the convergence in a tractable time, in spite of the combinatorial explosion and the costly processing of one network. Several parallel models have been used in conjunction in order to improve the scalability at execution on large-scale platforms. All are supported by the ParadisEO framework and have so been reused. They have been combined into a single hierarchical model (Fig. 11). At the lowest model, we find the parallel decomposition model. At the intermediate model, the parallel evaluation model is used. Finally, in the highest model, we have both the insular cooperative model and the parallel local searches that run independently. The relevance of the low-level parallel model (the decomposition of the evaluation) depends on both the number of computational resources and the grain of the process of evaluation. So, it has not been unconditionally used in the experimentations.

ParadisEO can either harness parallel/distributed architectures or metacomputing grids. The instances Arno 1.0 and Arno 3.0 have been solved on a dedicated cluster of 53 CPUs. The huge memory of such platform enables the use of precalculated data modeling the wave propagation and so decreases the CPU cost of evaluations and local searches. For those two instances, the
third and fine decomposition model has not been used. Tables 8 and 9 show the number of parallel tasks, dedicated to the evaluation or the exploitation of networks, done during the search. Some statistics are provided giving a general idea of the grain of such distributed tasks.

The two instances have been, respectively, solved after 44 and 39 h. Yet, the cumulative wall clock times of the work units performed by the workers are, respectively, 88 and 63 days.

Fig. 12 shows the convergence of the solutions representing the global archive found by the metaheuristic function of the number of generations done. Such convergence in the space of criterion is also quantified by the standard S-metric (Fig. 13) that measures the volume of the non-dominated space [41].

The third and hard instance Arno 3.1 has been solved on a metacomputing environment that utilizes idle time of workstations. In the engineering school Polytech-Lille, the CPU resources have been shown underused most of the time. Despite their low cost, such platforms are potentially very powerful. Table 10 summarizes some statistical results obtained at the end of the execution. The platform was composed of 100 non-dedicated workers. At any time, an average number of 88 workers were considered idle and so participate to our computations. ParadisEO is fault tolerant and automatically reschedules tasks as workers change of state. Moreover, application checkpointing peri-

Table 9
Numerical results for the execution of the parallel hybrid metaheuristic applied to instance Arno 3.0 on a dedicated cluster

<table>
<thead>
<tr>
<th>Platform</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Summary</strong></td>
<td></td>
</tr>
<tr>
<td>Number of proc.</td>
<td>53 × 2.2 GHz (AMD Opteron)</td>
</tr>
<tr>
<td>Wall clock time</td>
<td>39 h</td>
</tr>
<tr>
<td>Cumulative CPU time of workers</td>
<td>1596 h</td>
</tr>
<tr>
<td>Overall parallel performance</td>
<td>0.75</td>
</tr>
<tr>
<td><strong>Parallel evaluations</strong></td>
<td></td>
</tr>
<tr>
<td>Use of precalculated data</td>
<td>Yes</td>
</tr>
<tr>
<td>Parallel decomp. model</td>
<td>No</td>
</tr>
<tr>
<td>Number of tasks</td>
<td>856,859</td>
</tr>
<tr>
<td>Min. cost</td>
<td>0.31 s</td>
</tr>
<tr>
<td>Max. cost</td>
<td>11.34 s</td>
</tr>
<tr>
<td>Mean cost</td>
<td>1.14 s</td>
</tr>
<tr>
<td>Cumulative CPU time</td>
<td>270.43 h</td>
</tr>
<tr>
<td><strong>Parallel local searches</strong></td>
<td></td>
</tr>
<tr>
<td>Number of tasks</td>
<td>56,071</td>
</tr>
<tr>
<td>Min. cost</td>
<td>12.52 s</td>
</tr>
<tr>
<td>Max. cost</td>
<td>520.415 s</td>
</tr>
<tr>
<td>Mean cost</td>
<td>85.132 s</td>
</tr>
<tr>
<td>Cumulative CPU time</td>
<td>1325.95 h</td>
</tr>
</tbody>
</table>

Fig. 12. Graphical convergence of the elite of Pareto Optimal solutions found during the search function of the number of generations (Arno 1.0).

Fig. 13. Numerical convergence of the elite of Pareto Optimal solutions found during the search function of the number of generations (Arno 1.0). Efficiency bases on the S-metric.

Please cite this article as: E.-G. Talbi et al., Designing cellular networks using a parallel hybrid metaheuristic ..., Computer Communications (2006), doi:10.1016/j.comcom.2006.08.017
919 odically saves the state of the main running metaheuristic
920 so that it could be stopped and restarted later. The parallel
921 decomposition model has been used in conjunction with
922 the asynchronous parallelization of the evaluations. The
923 results have been obtained after 15 days of computation
924 but the cumulative computation time lasts more than 183
925 days.

6. Conclusion

927 In this paper, we have presented a multiobjective con-
928 strained combinatorial optimization model of a problem
929 involved by the radio network design in its full practical
930 complexity. Unlike mono-objective models, we find many
931 alternative solutions to the problem which can be
932 deployed. Those solutions represent the Pareto optimal
933 solutions of the problem. The obtained results and the
934 expertise of France Telecom engineers show that the model
935 may be improved. Indeed, some objectives and constraints
936 may be reformulated (interference, etc.) or added (homoge-
937 neity of the cells, cell connectivity, etc.). The DEMARNO
938 framework, which is ParadisEO dedicated to the network
939 design problem, allows flexibility of both the model and
940 the resolution algorithm and makes possible an efficient
941 resolution of future evolving models. The scalability of
942 the generic solution method, in term of the number of
943 objectives, will be evaluated. Other wave propagation mod-
944 els may also be added in the framework. We investigate the
945 design of UMTS systems in which capacity and signal qual-
946 ity must be taken into account during the planning phase.

Indeed in these systems, due to the cell breathing effect, the
947 actual shape of the cell depends on the traffic and explicit
948 QoS constraints must be introduced stating the effect of
949 intra-cell but also inter-cell interference.

<table>
<thead>
<tr>
<th>Table 10</th>
<th>Numerical results for the execution of the parallel hybrid metaheuristic applied to instance Arno 3.1 on a metacomputing platform of non-dedicated and heterogeneous workstations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platform</td>
<td>Metacomputing</td>
</tr>
<tr>
<td>Number of proc. 100 (heterogeneous and non-dedicated: from 0.4 to 3.2 GHz)</td>
<td></td>
</tr>
<tr>
<td>Wall clock time Almost 15 days</td>
<td></td>
</tr>
<tr>
<td>Cumulative CPU time of workers 30,861 h</td>
<td></td>
</tr>
<tr>
<td>Average number of active workers 88</td>
<td></td>
</tr>
<tr>
<td>Overall parallel performance 0.98</td>
<td></td>
</tr>
<tr>
<td>Use of precalculated data No</td>
<td></td>
</tr>
<tr>
<td>Parallel decomp. model Yes (x2)</td>
<td></td>
</tr>
<tr>
<td>Number of tasks 1897,046</td>
<td></td>
</tr>
<tr>
<td>Min. cost 7.815 s</td>
<td></td>
</tr>
<tr>
<td>Max. cost 247.967 s</td>
<td></td>
</tr>
<tr>
<td>Mean cost 21.51 s</td>
<td></td>
</tr>
<tr>
<td>Cumulative CPU time 11,340 h</td>
<td></td>
</tr>
<tr>
<td>Number of tasks 15,299</td>
<td></td>
</tr>
<tr>
<td>Min. cost 1832.57 s</td>
<td></td>
</tr>
<tr>
<td>Max. cost 24,380.7 s</td>
<td></td>
</tr>
<tr>
<td>Mean cost 4593.52 s</td>
<td></td>
</tr>
<tr>
<td>Cumulative CPU time 19,521 h</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 14. Highway instance (Arno 1.0). A random cellular network taken from the archive of solutions after the convergence of the metaheuristic has been reached.
Fig. 15. Urban instance (Arno 3.0). A random cellular network taken from the archive of solutions after the convergence of the metaheuristic has been reached.

Fig. 16. Urban instance (Arno 3.1). A random cellular network taken from the archive of solutions after the convergence of the metaheuristic has been reached.

Figs. 14–16 illustrate the decomposition in cells of one optimized network taken from the final archive (instances Arno 1.0, Arno 3.0 and Arno 3.1).

With ParadisEO, we have designed in DEMARNO a parallel multiobjective evolutionary algorithm in order to obtain a set of efficient solutions spread on the Pareto frontier. We have shown that the proposed algorithm is able to handle multiple objectives and constraints, and to create and maintain a set of solutions diversified on the Pareto frontier. The results obtained on different realistic problems show the importance of the different proposed search mechanisms. The proposed intensification and diversification search mechanisms have also been validated on well known generic multiobjective optimization problems such as the flow-shop scheduling problem and the vehicle routing problem. This work led not only to the efficient resolution of the problem, but a better understanding of the structure of the problem which makes the improvement of the model.

The complexity of the problem in terms of the number of solutions in the search space, the evaluation of the different objectives and constraints, and memory requirement is very important. Therefore, we have proposed three hierarchical parallel models of the algorithm to improve the quality and the robustness of the obtained Pareto front, to speed-up the search and to solve large instances of the problem. The experimental results obtained using ParadisEO show that the proposed parallel hierarchical approach can handle real problem instances in an efficient way on different parallel and distributed execution environments.

This study opens other directions for future works. It would be interesting to investigate the design of an interactive decision making algorithm. The final decision maker may exploits the Pareto frontier to guide in a progressive manner the algorithm toward the final solution. On the other side, we are now working on the problem of design and dimensioning of the fixed access network of mobile telecommunications systems of second and third generation (GSM and UMTS). It follows the step of design of cellular networks and consists in constructing a tree connecting the BTSs to the BSC (root of the tree) with a set of transmission link types. The main objective of this problem is to minimize the cost of the infrastructure while not decreasing average the availability of the network.

References


