Genetic Optimization for Optimum 3G Network Planning: an Agent-Based Parallel Implementation

Alessandra Esposito, Luciano Tarricone, Stefano Luceri, Marco Zappatore
Dept. Innovation Eng., University of Salento
Via Monteroni
73100 Lecce - Italy

***Abstract*- The continuous evolution of wireless networks, as well as the attention paid by the public opinion to human exposure to electromagnetic fields radiated by basestation antennas, render the development of software tools to support optimum design and planning of 3G networks highly desirable. Though many tools are already available, open problems are still on the table. One key issue is represented by optimization methods adopted to solve the problem of identifying optimum locations and electrical parameters for basestation antennas. In this paper, the recent technology of software agents is adopted, in conjunction with genetic algorithms and parallel computing, in order to perform effective and efficient optimization of 3G networks. Results demonstrate the appeal of such a strategy, tested on standard real cases. Impressive results are achieved for both the accuracy and the performance attained, with the use of low-cost computing platforms and freeware tools.**

I. INTRODUCTION

The design of efficient third-generation (3G) wireless networks is a complex task, deserving the convergence of several different skills in areas such as telecommunication systems, radiopropagation, information systems. Moreover, the optimum design of 3G networks must take into account both constraints coming out from quality of service, coverage, etc., and issues related with human exposure to electromagnetic (EM) fields generated by basestation (BS) antennas. Indeed, safety standards cast some limitations to field levels, and the public opinion is usually extremely sensitive to the possibility of controlling (or, better, reducing) human exposure to EM fields.

In such a many-folded scenario, the setting up of software tools and models to support optimum design and planning of 3G networks is highly desirable. A crucial role is played by optimization methods adopted to solve the problem of identifying optimum locations and electrical parameters for BS antennas. The problem is computationally intensive and must be attacked by using methods able to guarantee global search.

As a matter of fact, Genetic Algorithms (GAs) have proven to be an attractive optimization technique in a wide range of applications, including the optimum planning of wireless networks [1]-[4]. These works demonstrated the need to scale GAs towards parallel computing environments, in order to attack large real-life problems, as well as the importance of performing this at affordable costs.

In this paper, a very recent information technology, *Software Agents*, is adopted in conjunction with parallel (*grid*) computing in order to set up a very efficient and effective genetic optimizer, suitable to solve the problem of optimum planning of 3G wireless networks.

The paper is structured as follows. Section II proposes the formulation of the optimization problem of 3G network design. Section III recalls basic concepts on GAs, on their parallel implementation and their application in the specific problem. Section IV introduces agents and their adoption in the specific application. Section V describes the agent based-architecture and framework. Results are given in Section VI, proving the efficiency and effectiveness of the method on real cases. Finally, conclusions are drawn.

II. THIRD GENERATION WIRELESS SYSTEM PLANNING

Planning tools for wireless networks are becoming essential for offering a high-quality service network with appropriate use of resources and minimal EM exposure.

To build a planning tool, different aspects have to be accounted for, such as geographical data management, antenna technical data storage, EM fields estimation and network parameter optimization [5].

Among these aspects, one of the most crucial is the appropriate evaluation of the EM field levels generated by network antennas. Several radiopropagation methods can be used to estimate the mean intensity value of the received signal in a specific point. Such methods evaluate the attenuation as the transmitted-to-received power ratio, accounting also for possible wave interactions with the obstacles between transmitter and receiver. The EM simulator employed in this paper adopts Free Space Loss (FSL), the empiric COST231-Okumura-Hata and the semi-empiric COST231-Walfisch-Ikegami model [6]. Obviously, these approaches, are not suitable to address situations of near-field exposure and suffer from a limited accuracy especially in very complex urban scenarios. In such a case the potential user could identify areas of uncertainty in order to adopt dedicated strategies and models [7] on demand.

The other important module of a planning tool is the optimization one. It optimizes BS locations and antenna parameters in order to design high-quality networks, with reduced EM emissions. The network parameter optimization requires the formulation of the problem, specifying the objective function and the constraints, and the choice of the solution method (as described in Section III). The optimum 3G network planning problem (3GNPP) can be formalized as follows: *once the geographical area to be covered has been*

T. Sobh et al. (eds.), *Novel Algorithms and Techniques in Telecommunications and Networking,*
DOI 10.1007/978-90-481-3662-9_32, © Springer Science+Business Media B.V. 2010

discretized into points where the traffic demand and the EM levels are estimated, given a set of possible BS positions and antenna parameters range, find the best network configuration to minimize a function of the EM field values and to satisfy EM exposure, handover and downlink capacity constraints.

For the sake of brevity, we address the reader to a recent paper [4] for a complete analysis of the variables, constraints and objective functions concerning the problem. We just recall now some relevant issues.

The area is discretized by means of two different set of points. *SDP*: set of Demand Points (DPs), where the total traffic demand is partitioned. *STP*: set of Test points (TPs), where the EM field levels are estimated.

Concerning network parameters, each BS antenna (sector) is characterized by variable parameters such as azimuth, tilt, height-above-ground, total emitted power and few fixed parameters, such as gain, frequency and radiation pattern. In this work we choose to fix these parameters in order to limit the computational cost of optimization, and mainly because, in real network configurations, only a very limited range of antenna types is used.

The BS locations are grouped into subsets, and a maximum of one BS can be activated for each group. A network configuration is defined when each BS is located and all the parameters of each BS sector are set up. The network configuration feasibility is enforced by constraints that provide a realistic representation of a 3G cellular network and compliance with upper limits of EM field values on the TPs.

The specific target of 3GNPP is the achievement of the required quality of service (traffic coverage in this case), while pursuing a policy of control of the human exposure to EM fields (according to safety limits existing in several countries). Therefore, the objective function must take into account the EM levels detected at the TPs. Several scalar functions can be implemented to achieve this goal, i.e.

- the minimization of the sum over the total E field levels in STP (*MinSum*) to provide a global decreasing of the total observed level;
- the minimization of the maximum E field level in the area (*MinMax*) to prevent the formation of peak values;
- the minimization of the difference among the total E field levels in two TP subsets with the highest and lowest values (*MinDiff*) to lead toward solutions with a relatively smooth field distribution.

These objective functions are not monotone with respect to all the variable parameters: their behavior when a BS is moved or if sector tilt or power level are changed is not trivially predictable. In this work only downlink capacity and handover have been considered [8]-[10]. Downlink capacity is expected to be particularly relevant in the presence of asymmetrical communication and it gives us relevant indications about the amount of traffic that can be covered [11]. Other objective functions and models taking into account more network parameters (as accepted users, effective load, etc. [12]) and uplink direction [11] could be considered but this is outside of the scope of this work.

III. The Parallel Genetic Algorithm

GA is an iterative optimization method that exploits the analogies with genetic processes, in particular natural selection and heredity principles. The EM community is rather familiar with GA approach, and we address the interested reader to [13] for a tutorial introduction to GAs and to [14] for a more specific application to planning problems of wireless networks.

The strength of GAs lies on their robustness and large applicability to different classes of problems since they are capable of performing an efficient search also when the apriori knowledge of the problem is limited.

However, in some problems, the fitness evaluation and/or the feasibility check of each individual of the population can be very time consuming. The consequences are either the slowing down of the entire process or, fixing a reasonable execution time, a shallow exploration of the solution space. The necessity to overcome these limitations and the high modularity of the GA has led toward parallel GA (PGA) implementations.

The parallel approach can be applied to a GA in different ways, depending on how the population is distributed and on how information is shared among parallel instances. A basic classification of PGA has been reported in [15]. In the present paper, taking advantage from the strategies proposed in some recent papers [16][17], we propose an island-based PGA to solve the 3GNPP.

Our implementation of island-based PGA consists of a predefined number of sequential GA threads (instances), each processing a portion of the entire population.

The chromosome migration mechanism follows an adaptive ring topology, thus ensuring ring continuity in case of one or more threads fail or prematurely terminate their execution.

IV. Software Agents and PGA

In this section, software agents, and their amenability for an efficient implementation of PGA, are discussed. Software agents are autonomous entities capable of "flexible, autonomous action in their environment in order to meet their design objectives" [17]. In simpler terms, they are computational entities capable of autonomously taking initiative and of communicating with their peers to pursue a goal. Agent-based genetic optimizers have been implemented by several authors [18]-[20]. In most cases, a multi-agent framework is implemented, where each agent carries its own genetic material and interacts with other agents to reach the global optimization goal. This is also the case of the unique (at our best knowledge) work in the computational electromagnetic field [19]. A different approach is proposed here: an *island-based* model, obtained by embedding existing serial code into software agents. The software agent paradigm was chosen for this model for two major reasons:

1) agents intelligence can be used to manage global search behaviour (by controlling topology and migration parameters);

2) agents are considered the most suited programming paradigm for parallel applications in computational grids (CG) [20], CG being a very low-cost parallel computing environment (see Fig. 1).

Indeed, software agents have several features making them amenable to CG environments. First of all, they can migrate during the execution from one host to another in a network. This is particularly useful in dynamic and unstable environments, such as CGs, where load on computing resources may change enormously during execution time.

Moreover, agents can be dynamically created and destroyed and the computing application is highly transparent with respect to the hardware platform, the number of computers and the configuration of the computer network. This renders agent-based systems very flexible with respect to other libraries commonly used in CGs, such as MPICH-G2, which needs the previous installation and compilation of executables at each node.

On the other hand, performance of distributed agent-based applications can be low as communication between agents generally consumes more bandwidth than other communication models (such as MPI) [22].

Luckily, our problem requires very limited data exchange. Indeed chromosomes migration is sporadic with respect to frequency of generations. Moreover, messages that allow the dynamic adjustment of communication path and framework administration (e.g. keep-alive signal) require a very low bandwidth. Furthermore, since the time needed for exchanging messages between agents is hidden by the time needed for the generation of new chromosomes, the system is expected not to suffer too much from the enhanced communication burden of agent-based paradigm.

V. PGA ARCHITECTURE

The parallel genetic application framework is depicted in Fig. 2, showing a master-worker architecture.

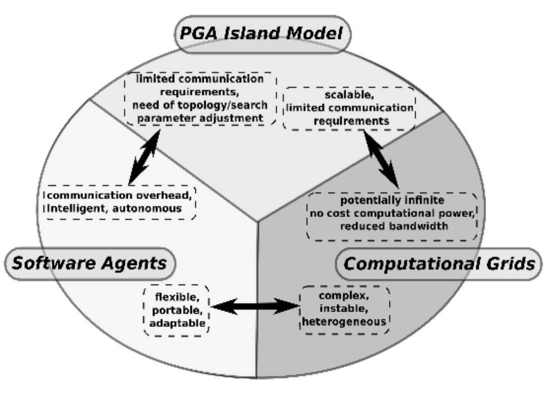

Fig. 1. PGA enabling technologies: the choice for agent-based paradigm is reinforced by suitability of computational grids to support parallel problems with limited communication needs

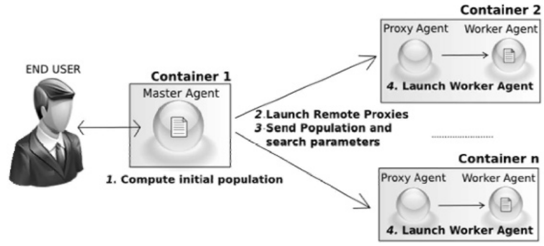

Fig. 2. The PGA framework is based on a master-worker model. The master is responsible for initializing the search, collecting outputs and monitoring search evolution. Worker agents perform searches in islands. Each worker is associated to a proxy agent, that maintains the communication with the master during the search

The *master* agent, apart from being the interface to the end user, is in charge of initializing the environment and of managing and controlling the execution of the framework itself. It launches the worker agents, collects their outputs and ranks them. It is also responsible of adjusting parameters related to search status, topology, and environment conditions change (node fault, unbalanced load, etc.).

The *worker* agents are obtained by embedding GA native C code into a Java method. They carry out a GA search in the sub-population they have been assigned, and exchange individuals at every generation producing an improvement (or after a predefined number of iterations if no improvement is obtained). Each worker agent is associated with a special agent, namely the *proxy*, having the responsibility of managing the interaction between the worker and the rest of the world (master or other workers) during the search. Proxies perform the following actions: forward chromosomes to other proxies and viceversa; inform the master agent about the current status of the associated worker; inform the worker about changes on the execution parameters.

The master/proxy/worker structure allows one to distinguish between local searches (carried out by worker agents) and global search management (carried out by the master which takes into account information provided by the proxies). Substantially, two communication paths are followed (see Fig. 3). The former is performed by workers and proxies and is finalized to perform the exchange of individuals: each time a worker wants to send individuals to other workers, it sends the chromosomes to its proxy, which is in charge of sending the individuals to the correct destination (according to the current network topology). The latter is pursued by the master and the proxies, and is finalized to the monitoring and control of the overall search.

This configuration is flexible and adaptable, featuring the key requirements needed to cope with the instability and unbalanceness of CG environments and providing (transparently to end-user) the dynamic adjustment of the communication path, (which may be due to network/node failure or, in the most common case, to search completion by one of the worker agents).

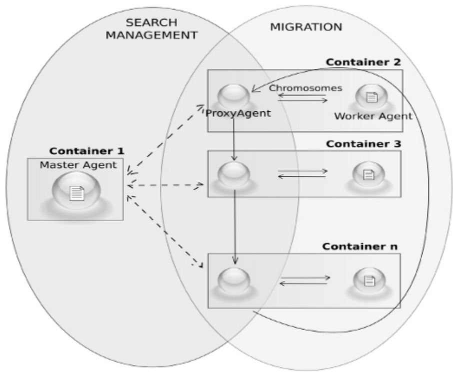

Fig. 3. The two agents communication paths (dashed and continuous lines), ensure asynchronous search management and migration of chromosomes

Software agents need a suited execution environment able to support communication, localization and migration. The de facto standard platform is JADE [22], adopted in our experimentation, which follows the official standard for communication named FIPA [23] and provides a robust framework for hosting agent-based systems.

VI. RESULTS

The PGA applied to 3GNPP has been tested using data from The Hague city scenario prepared by the IST-2000-28088 project MOMENTUM [24]. The considered area, a mix of urban and suburban zones, is 4 Km x 4 Km wide with 76 possible BS sites, each with a different height range. We have divided these potential sites into subgroups of 4 locations and have considered 16 possible values for emitted power (between 0 and 30 W) and sector azimuth, 8 alternative choices for height (each range associated to the relative BS location) and for mechanical tilt (between 0 and 8°) and sector azimuth. The parameters involved in the optimization process have been codified as explained in Fig. 4, using binary strings (chromosomes) to store their numerical values.

All services are considered: speech telephony, file download, Location-based Services, streaming multimedia, video telephony, web surfing, email, MMS. The corresponding traffic data are provided through average and busy hour traffic for each service and traffic snapshots. A *snapshot* is a photo of the traffic demand in a given instant, represented by a set of points, each using a specific service.

In the reported trials, the traffic associated with a snapshot is subdivided among 400 DPs, counting how many snapshot points are in the cell grid of a DP and reporting the different services to speech telephony (according to Tab. I [25]).

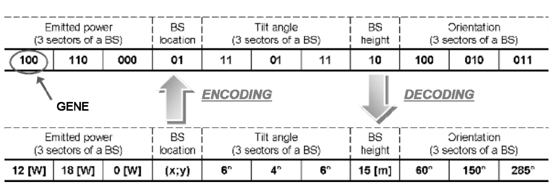

Fig. 4. A bitstring excerpt and the corresponding encoded values.

TABLE I
CONVERSION FOR THE DIFFERENT SERVICES WITH RESPECT TO SPEECH SERVICE

Traffic Type	Effective Downlink Capacity
Speech telephony	**1.0**
File download	6.9
Location base services (LBS)	1.0
Streaming multimedia	5.7
Video telephony	9.3
Web sharing	3.1
Email	3.6
MMS	3.6

In the following tests, (*MinSum*) has been chosen as objective function and, without loss of generality, the STP has been taken coincident with SDP. The used radiopropagation method is COST231-Okumura-Hata for urban case.

Tests have been performed within a network of 8 computers connected by a TCP/IP-based 100Mb/s network. Computer characteristics are
- CPU: P-IV 3.0Ghz - RAM: 1GB;
- CPU: AMD Athlon 1.6Ghz - RAM: 256MB

running under Linux and Windows operating systems.

The platform on which agents reside has to be up and running before the developed PGA framework execution. It includes one main-container and 7 remote containers (one on each host) connected to it. The framework starts once the master agent is launched in one of the available containers. Then the master agent, according to input data given by the user, copies all files and libraries on the other containers and orders the framework to start all agents needed for the simulation. It can be noticed that no previous installation of software is required apart from JADE.

Three instances with a different number of individuals of the global population (400, 800, and 1600 individuals) have been considered. For each of them the following trials have been tested:

- 1 sequential GA with the entire population;
- 2 GA threads running on half the whole population with and without chromosome migration;
- 4 GA threads running on 1/4th of the whole population with and without chromosome migration;
- 8 GA threads running on 1/8th of the whole population with and without chromosome migration.

Table II displays the best objective function value and the wall clock time of each trial with a randomly generated initial population for the 800 individuals instance. In details, the table columns represent: *Seed*, the seed for the random number generator used for the initial population; *n_host*, the number of used hosts (*islands*); *migr_size*, number of chromosomes to migrate; *n_pop:* number of individuals of each subpopulation; *Best*, objective function value of the best resulting solution and *Time*, wall clock time required for the entire execution, including preprocessing and postprocessing phases.

Concerning the migration case, a series of trials with different *migr_size* values have been carried out. The results (not reported here for the sake of brevity) show that in almost every case better results are reached when *migr_size* is the 10%

of the island population. This is the *migr_size* value used in trials described in Figures 5 and 6.

Trials executed with a randomly generated initial populations for the 400 and 1600 individuals instances (not reported here) show similar trends in objective function values and wall clock time. On the basis of data in Tables II, we can observe that (in agreement with [15]) the parallel implementation with or without migration leads in almost every trial to better results than the sequential case. Moreover, Fig. 5, shows that: 1) when exchange of individuals is permitted, the framework succeeds in getting better solutions than in the no-migration case, and in this specific case 2) the solution improves as the host number increases. This behavior is coherent with the island-based population idea: a set of smaller distributed subpopulations with individuals moving from one group to another generates better individuals than a single large static population.

As described in Sec. V, sequential GA is given by the execution of the original C implementation, while PGA consists of the original C code embedded inside a Java agent. Looking at the wall clock time column in Tab. I (also displayed in Fig. 6 for the 800 individuals case) and comparing the execution time of the sequential GA with the execution time of the two-hosts PGA, the overhead due to this kind of implementation can be noticed. The impact of such an overhead is more and more smoothed when more hosts are added. Indeed, a substantially linear trend with respect to the host number is featured by the PGA instances. Such a linear trend is justified by the following considerations:

- preprocessing time (generation of the entire population and distribution of individuals to the hosts) and postprocessing time (sorting of the solutions from the single GAs) is similar for 2, 4, and 8 hosts;

TABLE II
RESULTS FOR THE 800 INDIVIDUALS INSTANCE

Seed	n_host	migr_size	n_pop	Best	Time(s)
1	1	0	800	0.81519	9420
1	2	0	400	0.84803	7465
1	4	0	200	0.77555	4109
1	8	0	100	0.80308	2215
1	2	40	400	0.76244	7589
1	4	20	200	0.71829	4197
1	8	10	100	**0.66089**	2322
2	1	0	800	0.79037	9628
2	2	0	400	0.82898	7423
2	4	0	200	0.83449	4052
2	8	0	100	0.75772	2224
2	2	40	400	0.76561	7701
2	4	20	200	0.67473	4501
2	8	10	100	**0.66542**	2299
3	1	0	800	0.80542	9551
3	2	0	400	0.76923	7457
3	4	0	200	0.80294	4052
3	8	0	100	0.77710	2238
3	2	40	400	0.73768	7641
3	4	20	200	0.67838	4312
3	8	10	100	**0.63350**	2238

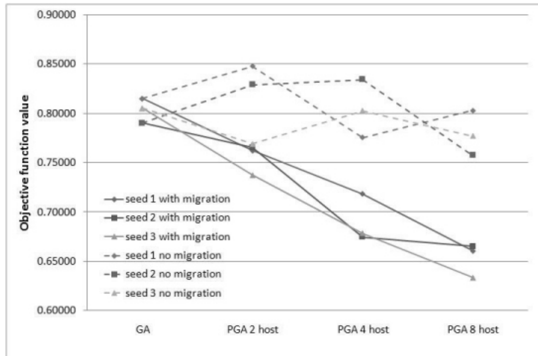

Fig. 5. Migration (solid lines) vs. non-migration (dashed-lines) case for three different initial population for the 800 individuals instance

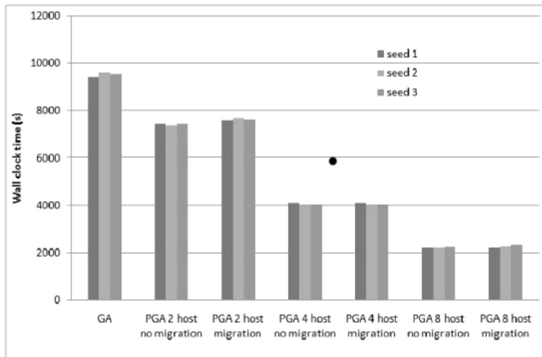

Fig. 6. Wall clock times in seconds of the 800 individuals instance.

- the *migr_size* value and the migration frequency do not influence the wall clock time, thanks to the asynchronous solution exchange mechanism.

Concerning the last issue, a series of tests executed by modifying the migration frequency (ranging from every 20 to every single generation) returns practically the same wall clock time, proving that the solution exchange time (carried out by the Java agent) is hidden by the time needed for the generation of new chromosomes (carried out by the C genetic algorithm implementation).

In conclusion, we observe that:

- agent-based GA is efficiently parallelized;
- the use of agents peculiar features is crucial to achieve high quality solutions in a very reduced time (linear speed-up is attained in a very low-cost local grid).

From the network planning point of view, since it is not straightforward to deduce the meaning of the best objective function values, a comparison between the average traffic distribution in The Hague and the EM distribution of two of the obtained solutions has been reported (see Fig. 7). The EM levels are represented in *dBμM/V*. As it can be seen, the resulting solution shows a good match among the BS positions and the areas with higher traffic values.

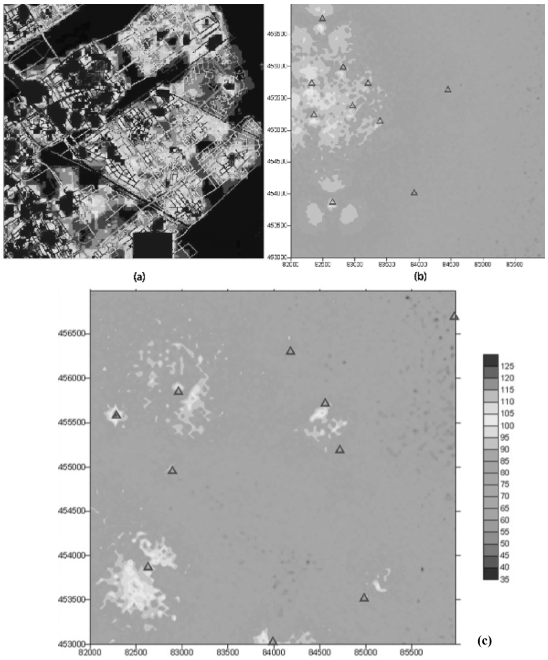

Fig. 7. Average traffic distribution in The Hague city scenario (a) vs. EM field distribution of two resulting solutions with different objective function values ((b) 1.2481), (c) 0.066542). Markers identify BS location.

Therefore, the optimum planning has produced a high quality network configuration with low EM impact (the maximum EM values are around 120 $dB\mu M/V$ = 0.316 V/m). While guaranteeing a high quality of service.

VII. Conclusions

In this paper the problem of optimizing locations and electrical parameters of BS antennas in a 3G wireless network has been attacked and solved by using GAs, in conjunction with the technology of software agents and adopting parallel computing strategies.

The adoption of suitable algorithmic choices, namely gene migration, rendered extremely natural and efficient by agents, has produced impressive effectiveness and high-performance in the solution of real cases. Quasi-linear speed-ups are achieved, with a high scalability of the application.

Moreover, software agents are intrinsically open to computational grids, thus paving the way to the adoption of very cost-effective computing platforms, as well as to the setting up of open and scalable software tools for 3G network optimum planning.

References

[1] Pantoja, M.; Bretones, A. and Martin, R.: "Benchmark Antenna Problems for Evolutionary Optimization Algorithms," *IEEE Trans. on Antennas and Propagation*, vol. 55, no. 4, pp. 1111–1121, Apr. 2007.

[2] Brunetta, L; Di Chiara, B; Mori, F; Nonato, M.; Sorrentino, R.; Strappini, M. Tarricone, L.: "Optimization Approaches for Wireless Network Planning," in *Proc. 2004 URSI EMTS, Intl. Symp. on Electromagnetic Theory*, Pisa (Italy), May 2004, pp. 182–184.

[3] Di Chiara, B.; Nonato, M.; Strappini, M.; Tarricone, L. Zappatore, M.: "Hybrid Meta-heuristic Methods in Parallel Environments for 3G Network Planning," in *Proc. EMC Europe Workhop 2005, EM Comp. of wireless Systems*, Rome (Italy), Sept. 2005, pp. 199–202.

[4] Crainic, T.; Di Chiara, B.; Nonato, M. and Tarricone, L., "Tackling Electrosmog in Completely Configured 3G Networks by Parallel Cooperative Meta-heuristics," *IEEE Wireless Comm., Special Issue 3G/4G/WLAN/WMAN Plann. and Optim.*, vol.13, no.6, pp.34–41,2006

[5] Angelucci, M.; Di Chiara, B.; Sorrentino, R.; Strappini, M. and Tarricone, L.: "Genetic Optimization of Radiobase-Station Sizing and Location using a GIS-based Framework: Experimental Validation," in *Proc. IEEE AP-S Intl. Symp. and USNC/CNC/URSI National Radio Science Meeting*, Columbus (Ohio), June 2003, ISBN 0-7803-7847-4.

[6] Saunders, S. R.; *Antennas and Propagation for Wireless Communication Systems*. New York: John Wiley & Sons Ltd, 1999.

[7] Catarinucci, L.; Palazzari, P. and Tarricone, L.: "Human Exposure to the Near Field of Radiobase Antennas - a Full-Wave Solution Using Parallel FDTD," *IEEE Trans. on MW Theory and Tech.*, vol. 51, no. 3, pp. 935–940, 2003.

[8] Jamaa, S. B.; Altman, Z.; Picard, J. M. and Fourestie, B.: "Combined Coverage and Capacity Optimisation for UMTS Networks," in *Proc. Telecommunications Network Strategy and Planning Symp. NETWORKS 2004, 11th International*, 2004, pp. 175 – 178.

[9] Maple, C.; Guo, L. and Zhang, J.: "Parallel Genetic Algorithms for Third Generation Mobile Network Planning," in *Proc. Intl. Conf. on Parallel Computing in Electrical Engineering*, 2004, pp. 229–236.

[10] Amaldi, E.; Capone, A. and Malucelli, F: "Base Station Configuration and Location Problems in UMTS Networks," in *Proc. 9th Intl. Conf. on Telecommunication Systems, Modelling and Analysis*, 2001.

[11] Amaldi, E.; Capone, A.; Malucelli, F. and Signore, F.: "Optimization Models And Algorithms For Downlink UMTS Radio Planning," in *Proc. IEEE Wireless Comm. and Networking*, vol.2,2003, pp.827–831.

[12] Golberg, D. E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc, 1992.

[13] S. Datta, Aragon, A and Saunders, S. R.: "Automatic optimisation for UMTS indoor coverage using distributed antenna systems," in *4th IEE Intl. Conf. on 3G Mobile Comm. Tech.*, vol. 494, 2003, pp. 42–47.

[14] Han, J.; Park, B.; Choi, Y. and Park, H.: "Genetic Approach with a New Representation Base Station Placement in Mobile Communications," in *Proc. 54th IEEE Conf. on Vehicular Technology*, vol. 4, 2001, pp. 2703–2707.

[15] Cantú-Paz, E.: "A Survey of Parallel Genetic Algorithms," University of Illinois at Urbana-Champaign, Tech. Rep., 1997.

[16] Calegari, P.; Guidec, F.; Kuonen, P. and Kobler, D.: "Parallel Island-based Genetic Algorithm for Radio Network Design," *Journal of Parallel and Distributed Computing, Special issue on parallel evolutionary computing*, vol. 47, no. 1, pp. 86–90, Nov. 1997.

[17] Wooldridge, M.: "Agent-based software engineering," *IEEE Proc. Software Engineering*, vol. 144, pp. 26–37, 1997.

[18] Cardon, A.; Galinho, T. and Vacher, J.: "Genetic Algorithms Using Multi-Objectives in a Multi-Agent System," *Robotics and Autonomous Systems*, vol. 33, pp. 179–190, 2000.

[19] Lymperopoulos, D.; Tsitsas, N. L. and Kaklamani, D. I.: "A Distributed Intelligent Agent Platform for Genetic Optimization in CEM: Applications in a Quasi-Point Matching Method," *IEEE Trans. on Antennas and Propagation*, vol. 55, no. 3, pp. 619–628, 2007.

[20] Foster, I.: "Brain Meets Brawn: Why Grid and Agents Need Each Other," in *AAMAS'04*, New York, USA, July 2004.

[21] Lymperopoulos D. *et al.*: "Software Agents for Parametric Computational Electromagnetic Applications," in *Advances in Information Technologies for Electromagnetics*, A.Esposito and L.Tarricone, Eds. Springer, 2006, pp. 345–379.

[22] Jade web site. [Online]. Available: http://jade.cselt.it

[23] Fipa web site. [Online].Available: http://fipa.org/repository/index.html

[24] Ist momentum web site. [Online]. Available: http://momentum.zib.de

[25] Eisenblätter, A.; Fügenschuh, A.; Fledderus, E. R.; Geerdes, H. F.; Heideck, B.; Junglas, D.; Koch, T.; Kürner, T. Martin, A.: "Mathematical Methods for Automatic Optimization of UMTS Radio Networks," IST-2000-28088 MOMENTUM, Tech. Rep. D4.3, 2003.