Solving the Location Areas Scheme in Realistic Networks by Using a Multi-objective Algorithm

Víctor Berrocal-Plaza, Miguel A. Vega-Rodríguez, Juan M. Sánchez-Pérez, and Juan A. Gómez-Pulido

Dept. Technologies of Computers & Communications, University of Extremadura
Escuela Politécnica, Campus Universitario S/N, 10003, Cáceres, Spain
{vicberpla,mavega,sanperez,jangomez}@unex.es

Abstract. The optimization of the management tasks in current mobile networks is an interesting research field due to the exponential increase in the number of mobile subscribers. In this paper, we study two of the most important management tasks of the Public Land Mobile Networks: the location update and the paging, since these two procedures are used by the mobile network to locate and track the Mobile Stations. There are several strategies to manage the location update and the paging, but we focus on the Location Areas scheme with a two-cycle sequential paging, a strategy widely applied in current mobile networks. This scheme can be formulated as a multi-objective optimization problem with two objective functions: minimize the number of location updates and minimize the number of paging messages. In previous works, this multi-objective problem was solved with single-objective optimization algorithms by means of the linear aggregation of the objective functions. In order to avoid the drawbacks related to the linear aggregation, we propose an adaptation of the Non-dominated Sorting Genetic Algorithm II to solve the Location Areas Planning Problem. Furthermore, with the aim of studying a realistic mobile network, we apply our algorithm to a scenario located in the San Francisco Bay (USA). Results show that our algorithm outperforms the algorithms proposed by other authors, as well as the advantages of a multi-objective approach.

Keywords: Location Areas Planning Problem, Non-dominated Sorting Genetic Algorithm II, Mobile Location Management, Multi-objective Optimization, Stanford University Mobile Activity Traces.

1 Introduction

The Public Land Mobile Networks (PLMNs) are the networks that provide mobile communications to the public. These networks can be represented by a hierarchical structure with three levels: Mobile Subnet, Radio Network, and Core Network [1]. The first level (Mobile Subnet) is constituted by the subscriber terminals (or Mobile Stations, MS). The second level (Radio Network or Access Subnet) divides the coverage area in several smallest regions, known as cells, among which the radio-electric resources are distributed and reused. This level is constituted by the Base

A.I. Esparcia-Alcázar et al. (Eds.): EvoApplications 2013, LNCS 7835, pp. 72–81, 2013.

Stations (BSs) and the Base Station Controllers (BSCs). A BS is the network entity that provides mobile access to the MSs, and a BSC is the network entity that performs control tasks associated with several BS. And the third level (Core Network) comprises all systems and registries that perform the management, control and operation tasks, e.g. the Mobile Switching Centers (MSCs), the Home Location Registries (HLRs), and the Visitor Location Registries (VLRs) are the entities responsible for managing the subscriber location update (LU) and paging (PA). These management tasks (LU and PA) are used by the PLMNs to know the exact location of the subscriber (in terms of cells) in order to automatically route an incoming call to him, so they are considered as two of the most important tasks in the PLMNs [2].

By using the LU procedure, the MS notifies the mobile network that its location should be updated. There are several strategies to manage the LU, mainly classified into two groups: static LU and dynamic LU [3-5]. In static strategies, all subscribers perceive the same logical topology of the mobile network, which is calculated by the Location Management system (the system that controls the location update and the paging). However, in dynamic strategies, each MS might perceive a different logical topology and it decides to perform a location update according to its calling and mobility patterns. With these last strategies, the signaling load associated with the LU and PA procedures might be reduced considerably, but they require higher network capabilities than the static strategies, since the mobile network has to store a logical configuration per subscriber. That is why the static strategies are more popular than the dynamic ones. Always Update, Never Update, and Location Areas are examples of static LU strategies [3]. In the Always Update strategy, the MS updates its location whenever it moves to a new cell, and hence, the paging is only performed in the current cell of the callee user. In contrast to the Always Update strategy, there is not location update in the Never Update strategy, and thus, all network cells have to be polled in order to know the location of the callee user. The Location Areas (LA) strategy is halfway between the Always Update and the Never Update strategies. By means of the Location Areas scheme, the network cells are grouped into logical areas such that the MS only updates its location when it moves to a new logical area, and the paging is only performed in the cells within the logical area of the callee user.

On the other hand, the mobile network uses the PA procedure to know the exact cell associated with the callee user [4]. Commonly, this procedure has to be performed before the timer expires (known as maximum paging delay). Two-cycle Sequential Paging, Blanket Polling, and Shortest Distance are examples of paging procedures with delay constraint. In the Two-cycle Sequential Paging, the cells that have to be polled are grouped into two paging areas that are polled sequentially. First, the system checks the last cell known for the callee user, and then, if it is necessary, the rest of cells of that location area. This is one of the PA procedures more used in mobile networks. Fig. 1 shows an example of the LU and PA procedures.

In the last decade, due to the exponential increase in the number of mobile subscribers, several authors have focused their researches on applying new optimization techniques to the Location Management tasks, and concretely in the Location Areas scheme. P. R. L. Gondim is one of the first authors arguing that the Location Areas Planning Problem is an NP-hard combinatorial optimization problem (due to the huge size of the objective space), and he defined a Genetic Algorithm (GA) for finding quasi-optimal configurations of Location Areas [6]. P. Demestichas et al. proposed three algorithms (GA, Simulated Annealing (SA), and Taboo Search (TS)) to research the Location Areas scheme in different environments [7]. J. Taheri and A. Zomaya implemented several algorithms to solve the Location Areas Planning Problem: Hopfield Neural Networks (HNNs), GA, SA, and combinations of GA with HNN (GA-HNN) [8-11]. R. Subrata and A. Zomaya proposed a dynamic LU strategy based on the Location Areas scheme [12]. In their work, R. Subrata and A. Zomaya use the network provided by the Stanford University (SUMATRA [13]: Stanford University Mobile Activity TRAces, a trace generator that is well-validated against real world data measured in the geographical area of the San Francisco Bay, USA). Finally, S. M. Almeida-Luz et al. developed the Differential Evolution (DE) algorithm [14-15] and the Scatter Search (SS) algorithm [16-17] to also solve the SUMATRA network [13].

In all of these algorithms, the objective functions of the Location Areas Planning Problem are linearly combined into a single objective function. This technique reduces the complexity of the optimization algorithm but has got associated several drawbacks, as can be seen in Section 3. That is why we propose an adaptation of the Non-dominated Sorting Genetic Algorithm II (NSGA-II, a Multi-Objective Evolutionary Algorithm) to solve the Location Areas Planning Problem. With a multi-objective optimization algorithm, we can obtain a set of solutions among which the network operator could select the one that best fits the network real state, i.e. when the signaling load generated by other management systems is considered.

The paper is organized as follows: Section 2 defines the Location Areas Planning Problem and presents the SUMATRA network that we have studied. Section 3 shows the main features of a multi-objective optimization algorithm and our adaptation of NSGA-II to solve the Location Areas Planning Problem. Results, comparisons with other authors, and an analysis of the obtained solutions are presented in Section 4. Finally, our conclusions and future work are discussed in Section 5.

2 Location Areas Scheme

The Location Areas scheme is being widely used in current mobile networks. By means of this strategy, the network cells are grouped into logical areas (or Location Areas, LAs) with the aim of reducing the number of signaling messages associated with the subscriber location update, since the MS is free to move inside a given LA without updating its location. Furthermore, the paging procedure is only performed in the cells within the current LA of the callee user. It should be noted that the main challenge of the Location Areas scheme is to find the LA configuration that minimizes simultaneously the number of location update and the number of paging messages.

To reduce the number of location updates, the size of the LAs should be increased, leading to an increment in the number of paging messages because more cells have to be paged. And vice versa, to reduce the number of paging messages, the size of the

LAs should be reduced, leading to an increment in the number of location updates. Therefore, the Location Areas scheme defines a Multi-objective Optimization Problem (MOP) that can be formulated as:

$$\boldsymbol{f}_1 = \min\left\{ \mathrm{LU}_{\cos t} = \sum_{t=T_{ini}}^{T_{fin}} \sum_{i=1}^{N_{user}} \gamma_{t,i} \right\}, \tag{1}$$

$$\boldsymbol{f}_2 = \min\left\{ \mathrm{PA}_{\cos t} = \sum_{t=T_{ini}}^{T_{fin}} \sum_{i=1}^{N_{user}} \rho_{t,i} \left(\alpha_{t,i} + \left(1-\alpha_{t,i}\right) \times \mathrm{NA}[\mathrm{LA}_t[i]]\right) \right\}, \tag{2}$$

subject to

$$\sum_{i=1}^{N_{Cell}} \sum_{k=1}^{N_{Area}} \mu_{i,k} = 1, \tag{3}$$

where the involved variables are:

- $\gamma_{t,i}$: A binary variable that is equal to 1 when the MS *i* moves to a new Location Area in the time *t*; otherwise $\gamma_{t,i}$ is equal to 0.
- $\rho_{t,i}$: A binary variable that is equal to 1 when the MS *i* has an incoming call in the time *t*; otherwise $\rho_{t,i}$ is equal to 0.
- $\alpha_{t,i}$: A binary variable that is equal to 1 when, in the time *t*, the MS *i* is located in its last updated cell; otherwise $\alpha_{t,i}$ is equal to 0.
- NA: Vector that stores the size (in terms of number of cells) of each Location Area.
- LA_t: Vector that stores the Location Area associated with each user in the time *t*.
- $\mu_{i,k}$: A binary variable that is equal to 1 when the cell *i* belongs to the Location Area *k*; otherwise $\mu_{i,k}$ is equal to 0.
- N_{user}: Number of mobile users.
- N_{Cell}: Number of network cells.
- N_{Area}: Number of Location Areas.
- [T_{ini}, T_{fin}]: Time interval during which the LU and PA costs are calculated.

Equation (1) defines the first objective function of the Location Areas Planning Problem: minimize the number of location updates. Equation (2) shows the second objective function: minimize the number of paging messages required to locate a callee user by using the Two-cycle Sequential Paging. In this work, we use the paging procedure proposed in [12, 15, 17]: firstly, the callee MS is searched in the last updated cell and, if the MS is not found in this cell, the other cells of the Location Area are simultaneously polled in order to know the exact cell in which the MS is located. Constraint (3) establishes that a cell cannot be associated with several Location Areas, and has to be associated always with a Location Area. Therefore, the maximum Location Area size is limited to NCell (i.e. the Never Update strategy, when all cells belong to the same Location Area), and the maximum number of Location Areas is also limited to NCell (i.e. the Always Update strategy, when each cell belongs to a different Location Area).

In previous works, the Location Areas Planning Problem was solved by means of Single-objective Optimization Algorithms (SOA). For it, the linear aggregation of the objective functions was used to adapt this multi-objective optimization problem to a single-objective optimization problem. Equation (4) shows the objective function proposed in [8-12, 14-17], where β is a constant defined to consider that the cost associated with a location update is higher than the cost of performing the paging procedure, since the location update involves more network entities than the paging. Commonly, this coefficient is configured equal to 10, $\beta = 10$.

$$\boldsymbol{f}^{SOA} = \min\left\{\mathrm{Cost}_{tot}(\beta) = \beta \times \mathrm{LU}_{cost} + \mathrm{PA}_{cost}\right\}, \tag{4}$$

In this paper, with the aim of avoiding the drawbacks associated with the linear aggregation (see Section 3), we propose a multi-objective optimization algorithm to solve the Location Areas Planning Problem. With this strategy, the network operator could select the solution that best adjusts to the network real state.

2.1 Stanford University Mobile Activity Traces

The Stanford University Mobile Activity TRAces (SUMATRA) is a set of mobile activity traces that are available to the public via web [13]. In this work, we have studied the BALI-2 network, which provides a real-time data of the users' mobile activity measured in the San Francisco Bay (USA) during 24 hours. BALI-2 defines a mobile network with 90 cells and 66,550 subscribers. The main appeal of this network is that it will allow us to study the behavior of the Location Areas Scheme in a realistic mobile network, since its trace is well validated against real world data. Fig. 2 shows an approximation of the BSs planning and its associated graph. Note that each circle represents a BS and the edges represent the neighborhood among cells (i.e. two cells are neighboring if they are connected by an edge).

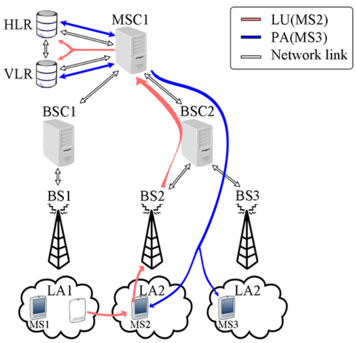

Fig. 1. Example of location update and paging in the Location Areas scheme

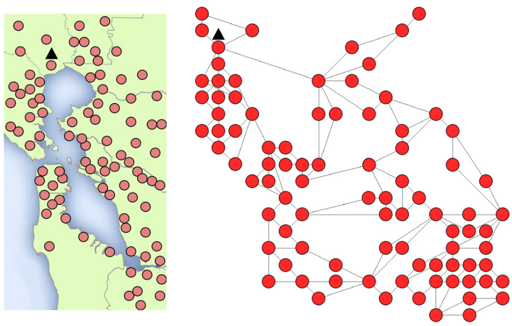

Fig. 2. BALI-2 mobile network: BS planning and associated graph

3 Multi-objective Optimization Paradigm

A Multi-objective Optimization Problem (MOP) is the optimization problem in which two or more conflicting objective functions have to be optimized under certain constraints [18], e.g. the Location Areas Planning Problem. Due to the fact that there are two or more conflicting objectives, the main task of a multi-objective optimization algorithm consists in finding a set of solutions, each one related to a trade-off among objectives. Commonly, these similar-quality solutions are called non-dominated solutions and they are grouped in the known as Pareto front. For definition, the solution $\mathbf{x}^1$ is said to dominate the solution $\mathbf{x}^2$ (denoted by $\mathbf{x}^1 \prec \mathbf{x}^2$) when $\mathbf{x}^1$ outperforms $\mathbf{x}^2$ in one or more objectives, and $\mathbf{x}^1$equals $\mathbf{x}^2$ in the rest of objectives.

Traditionally, the linear aggregation of the objective functions was used to solve a MOP by means of single-objective optimization algorithms, e.g. see Equation (4). This strategy allows the use of less complex algorithms but has got associated several drawbacks. Firstly, an accurate knowledge of the problem is required in order to properly configure the values of the weight coefficients, which could be real values. Secondly, the proper value of the weight coefficients might vary in the time (e.g. in the Location Areas Planning Problem, different states of the signaling network could require different values of β). And thirdly, a single-objective optimization algorithm must perform an independent run for each combination of the weight coefficients.

In this paper, we have implemented an adaptation of the Non-dominated Sorting Genetic Algorithm II to solve the Location Areas Planning Problem. A detailed explanation of the adaptation of this Multi-Objective Evolutionary Algorithm (MOEA) is discussed in Subsection 3.1.

3.1 Non-dominated Sorting Genetic Algorithm II

The Non-dominated Sorting Genetic Algorithm II (NSGA-II) is a population-based metaheuristic algorithm in which the EVolutionary OPerators (EVOPs: recombination of parents or crossover, mutation, and natural selection of the fittest individuals) of biological systems are used to iteratively improve a set of solutions. This algorithm was proposed by K. Deb et al. in [19] with the goal of reducing the computational complexity of its predecessor NSGA [20]. The pseudo-code of our adaptation of NSGA-II is presented in Algorithm 1. In this pseudo-code: Npop is the population size, P_C is the crossover probability, P_M is the mutation probability, and the stop condition is the Maximum Number of Cycles (MNC) executed.

Firstly, the population should be initialized and evaluated, i.e. a solution must be calculated and evaluated for each individual. In this work, every individual (its genome) is represented by a vector that stores the Location Area associated with each network cell. And secondly, the EVOPs are applied iteratively until the stop condition is not satisfied. The crossover operation is used to generate the offspring population, where each offspring has genetic information of two parents. We have used an elitist crossover, in which the number of crossover points and their positions are randomly determined between [1, 4] and [0, NCell-1], respectively. The mutation operation is applied to change the genetic information of the offspring. We have defined two mutation operations: Gene Mutation (GM) and Merge-LA Mutation (MLAM). The Gene Mutation consists in changing the Location Area of a boundary cell by its neighboring Location Area of lower size (in terms of number of cells). And the Merge-LA Mutation consists in merging the smallest Location Area with its neighboring Location Area with fewer cells. These two mutation operations cannot be applied simultaneously over the same individual, and they are configured such that $2P_{GM} = 2P_{MLAM} = P_M$, where P_{GM} is the Gene Mutation probability and P_{MLAM} is the Merge-LA Mutation probability. Finally, the natural selection is performed with the goal of selecting the Npop fittest individuals, which will be the parent population in the next cycle. K. Deb et al. [19] provide a methodology to select the best solutions (or individuals) in the multi-objective context, i.e. they provide a multi-objective fitness function. This methodology consists of two functions: the Non-dominated Sorting and the Crowding Distance. The Non-dominated Sorting is a function that applies the non-dominance concept to arrange a set of solutions in groups (or fronts). And the Crowding Distance measures the density of solutions surrounding a particular point of the objective space. For detailed information about NSGA-II, please see [19].

In order to perform a fair comparison with other works, we use the same stop condition (MNC=1000) and the same population size (Npop=300) [15]. The other parameters of NSGA-II have been configured by means of a parametric study, in which we have performed 30 independent runs per experiment. The parameter configuration that provides the higher Hypervolume is: $P_C = 0.9$, $P_M = 0.2$. The Hypervolume is one of the most popular multi-objective indicators, and it is used to know the quality of a multi-objective optimization algorithm. This indicator measures the area of the objective space covered by the Pareto front (if we assume two objectives).

Algorithm 1. Pseudo-code of our adaptation of NSGA-II

```
% Initialization of the population
Indv ← Initialization(Npop);
% Evaluation of the population
Indv ← ObjectiveFunctionsEvaluation(Indv);
Indv ← MOFitnessFunctionEv(Indv);
% Main loop
while (stop condition ≠ TRUE){
% Crossover or recombination of parents
Offsp ← Crossover(Indv,P_C);
% Mutation of the offspring
Offsp ← Mutation(Offsp,P_M);
% Evaluation of the offspring
Offsp ← ObjectiveFunctionsEvaluation(Offsp);
% Evaluation of all population
[Indv,Offsp] ← MOFitnessFunctionEv(Indv,Offsp);
% Selection of the fittest individuals
Indv ← NaturalSelection(Indv,Offsp);
}
```

4 Experimental Results

With the purpose of checking the behavior of our algorithm in a realistic mobile network, we have studied one of the networks developed by the Stanford University: BALI-2, the network that provides real-time information of the mobile activity. Furthermore, in order to verify the quality of our solutions, we have compared our results with those obtained by other authors. Due to the fact that there is no other work that addresses the Location Areas Planning Problem in a multi-objective manner, we must compare with single-objective optimization algorithms [15, 17]. For it, we have to find in our Pareto front the solution that best fits the Equation (4) with β equal to 10, since it is the objective function used by these single-objective optimization algorithms. In addition, we have calculated statistical data of the Hypervolume (median, HV_{median}, and interquartile range, HV_{iqr}) for 30 independent runs. These statistical data are: $HV_{median}(\%) = 93.75\%$, $HV_{iqr} = 7.8700e^{-4}$. It should be noted the low value of the HV_{iqr}, this denotes that our algorithm is very stable. Fig. 4 shows the Pareto front related to the median HV, the reference points that we have used to calculate the HV (these reference points have been calculated by means of the extreme solutions: Always Update strategy and Never Update strategy, see Section 2), and the solution of our Pareto front that best fits the Equation (4) with β equal to 10. In this figure, we can observe the high spread of our Pareto front, which includes the two extreme solutions. Fig. 3-(a) presents a comparison among the Always Update strategy, the Never Update strategy, and our proposal. This figure shows us that our algorithm clearly outperforms the two classic strategies. Finally, Fig. 3-(b) shows a comparison between our algorithm (NSGA-II) and the algorithms proposed in [15] and [17]: Differential Evolution (DE) and Scatter Search (SS). In this figure we can see that our algorithm outperforms these two single-objective optimization algorithms in the hours with higher mobile activity (8:00h-20:00h). A summary of these two last figures is presented in Table 1, in which we can observe that our algorithm also obtains better results than one of the dynamic strategies proposed in [12]: Distance-Based Location Area (DBLA). Therefore, we can conclude that our algorithm is competitive, since it achieves better results than other algorithms and it provides a set of solutions among which the network operator could select the one that best adjusts to the network real state.

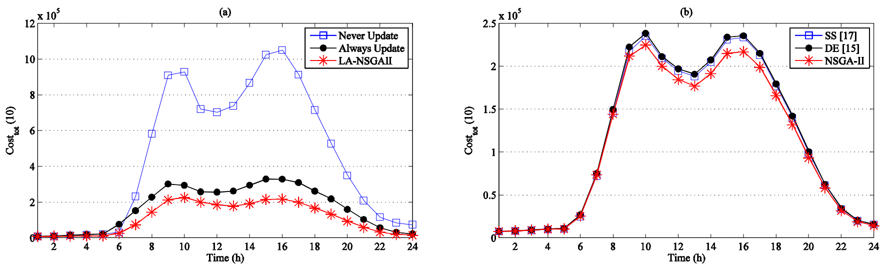

Fig. 3. BALI-2: (a) Comparison among strategies. (b) Comparison among algorithms.

Table 1. Comparison among strategies

Algorithm	$\sum_{t=0h}^{24h} \mathrm{Cost}_{tot}(10)$
LA-NSGA-II	2,619,519
LA-SS[17]	2,756,836
LA-DE[15]	2,799,289
Always Update	4,010,317
Never Update	10,836,953
DBLA[12]	2,695,282

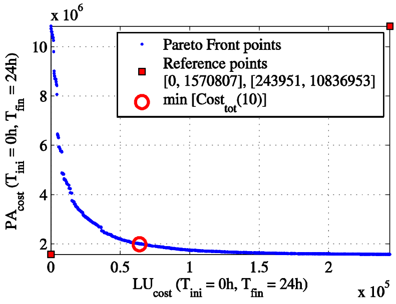

Fig. 4. Median Pareto front of NSGA-II

5 Conclusions and Future Work

The main contribution of this work is that we have adapted the Non-dominated Sorting Genetic Algorithm II (NSGA-II, a Multi-Objective Evolutionary Algorithm) to solve the Location Areas Planning Problem in a realistic mobile network: BALI-2 [13]. Due to the fact that there is no other work that addresses this problem with a multi-objective optimization algorithm, we must compare with single-objective optimization algorithms. Results show that our algorithm is promising, since it achieves better solutions than the algorithms proposed by other authors, and it gives more vision to the network operator. In a future work, it would be interesting to develop other multi-objective optimization algorithms and compare them with NSGA-II. Furthermore, it would be a good challenge to study the Location Areas Planning Problem when the network architecture is considered.

Acknowledgement. The work of Víctor Berrocal-Plaza has been developed under the Grant FPU-AP2010-5841 from the Spanish Government. This work was partially funded by the Spanish Ministry of Economy and Competitiveness and the ERDF (European Regional Development Fund), under the contract TIN2012-30685 (BIO project).

References

1. Agrawal, D.P., Zeng, Q.A.: Introduction to Wireless and Mobile Systems, 3rd edn. Cengage Learning, Stamdford (2010)
2. Garg, V.: Wireless Communications and Networking, 1st edn. Elsevier, San Francisco (2007)
3. Tabbane, S.: Location management methods for third-generation mobile systems. IEEE Commun. Mag. 35(8), 72–84 (1997)
4. Wong, V.W.-S., Leung, V.C.M.: Location management for next generation personal communications networks. IEEE Network 14(5), 18–24 (2000)

5. Kyamakya, K., Jobmann, K.: Location management in cellular networks: classification of the most important paradigms, realistic simulation framework, and relative performance analysis. IEEE Transactions on Vehicular Technology 54(2), 687–708 (2005)
6. Gondim, P.: Genetic algorithms and the location area partitioning problem in cellular networks. In: Proceedings of IEEE 46th Vehicular Technology Conference on Mobile Technology for the Human Race, vol. 3, pp. 1835–1838 (1996)
7. Demestichas, P., Georgantas, N., Tzifa, E., Demesticha, V., Striki, M., Kilanioti, M., Theologou, M.: Computationally efficient algorithms for location area planning in future cellular systems. Computer Communications 23(13), 1263–1280 (2000)
8. Taheri, J., Zomaya, A.: The use of a Hopfield neural network in solving the mobility management problem. In: Proceedings of The IEEE/ACS International Conference on Pervasive Services, pp. 141–150. IEEE Computer Society, Washington (2004)
9. Taheri, J., Zomaya, A.: A genetic algorithm for finding optimal location area configurations for mobility management. In: Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary, pp. 568–577. IEEE Computer Society, Washington (2005)
10. Taheri, J., Zomaya, A.: A Simulated Annealing approach for mobile location management. In: Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, pp. 194–201. IEEE Computer Society, Washington (2005)
11. Taheri, J., Zomaya, A.: A combined genetic-neural algorithm for mobility management. Journal of Mathematical Modeling and Algorithms 6(3), 481–507 (2007)
12. Subrata, R., Zomaya, A.Y.: Dynamic Location Area Scheme for Location Management. Telecommunication Systems 22(1-4), 169–187 (2003)
13. Stanford University Mobile Activity TRAces (SUMATRA), `http://infolab.stanford.edu/sumatra` (accessed in 2012)
14. Almeida-Luz, S.M., Vega-Rodríguez, M.A., Gómez-Pulido, J.A., Sánchez-Pérez, J.M.: Differential evolution for solving the mobile location management. Applied Soft Computing 11(1), 410–427 (2011)
15. Almeida-Luz, S.M., Vega-Rodríguez, M.A., Gómez-Pulido, J.A., Sánchez-Pérez, J.M.: Applying Differential Evolution to a Realistic Location Area Problem Using SUMATRA. In: Proceedings of The Second International Conference on Advanced Engineering Computing and Applications in Sciences, pp. 170–175. IEEE Computer Society, Washington (2008)
16. Almeida-Luz, S.M., Vega-Rodríguez, M.A., Gómez-Pulido, J.A., Sánchez-Pérez, J.M.: Applying Scatter Search to the Location Areas Problem. In: Corchado, E., Yin, H. (eds.) IDEAL 2009. LNCS, vol. 5788, pp. 791–798. Springer, Heidelberg (2009)
17. Almeida-Luz, S.M., Vega-Rodríguez, M.A., Gómez-Pulido, J.A., Sánchez-Pérez, J.M.: Solving a Realistic Location Area Problem Using SUMATRA Networks with the Scatter Search Algorithm. In: Proceedings of the Ninth International Conference on Intelligent Systems Design and Applications, pp. 689–694. IEEE Computer Society, Washington (2009)
18. Knowles, J., Thiele, L., Zitzler, E.: A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers. Technical report 214, Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Switzerland (2006)
19. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002)
20. Srinivas, N., Deb, K.: Multiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary Computation 2(3), 221–248 (1994)