Metaheuristic Approaches for Optimal Broadcasting Design in Metropolitan MANETs

E. Alba[1], A. Cervantes[2], J.A. Gómez[4], P. Isasi[2], M.D. Jaraíz[4], C. León[3], C. Luque[2], F. Luna[1,⋆], G. Miranda[3], A.J. Nebro[1], R. Pérez[2], and C. Segura[3]

[1] Computer Science Department, University of Málaga
[2] Computer Science Department, University Carlos III of Madrid
[3] Department of Statistics, O.R. and Computation, University of La Laguna
[4] Department of Informatics, University of Extremadura

Abstract. Mobile Ad-hoc Networks (MANETs) are composed of a set of communicating devices which are able to spontaneously interconnect without any pre-existing infrastructure. In such scenario, broadcasting becomes an operation of tremendous importance for the own existence and operation of the network. Optimizing a broadcasting strategy in MANETs is a multiobjective problem accounting for three goals: reaching as many stations as possible, minimizing the network utilization, and reducing the duration of the operation itself. This research, which has been developed within the OPLINK project (`http://oplink.lcc.uma.es`), faces a wide study about this problem in metropolitan MANETs with up to seven different advanced multiobjective metaheuristics. They all compute Pareto fronts of solutions which empower a human designer with the ability of choosing the preferred configuration for the network. The quality of these fronts is evaluated by using the hypervolume metric. The obtained results show that the SPEA2 algorithm is the most accurate metaheuristic for solving the broadcasting problem.

1 Introduction

With the rapid development of wireless communications technologies and the proliferation of mobile devices like cell phones, PDAs or laptops, mobile *ad hoc* networks (MANETs) have emerged as an important research field in current and future communication networks because they do not require infrastructure support and can be quickly deployed with low costs. MANETs consist of a collection of mobile, self-configurable hosts, called *nodes* or *devices*, which are free to move randomly and organize themselves arbitrarily. The mobility of devices along with the range-limited wireless links make the network topology to change rapidly and unpredictably over time. This dynamical behavior constitutes one of the main obstacles for performing efficient communications on such networks.

This work is focussed on the problem of broadcasting in MANETs. In such networks, broadcasting is not only the basic mechanism for solving many network

⋆ Corresponding author. Email: `flv@lcc.uma.es`

layer problems (e.g. routing) but also a common operation at the application level. Hence, choosing a broadcasting strategy properly will result in a major impact in network performance. In this paper, we are considering the problem of broadcasting over the so-called *Metropolitan* MANETs [1], where the density of devices is heterogeneous and continuously changing. This leads to networks composed of subsets of ad hoc networks that may merge and disjoin dynamically during the operation, so the network topologies change frequently (Fig. 1). In this context, rather than providing a generic protocol performing well on average situations, our proposal lies in optimally tuning the broadcasting service for a set of networks.

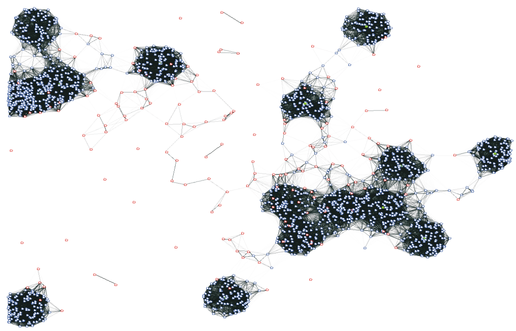

Fig. 1. An example of metropolitan MANET

Optimizing a broadcasting strategy is a multiobjective optimization problem (MOP), in which multiple functions have to be satisfied at the same time: maximizing the number of stations reached, minimizing the network use, and minimizing the duration of the process. In this work, the broadcasting strategy considered for optimization is DFCN [2], and the target networks are metropolitan MANETs. This MOP has been called DFCNT standing for *DFCN Tuning*. DFCNT is a real-world, complex optimization MOP which is very hard to be solved exactly within a reasonable amount of time. Heuristics become the choice here. These methods sacrifice the guarantee of finding the optimal solutions for the sake of (hopefully) getting accurate (also optimal) ones efficiently. Among them, metaheuristics [3] are well-known optimization algorithms that are based on combining basic heuristic methods in a higher level structured framework. The aim is to efficiently and effectively explore the search space of the given problem while seeking for its optimal solutions.

The goal of this work is to present a broad study in which seven different state-of-the-art multiobjective metaheuristics for solving the optimal broadcasting problem are evaluated. These metaheuristics are: NSGA-II [4] and SPEA2 [5] (genetic algorithms), MOPSO [6] (particle swarm optimization), AbYSS [7] (scatter search), cMOGA [8] (cellular genetic algorithm), and two adaptations of Evolution Strategies (ES) [9] and Differential Evolution (DE) [10] to the multiobjective optimization field. We have been able to address this extensive study on such a complex problem thanks to the collaboration developed among the different research teams of the OPLINK project (`http://oplink.lcc.uma.es`). The optimizers have been compared by using the standard methodology in the multiobjective field and the drawn conclusions have been supported with an statistical analysis.

The paper is structured as follows. The next section provides the reader with a brief description of the DFCNT MOP that is being addressed in this work. Section 3 describes all the metaheuristics used. Metrics, parameterization, and results are presented in Sect. 4. Finally, conclusions and lines of future work are given in Sect. 5.

2 DFCNT: Problem Statement

Metropolitan mobile ad hoc networks are MANETs with some particular properties. First, they have one or more areas where the node density is higher than the average. They are called *high density areas*, and they can be statistically detected. Second, high density areas do not remain active full time, i.e., they can appear and disappear from the network. To deal with such kind of networks, there is no other solution than resorting to software simulations. In this work we have used *Madhoc* [11], a metropolitan MANET simulator (`www-lih.univ-lehavre.fr/~hogie/madhoc`). Very different realistic scenarios can be implemented within Madhoc (open areas, metropolitan environments, highways, etc.) because of its highly customizable behavioral parameters. We have configured the simulator for modelling a mall environment according to three main parameters: the size of the simulation area, the density of mobile stations, and the type of environment. For our experiments, we have used a simulation area of 40,000 square meters, a density of 2,000 devices per square kilometer, and a mall-like environment (see [8] for further details). We consider that the broadcasting is completed when either the coverage reaches 100% or it does not vary in a reasonable period of time (set to 1.5 seconds after some preliminary experimentation). This point is important since an improper termination condition will lead us to bad results or very slow simulations.

Once we have the target network, the broadcasting strategy used to be tuned is DFCN [2]. We have chosen this protocol because it has been specifically designed for metropolitan MANET scenarios. After carefully analyzing DFCN, we identified five parameters which determine the behavior of the protocol. For all the details about the DFCNT parameters, the reader is referred to [8]. These five parameters define five decision variables which correspond to a DFCN configuration. The objectives to optimize are: minimizing the duration of the broadcasting process, maximizing the network coverage, and minimizing the number of transmissions. Thus, we have defined a triple objective MOP, called DFCNT (*DFCN Tuning*). As we stated before, this problem is defined by a given target network in which the DFCN broadcasting strategy is used. We do not seek for a protocol performing well on every possible scenario.

3 Metaheuristics for Solving DFCNT

In this section we give a brief overview of the seven optimization methods used in this work. For further details, please use the given references on each algorithm.

- **NSGA-II:** The NSGA-II algorithm was proposed by Deb *et al.* [4]. It is a genetic algorithm based on obtaining a new population of individuals from the original one by applying the typical genetic operators (selection, crossover, and mutation); then, the most promising individuals for the next generation are chosen according to their rank and crowding distance. When it is used to solve continuous problems, NSGA-II applies binary tournament selection, SBX crossover and polynomial mutation.
- **SPEA2:** It is also a genetic algorithm, proposed by Zitler *et al.* in [5], in which each individual has assigned a fitness value that is the sum of its strength raw fitness and a density estimation. The algorithm applies the selection, crossover, and mutation operators to fill an archive of individuals; then, the non-dominated individuals of both the original population and the archive are copied into a new population. If the number of non-dominated individuals is greater than the population size, a truncation operator based on calculating the distances to the k-th nearest neighbor is used.
- **MOPSO:** This technique, proposed by Coello and Lechuga in [6], combines particle swarm optimization with the archiving strategy of PAES [12]. Non-dominated solutions in the swarm are stored in an external repository and they are used to guide the search. This repository is used both to store the solutions of the problem and to maintain a diverse population of particles. We have used the basic version of MOPSO [6], without the mutation operator.
- **AbYSS:** The Archive-Based hYbrid Scatter Search [7,13] is our own proposal to adapt the scatter search template to solve bounded continuous single objective optimization problems [14] to the multiobjective domain. The SBX crossover and the polynomial mutation operators are used in the solution combination and improvement methods, respectively. AbYSS uses an external archive for storing non-dominated solutions, which is managed by using the ranking and crowding of NSGA-II [4] as a niching measure.
- **cMOGA:** cMOGA [8] is a cellular genetic algorithm [15] which uses an external archive for storing the non-dominated solutions found during the search. The management strategy of this external archive is similar to the one used by PAES (as MOPSO does). This algorithm also uses SBX and polynomial mutation as genetic operators in its breeding loop.
- **Differential Evolution (DE):** We use an approach based on the reproduction scheme of DE [10] within the NSGA-II reproductive loop. That is, the algorithm works on a population of N vectors and, at each iteration, a new offspring population of size N is generated by using the DE scheme. The ranking and crowding mechanisms from NSGA-II are used to choose the N best vectors out of $2N$ for the next iteration.
- **Evolution Strategy (ES):** This multiobjective ES algorithm is a rather standard self-adaptive evolution strategy [9]; in particular, it is a $(\mu+\lambda)$ ES

using the selection scheme of NSGA-II (based on ranking and crowding distances). The mutation operator used is the gaussian mutation, which is applied using a variation of the 1/5 success rule [9].

4 Experiments

This section is devoted to present the experiments performed for this work. We firstly introduce the metric used for measuring the performance of the resulting Pareto fronts. Secondly, the parameterization of all the multiobjective algorithms is detailed. Finally, we thoroughly discuss and compare the results for DFCNT.

4.1 Quality Indicator: Hypervolume

We have used the Hypervolume [16] metric for assessing the performance of the multiobjective optimizers. This is a widely used, Pareto-compliant metric that measures both convergence and diversity in the resulting Pareto fronts. It is based on calculating the volume (in the objective space) covered by members of a non-dominated set of solutions Q. Let v_i be the volume enclosed by solution $i \in Q$. Then, a union of all hypercubes is found and its hypervolume (HV) is calculated:

$$HV = volume\left(\bigcup_{i=1}^{|Q|} v_i\right). \tag{1}$$

Algorithms with larger values of HV are desirable. Since this metric could be biased by arbitrary scaling of objectives, we have evaluated the metric by using normalized objective function values.

4.2 Parameterization

All the algorithms stop when 25,000 function evaluations of DFCNT have been computed. Due to the stochastic nature of Madhoc, five simulations per function evaluation have been performed, so that the fitness values of the functions are computed as the average resulting values of these five different trials. A maximum archive size of 100 non-dominated solutions is set in those algorithms using external archives. SBX crossover and polynomial mutation have been used in NSGA-II, SPEA2, AbYSS, and cMOGA. The detailed configuration of each algorithm is as follows:

- NSGA-II: It uses a distribution index of 10 for both SBX and polynomial mutation operators. The crossover and mutation rates have been set up to 0.9 and 0.2, respectively (as suggested by the designers of the algorithm).
- SPEA2: It has been configured with a population size of 100 individuals. The η_c value for SBX was fixed to 10 and the crossover rate to 1.0. In the polynomial mutation, η_m was set up to 20 and the mutation probability to 0.01.

– MOPSO: We have used 100 particles and 250 iterations to meet 25,000 function evaluations. The values for computing the new velocity vectors and new position of the particles are: $w = 0.4$, $c1 = c2 = 2.0$, and $X = 0.4$ (see [6] for the details).
– AbYSS: The configuration of both the SBX crossover and polynomial mutation (solution combination and improvement methods) uses a distribution index of 10. The size of the initial set P is 20, the number of iterations in the improvement method is 5, and the size of $RefSet_1$ and $RefSet_2$ is 10 solutions.
– cMOGA: It uses the same configuration for SBX and polynomial mutation as AbYSS. The algorithm has been configured with a population of 100 individuals arranged in a 10×10 square toroidal grid with a NEWs neighborhood, and using binary tournament selection, a crossover rate of 1.0, and mutation rate of 0.2.
– DE: From the set of predefined strategies, DE/rand/1/bin has been selected, which respectively means that a random vector is mutated, one single vector is considered for perturbation of this randomly chosen vector, and binomial crossover is used for recombination (see [10] for the details). Two additional parameters are needed to be assigned in DE, the weighting factor (F) and the crossover rate (CR). The chosen values for these parameters have been 0.5 and 1.0, respectively.
– ES: The ES algorithm has been configured with 100 individuals so that $\mu = \lambda = 100$ (and therefore 250 iterations). The step-size meta-control parameter Δ is equal to 0.7.

4.3 Results

We have made 30 independent runs of each experiment. The results are shown in Table 1 (best algorithms are at the top) which includes the median, $\tilde{x}$, and interquartile range, IQR, as measures of location (or central tendency) and statistical dispersion. Since we are dealing with stochastic algorithms and we want to provide the results with confidence, the following statistical analysis has been performed in all this work. First a Kolmogorov-Smirnov test is performed in order to check whether the values of the results follow a normal (gaussian) distribution or not. If so, the Levene test checks for the homogeneity of the variances. If samples have equal variance (positive Levene test), an ANOVA

Table 1. HV values

Algorithm	$\tilde{x}_{IQR}$
SPEA2	8.807e−01 $_{4.7e-03}$
ES	8.755e−01 $_{2.7e-03}$
NSGA-II	8.741e−01 $_{5.9e-03}$
AbYSS	8.687e−01 $_{3.8e-02}$
DE	8.672e−01 $_{8.4e-03}$
MOPSO	8.644e−01 $_{3.8e-03}$
cMOGA	8.480e−01 $_{5.4e-02}$

test is done; otherwise we perform a Welch test. For non-gaussian distributions, the non-parametric Kruskal-Wallis test is used to compare the medians of the algorithms. We consider here a confidence level of 95% (i.e., significance level of 5% or p-value under 0.05), which means that the differences are unlikely to have occurred by chance with a probability of 95%. After applying this test, we verify that statistical difference exists amongst the values included in Table 1.

From the values in the table, we can observe that SPEA2 obtains the best metric value, thus indicating that it is the most promising metaheuristic to solve DFCNT. This is an expected result since DFCNT is a three-objective MOP, and it is well-known that the density estimator of SPEA2 overcomes those used in NSGA-II and PAES (used in the rest of algorithms) in MOPs having more than two objectives [17]. In this sense, it is remarkable the HV values reached by the ES algorithm, which outperforms NSGA-II, the reference algorithm in multiobjective optimization. The last position of cMOGA in Table 1 is a consequence of its simplified archive management strategy, in which the individuals in the archive are not reused in the breeding loop of the algorithm. Anyway, the differences in the metric values are not large, indicating that all the algorithms compute similar fronts.

It is worth mentioning the time required to run each experiment. In a modern Intel Pentium 4 based PC, this time is in the order of 2.5 days. We have carried out 25,000 function evaluations, which is a typical value when measuring the performance of multiobjective algorithms using standard benchmarks, but this number is probably insufficient to obtain accurate fronts of solutions from a difficult problem such as DFCNT. However, increasing the number of evaluations can turn the runtime of the algorithms as impractical, what suggests that parallel techniques have to be considered if we intend to enhance the obtained results.

5 Conclusions and Future Works

We have analyzed seven multiobjective metaheuristics for solving DFCNT, the tuning of the DFCN broadcasting protocol for metropolitan MANETs. It is a complex real world problem, which uses Madhoc, a metropolitan MANET simulator, to model a realistic scenario that represents a shopping mall. This leads to a MOP having five decision variables and three objectives.

The chosen metaheuristics are representative of the state-of-the-art, and they include the well-known algorithms NSGA-II and SPEA2. Apart from these two methods, we have considered other metaheuristics: scatter search, cellular genetic algorithm, evolution strategy, differential evolution, and particle swarm optimization. The implementation of these algorithms and their application to solve DFCNT have been a join work of the members of the OPLINK project, involving research groups of four Spanish universities. The experiments carried out reveal that SPEA2 is the algorithm yielding the best results, according to the hypervolume metric. However, the metric values seem to indicate that all the algorithms produce fronts which are close together.

As lines of future work, more research has to be done to improve the Pareto fronts of the DFCNT problem. The study of fine-tuning the algorithms to enhance their performance and the use of parallel techniques to compute more function evaluations in a reasonable amount of time are matters of ongoing and future developments.

Acknowledgements. The authors are partially supported by the Spanish Ministry of Education and Science, and by European FEDER under contract TIN2005-08818-C04 (the OPLINK project, http://oplink.lcc.uma.es).

References

1. Conti, M., Giordano, S., Maselli, G., Turi, G.: MobileMAN: Mobile Metropolitan Ad Hoc Networks. In: Proc. of the Eight Int. IFIP-TC6 Conf, pp. 194–205 (2003)
2. Hogie, L., Guinand, F., Bouvry, P.: A Heuristic for Efficient Broadcasting in the Metropolitan Ad Hoc Network. In: Negoita, M.G., Howlett, R.J., Jain, L.C. (eds.) KES 2004. LNCS (LNAI), vol. 3213, pp. 727–733. Springer, Heidelberg (2004)
3. Glover, F.W., Kochenberger, G.A.: Handbook of Metaheuristics. Kluwer, Dordrecht (2003)
4. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. IEEE Trans. on Evolutionary Computation 6(2), 182–197 (2002)
5. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Technical report, Swiss Federal Inst. of Technology (2001)
6. Coello, C.A., Lechuga, M.S.: MOPSO: A proposal for multiple objective particle swarm optimization. In: CEC 2002, pp. 1051–1056 (2002)
7. Nebro, A.J., Luna, F., Dorronsoro, B., Alba, E., Beham, A.: AbYSS: Adapting Scatter Search to Multiobjective Optimization. Technical Report ITI-2006-2, Dpto. Lenguajes y Ciencias de la Computación (2006)
8. Alba, E., Dorronsoro, B., Luna, F., Nebro, A., Bouvry, P.: A Cellular Multi-Objective Genetic Algorithm for Optimal Broadcasting Strategy in Metropolitan MANETs. Computer Communications 30(4), 685–697 (2007)
9. Bäck, T., Rüdolph, G., Schwefel, H.: A survey of evolution strategies. In: 4th Int. Conf. on Genetic Algorithms, pp. 2–9 (1991)
10. Storn, R., Price, K.: Differential evolution – a simple efficient adaptive scheme for global optimization over continuous spaces. Technical Report 95-012, Int. Compt. Sci. Inst., Berkeley, CA (1995)
11. Hogie, L., Guinand, F., Bouvry, P.: The Madhoc Metropolitan Adhoc Network Simulator. Université du Luxembourg and Université du Havre, France, Available at (2005), `http://www-lih.univ-lehavre.fr/~hogie/madhoc/`
12. Knowles, J., Corne, D.: The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Multiobjective Optimization. In: Proceedings of the 1999 Congress on Evolutionary Computation, CEC, pp. 9–105 (1999)
13. Luna, F., Nebro, A., Dorronsoro, B., Luna, F., Alba, E., Bouvry, P., Hogie, L.: Optimal Broadcasting in Metropolitan MANETs Using Multiobjective Scatter Search. In: Rothlauf, F., Branke, J., Cagnoni, S., Costa, E., Cotta, C., Drechsler, R., Lutton, E., Machado, P., Moore, J.H., Romero, J., Smith, G.D., Squillero, G., Takagi, H. (eds.) EvoWorkshops 2006. LNCS, vol. 3907, pp. 255–266. Springer, Heidelberg (2006)

14. Glover, F., Laguna, M., Martí, R.: Scatter Search. In: Advances in Evolutionary Computing: Theory and Applications, pp. 519–539. Springer, New York (2003)
15. Alba, E., Tomassini, M.: Parallelism and evolutionary algorithms. IEEE Trans. on Evolutionary Computation 6(5), 443–462 (2002)
16. Zitzler, E., Thiele, L.: Multiobjective Optimization Using Evolutionary Algorithms – A Comparative Study. In: Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.-P. (eds.) Parallel Problem Solving from Nature - PPSN V. LNCS, vol. 1498, pp. 292–301. Springer, Heidelberg (1998)
17. Kukkonen, S., Deb, K.: A fast and effective method for pruning of non-dominated solutions in many-objective problems. In: Runarsson, T.P., Beyer, H.-G., Burke, E., Merelo-Guervós, J.J., Whitley, L.D., Yao, X. (eds.) Parallel Problem Solving from Nature - PPSN IX. LNCS, vol. 4193, pp. 553–562. Springer, Heidelberg (2006)