Noname manuscript No.
(will be inserted by the editor)

Finding Scalable Configurations for AEDB Broadcasting Protocol using Multi-objective Evolutionary Algorithms

Patricia Ruiz · Bernabe Dorronsoro · Pascal Bouvry

Received: 25th January 2012 / Accepted: date

Abstract Energy consumption is one of the main concerns in mobile ad hoc networks (or MANETs). The lifetime of its devices highly depends on the energy consumption as they rely on batteries. The adaptive enhanced distance based broadcasting algorithm, AEDB, is a message dissemination protocol for MANETs that uses cross-layer technology to highly reduce the energy consumption of devices in the process, while still providing competitive performance in terms of coverage and time. We use two different multi-objective evolutionary algorithms to optimize the protocol on three network densities, and we evaluate the scalability of the best found AEDB configurations on larger networks and different densities.

Keywords Broadcasting protocols · Optimization algorithms · ad hoc networks · energy efficiency.

Patricia Ruiz
University of Luxembourg
Luxembourg
Tel.: +352-44-666445514
Fax: +352-44-666445500
E-mail: patricia.ruiz@uni.lu

Bernabe Dorronsoro
Interdisciplinary Center of Security,
Reliability, and Trust
Luxembourg
Tel.: +352-44-666445619
Fax: +352-44-666445500
E-mail: bernabe.dorronsoro@uni.lu

Pascal Bouvry
University of Luxembourg
Luxembourg
Tel.: +352-44-666445258
Fax: +352-44-666445500
E-mail: pascal.bouvry@uni.lu

1 Introduction

Broadcasting is considered as one of the most important low level operations in networking, as many applications and even other protocols rely on its service. In the case of wireless networks, these dissemination algorithms are generally associated with the broadcast storm problem [30]. However, due to the recently appearance of mobile ad hoc networks (MANETs), and all the drawbacks inherited from them (battery life, mobility of devices, limited transmission range, etc.), the main problem in broadcasting is not only reducing the number of forwardings but also trying to overcome all these undesirable aspects.

One of the main drawbacks of MANETs is the dependence on the battery life of the devices, as when they run out of battery the network capabilities decrease, and might lead to the disappearance of the network. This is the reason why many researchers focus on reducing the energy consumption of devices conforming the MANET [28,33].

In this work, we are improving the performance of the adaptive enhanced distance based broadcasting algorithm (AEDB hereinafter) [32]. We look for robust solutions that offer good performance for a number of densities and scenarios of different sizes. AEDB is an energy aware and distance based broadcasting algorithm that uses a cross-layer design to reduce the energy consumption. The mechanism of AEDB relies on some thresholds that allow every device to make decisions on whether to forward the received message or not, and what transmission energy to use when forwarding. These thresholds have been experimentally chosen. In this work, we optimize the values of these thresholds using some multiobjective techniques, and some of the proposed solutions are analyzed.

The contribution of this paper is threefold. First, we provide a new multi-objective definition of the problem of optimizing AEDB performance, more precise that the one studied in [34], which is considering the broadcast time as a constraint and is taking into account the total number of forwardings in the network. This is done in order to avoid the undesirable situation in which most (or even all) devices forward the message using very low transmission power, causing a high use of the channel. Second, we perform a comparison on the scalability of the multi-objective algorithms when optimizing AEDB for different network densities. Third, we analyze the scalability of the best chosen solutions on a number of networks of different sizes and densities, comparing the results with the best previously existing AEDB configuration.

The rest of the paper is organized as follows. Next section provides a brief state of the art in energy aware broadcasting algorithms for MANETs and on the use of metaheuristics for protocol optimization. In Sect. 3, the description of AEDB is presented. Section 4 introduces the problem in hands, and Sect 5 the optimization algorithms used. The configuration used for both algorithms as well as its simulation parameters are explained in Sect. 6. The results obtained are presented and analyzed in Sect. 7. And finally, we conclude the paper in Sect. 8.

2 Related Work

We describe in this section the most outstanding related papers to our work. In Sect. 2.1, we revise the state of the art on broadcasting protocols with energy efficiency considerations for MANETs. Then, we summarize the main existing works using optimization algorithms to enhance the performance of protocols for MANETs in Sect. 2.2.

2.1 Energy Aware Broadcasting Protocols for MANETs

As we mentioned before, the energy consumption in mobile ad hoc networks is a hot topic, since devices can run out of battery provoking the network degradation. Many researchers are focused on this aspect and therefore much work has been done. Below, we mention some of the most outstanding solutions that have been proposed for saving energy in broadcasting algorithms dealing with ad hoc networks.

Gomez et al. showed in [20] that a variable transmission range can outperform a common range transmission approach in terms of power saving, with the possibility of increasing the capacity too. They also claimed that there is an optimal setting for the transmission range, not necessarily minimum, which maximizes the available capacity of the nodes in presence of mobility.

In [8], nodes exchange information in the beacons in order to know the transmission power needed to reach the two hop neighbors. The source node examines if the furthest node in the one hop neighbor is reachable through any other neighbor, if so, it calculates the power needed via two hops. If the sum of powers needed using two hops is less than the power of sending the message directly, the source node will exclude the furthest node from the neighborhood and reduce the transmission range to reach the new furthest neighbor.

In [28] the transmission range is set in terms of the local density. To estimate the local density, a node calculates the number of neighbors by listening the radio channel and evaluating the distance from each neighbor (signal strength or timing differential).

Extensive studies on energy efficient algorithms for finding the minimum-energy broadcast tree (MEBT) have been proposed [6,7]. Also, in [29], a minimum energy shared multicast tree built in a distributed fashion is presented, where the transmission power is either fixed or adjustable.

In [38] each node continuously monitors, records and updates the transmission power level it needs to reach all its neighbors by overhearing all messages, even the ones that are not intended for it. This is not a broadcasting approach, so when sending a message, the source node will use the power needed to reach the intended neighbor.

In vehicular ad hoc networks, it is also a tendency to adjust the transmission range used in order to reduce the number of collisions, interferences, etc. In [31] nodes exchange beacons periodically with other vehicles in range containing information about the path loss, and neighbors are sorted according to the average path loss. There is an specific target number of neighbors to reach, so that when a broadcast message is received, the node checks the transmission power necessary to reach the targeted number of nodes.

Ruiz and Bouvry proposed in [33] an enhanced distance based broadcasting algorithm (called EDB) that not only considerably reduces the number of collisions but also the energy consumption in the broadcast process without degrading the network connectivity. Later in [32] the authors improved the protocol by making it adaptive to discard some one hop neighbors in very dense networks in order to save energy.

Another distance based adaptive protocol is DDAPF [24], that dynamically adjusts the probability of forwarding algorithm in terms of the distance to the source node.

2.2 Protocols Optimization for MANETs with Metaheuristics

We can find in the literature some works dealing with metaheuristics to optimize a number of different problems on mobile ad hoc networks. Many of these works consider the use of metaheuristics in the network elements to optimize some problem (as finding efficient routing paths) [1, 9, 11, 23, 36, 39]. They are often implemented in the devices or in some centralized structure. However, these approaches are in general very difficult to use in real MANETs because they require either intensive computations in the battery limited devices or the presence of some infrastructure in the network.

There are in the literature a few papers presenting the use of metaheuristics to optimize the behavior of protocols for MANETs. In this case, the protocol optimization is an offline process that (usually) looks for the optimal configuration of the considered protocol to enhance some aspect of the network. Some examples are optimizing the network QoS, the network use, or the energy used, as it is the case considered in this work. Probably, the first studies in this line were those by Alba et al. [2–4], in which a broadcasting protocol for MANETS called DFCN [22] was optimized for three different environments, namely, a shopping mall, an urban area, and a highway scenario. The problem was solved with multi-objective techniques, since the protocol was optimized in terms of its coverage (number of devices reached by the broadcasting message), the network use, and the total broadcasting time. After these initial papers, a number of works appeared studying the same problem in the literature [14–16, 26, 27].

García-Nieto et al. published some works on the optimization of the parameters of AODV routing protocol for vehicular ad hoc networks [17] and a file transfer protocol [18]. Later, Toutouh et al. [37] presented a parallel genetic algorithm to optimize the energy used by the OLSR routing algorithm in VANETs subject to acceptable QoS requirementes. Contrary to the previously commented works, in these two ones the authors are using single-objective techniques to optimize a weighted sum of the defined goals. The consequence is that the algorithm is providing only one solution to the problem, which is strongly biased by the weights used in the fitness function, while a multi-objective technique would offer a wide range of very diverse solutions to the problem, none better than the other, that will allow the protocol designer to choose the most appropriate one.

Recently, Ruiz et al. [34, 35] propose the use of the CellDE multi-objective algorithm to optimize the performance of EDB and AEDB broadcasting protocols in MANETs by maximizing the coverage achieved in the dissemination process and minimizing the time and the energy used. In this work, we define a different problem to optimize AEDB. It lies in maximizing the coverage and minimizing the energy used and the number of forwarded messages in the network. This latter objective is added as an attempt to avoid the undesirable case in which many messages are sent with very low transmission range, since it would cause a network congestion. The optimization is carried out subject to a constraint on the maximum allowed broadcasting time.

3 Adaptive Enhanced Distance Based Broadcasting Algorithm, AEDB

In the minimum energy broadcasting problem every node is able to adjust the transmission range in order to reduce the power consumption of the dissemination process while still guaranteeing full coverage in the network. In a real scenario with obstacles, devices moving, fading, path loss, packet loss, etc. guaranteeing the full coverage might be very ambitious or impossible (due to network partitioning), and in some cases even unnecessary. In safety, control or important messages it might be worth the overhead needed for delivering the message to all nodes in the network. But for all the other messages (info, ads, etc.), it would rather be more efficient to consider the possibility of not guaranteeing full coverage, and thus, saving all the overhead derived from acknowledgements, retransmissions, etc.

In this work, we consider this second family of protocols, where full coverage is not required.

The adaptive enhanced distance based broadcasting algorithm (AEDB hereinafter) aims at saving energy in sparse networks as well as in dense ones. AEDB [32] is an extension of EDB [33], a broadcasting algorithm that reduces the transmission power for disseminating a message. As any distance based broadcasting algorithm, nodes are candidates to forward the message if the distance to the source node is higher than a predefined threshold. Thus, there exists a forwarding area, and only nodes located in it are potential forwarders. In this case, we are using a crosslayer technique that informs the upper layers about the signal strength of messages received. Therefore, the decision is not taken in terms of distance (m) but power (dBm). This predefined value for the energy is called the *borders_Threshold*.

EDB tries to save energy by reducing the transmission power when forwarding the broadcasting message. The new transmission power is the one that reaches the furthest neighbor. The energy needed is estimated according to the reception energy detected in the beacons exchanged (every 1 second). In order to be aware of the nodes mobility, an extra fixed amount of energy is added to the one estimated. This is called the *margin_Threshold*.

In denser networks, the probability of having a node close to the limit transmission range is higher, therefore, EDB does not reduce the transmission power. Indeed, when the network is very dense the connectivity is usually very high. Thus, reducing the transmission power allowing the loss of some one hop neighbors will save energy without any detriment in the performance of the broadcasting process. Contrary, when the network is sparse, the node must maintain the network connectivity, as not doing so would make more difficult to spread a message through the whole network.

AEDB considers the possibility of discarding some neighbors from the one hop neighborhood in dense networks. In fact, the algorithm is able to adapt its behaviour to the network density. Potential forwarders set a random delay before resending. If, during this time, many nodes located in the forwarding area are detected (called *neighbors_Threshold*), the transmission range is reduced and some one hop neighbors are discarded. The new furthest neighbor is the node located in the forwarding area that is the closest one to the source node. A more detailed explanation can be found in [32].

Algorithm 1 Pseudocode of the new Adaptive EDB.

```
Data: m: the incoming broadcast message.
Data: r: the node receiving broadcast message.
Data: s: the node that sent m.
Data: p: the received signal strength of m sent by s.
Data: pmin: the minimum signal strength received from any s.
Data: potentialForwarders: # neighbors in the forwarding area.

if m is received for the first time then
    calculate p;
    update pmin;
    if pmin > borders_Threshold then
        r → drop message m;
    else
        waiting ← true;
        wait time rand ∈ [delay interval];
    end if
else if waiting then
    calculate p;
    if p > pmin then
        update pmin;
    end if
end if
if pmin > borders_Threshold then
    r → drop message m;
else
    if # potentialForwarders > neighbors_Threshold then
        estimate p to reach closest neighbor to borders_Threshold
    else
        discard s from the one hop neighbors list.
        estimate p to reach furthest neighbor
    end if
    transmit m;
end if
waiting ← false;
```

4 Optimization of AEDB Protocol

The quality of the performance of a broadcasting algorithm in ad hoc networks is usually related to some standard measurements. The aspects we are considering and that are the most common ones in these kind of protocols are:

1. the **coverage** obtained, i.e., the number of devices that after the dissemination process receive the broadcast message;
2. the **energy used** by the broadcast process, measured as the sum of the energy every device consumes to forward the message;
3. the number of **forwardings**, considered as the amount of nodes that after receiving the broadcasting message decide to resend it;
4. and the **broadcast time**, considered as the time needed to spread a message in the network, since the source node sends the message until the last node receives it.

Previously in [34], AEDB was optimized considering three objectives: (1) the energy used, (2) the coverage achieved, and (3) the broadcast time. The optimization algorithm used found many solutions where the value of the *borders_Threshold* was close to the upper limit (-70 dBm). That meant the algorithm was promoting lower transmission power and higher number of forwardings. The number of forwardings itself was not optimized in that work, but was intrinsically related because the higher the number of forwardings the longer the broadcasting time.

In this work, apart from comparing two different optimization algorithms, selecting the most appropriate values for the parameters and analyzing them in large scale networks, we considered it was worth verifying the hint concluded in [34]. Therefore, in this case, we are considering the number of forwardings as another objective for the optimization algorithm.

From the point of view of the designer of the broadcasting algorithm, the higher the number of objectives the more complex the decision making and the optimization process. Thus, for this work, we consider only three objectives and a constraint.

We can observe in the results obtained in [34] that the best solutions found do not take longer than 2 seconds for disseminating the broadcasting message. Therefore, in the evaluation process of the optimization we consider a solution is no longer valid if the broadcasting time is higher than 2 seconds, and analyze the following three objectives: (1) energy used, (2) coverage achieved, and (3) number of forwardings used.

As mentioned in Sect. 3, AEDB has a set of fixed parameters whose values determine the behaviour of the protocol. Those thresholds are explained after and listed here: *borders_Threshold*, *margin_Forwarding*, the *delay* interval, and *neighbors_Threshold*.

- The value of the *borders_Threshold* sets the size of the forwarding area. The higher the threshold, the higher the number of potential forwarders, the coverage, the network resources and the number of collisions.
- The *margin_Forwarding* is related to both the energy saved and the coverage achieved. It is the extra amount of energy added to the estimated transmission power. The higher the margin value, the higher the coverage reached as well as the energy used.
- The value of the *delay* interval sets the waiting time and also affects the behaviour of the protocol. If the delay is very high, the time used to spread the message will be high, but if it is very small, the number of collisions will probably increase.
- Finally, the *neighbors_Threshold* that fixes the minimum number of neighbors in the forwarding area needed to discard some nodes. It affects the use of the network and the energy used. The lower the value, the lower the energy used and the higher number of forwardings.

The purpose of this work is to tune all these parameters using multi-objective techniques (based on Pareto dominance) in order to obtain the best possible behaviour of the protocol, considering the three objectives and the constraint explained above. Below, we include a formal definition of the problem.

s : instance of the ns3 simulator
$\mathbf{dmin} = d_1 \in \mathbb{R} | d_1 \in \textit{minimum delay}$
$\mathbf{dmax} = d_2 \in \mathbb{R} | d_2 \in \textit{maximum delay}$
$\mathbf{b} = b_1 \in \mathbb{R} | b_1 \in \textit{border\_threshold}$
$\mathbf{m} = m_1 \in \mathbb{R} | m_1 \in \textit{margin\_threshold}$
$\mathbf{n} = n_1 \in \mathbb{R} | n_1 \in \textit{neighbor\_threshold}$
$\mathbf{z} = (e, c, f, bt)$
$s(dmin, dmax, d, m, n) = z$

$$f(dmin, dmax, b, m, n) = \begin{cases} min\,\{e\} \\ max\,\{c\} \\ min\,\{f\} \end{cases} ;\text{s. t. } bt < 2 \quad (1)$$

where z is the set of objectives: e stands for energy saved, c for coverage, f for number of broadcastings and bt is the broadcasting time. The domain of the variables *minimum delay*, *maximum delay*, *border_threshold*, *margin_threshold*, and *neighbor_threshold* are presented in detail in Table 1.

5 The Optimization Algorithms

Metaheuristics [19] are iterative stochastic optimization tools that are able to provide good solutions in reasonable time for highly complex optimization problems. Generally, metaheuristics make no assumptions about the problem to solve, so they are generic tools that only need an adequacy (or fitness) function to guide the search towards better solutions.

Evolutionary Algorithms (EAs) [5] are a popular family of metaheuristics. One important feature of EAs is that they work with several candidate solutions at the same time, therefore simultaneously exploring several different regions of the search space. This allows EAs to better explore the search space and reducing the probabilities of getting stuck in local optima with respect to other metaheuristics families.

In this paper, our problem is defined as a three-objectives one, since to optimize the protocol performance we need to maximize its coverage, and minimize the energy used and the number of forwarded messages by the devices. Therefore, we rely on multi-objective optimization algorithms. Specifically, we use NSGAII [10], which is probably the most referenced multi-objective algorithm in the literature, and CellDE, because it is a highly competitive evolutionary algorithm for multi-objective optimization that has proven to perform specially well for three-objectives problems with continuous variables [12], as it is the case of our problem. In this work, we used the implementations of the algorithms available in jMetal framework [13], with the configurations provided in their original papers (set by default in jMetal).

The use of these two multi-objective optimization algorithms will allow us to find a finite set of non-dominated configurations (none is better than the others for the three objectives) for our AEDB protocol containing the solutions with the best possible trade-off among the three objectives. This will help in understanding the impact of the different parameters on the behavior of the protocol, as well as choosing the best possible solution for our particular scenario.

6 Empirical Setup

As mentioned before, we use in this paper CellDE and NSGAII algorithms to look for the optimal configuration of the AEDB parameters (defined in Sect. 4) to get the best possible performance of the considered broadcasting protocol. Measuring the quality of a given parameter configuration (i.e., a tentative solution to the problem) is a complex task that must evaluate the solution in terms of the coverage, the energy used, the number of forwarded messages, and the broadcasting time obtained by the optimized protocol in any network configuration.

Therefore, we rely on a simulator to evaluate every solution. The simulator we used is the network simulator ns3 [25], a highly realistic discrete event simulator written in C++ that allows us to deal with mobile ad hoc networks. As an attempt to obtain concluding results in the evaluation of solutions, we simulate every protocol configuration (i.e., every solution) on 10 different networks. The fitness value for every objective is defined as the average of the values obtained for the 10 networks in every objective. We always used the same 10 different seeds in our ns3 simulations to evaluate the solutions.

As we previously stated in Sect. 5, we used the implementation of CellDE and NSGAII provided in jMetal framework [13], and we used the default configurations, which match the ones proposed by the authors of the algorithms. The termination condition of the algorithms was set to $10,000$ evaluations performed. Individuals are encoded as an array of 5 real values, and the values for the integer variables are rounded to the next integer value for evaluation.

These two algorithms try to find the combination of parameters that gives the best possible values for the different objectives. The parameters (defined in Sect. 4) are: (1) minimum value for the delay, (2) maximum value for the delay, (3) value for the border_Threshold, (4) value for the margin_Threshold and (5) value for the neighbors_Threshold.

The objectives (presented also in Sect. 4) are: (1) the energy used, (2) the coverage achieved and (3) the number of forwarded messages. Additionally, the broadcast time is considered as a problem constraint, rejecting those solutions that require more than two seconds in the broadcast process.

To measure the quality of each solution found, the multi-objective algorithms call ns3, giving as input the values of the parameters in that solution, and ns3 returns the values obtained for the different objectives using this configuration. In order to have reliable results, we perform 10 executions of ns3 every time the optimization algorithms check the quality of the solution (this happens $10,000$ times), thus, we analyze the behaviour of AEDB over 10 different networks.

We choose an interval for each parameter in order to find reasonable solutions and limit the search space. These values are shown in Table 1. The algorithm originally creates a set of random feasible solutions (values chosen from the intervals shown in Table 1), and automatically evolves them to better solutions.

Table 1 Domain of the variables to optimize

minimum delay	[0, 1] s
maximum delay	[0, 5] s
border_Threshold	[-95, -70] dBm
margin_Threshold	[0, 3] dBm
neighbors_Threshold	[0, 50]

Regarding the configuration of ns3 for the simulation of the broadcasting algorithm, the mobility model used to emulate the movements of the devices is the *random walk*, also known as *brownian motion mobility model* [21]. In it, nodes move with a randomly chosen speed and direction during a fixed amount of time (20 seconds in our case). After that, other random values for the speed and direction are chosen. The simulation environment used is a square area of 500 m side. The speed of the nodes can vary from 0 to 2m/s (i.e., between 0 and 7.2km/h).

We study different network densities in the optimization process. The first one is a sparse network with 100 devices/km^2, the second one has 200 devices/km^2, and finally the densest one with 300 devices/km^2. All the parameters are summarized in Table 2.

Table 2 Configuration of ns3 for the simulations

Devices/km^2	100-200-300
Speed	[0, 2] m/s
Size of the area	500 m $\times$ 500 m
Default transmission power	16.02 dBm
Direction and speed change	every 20 s

In the simulations, the network evolves for 30 seconds in order to have the nodes uniformly distributed in the area. Then, after these 30 seconds, a node starts the broadcasting process. The simulation stops after 40 seconds.

7 Simulation Results

We present and analyze in this section the results obtained in our experimentation. Section 7.1 shows the results obtained by the optimization algorithms, comparing their performance for the different network densities considered. After that, we analyze the solutions reported by the algorithms in Sect. 7.2. A subset of the solutions contained in the Pareto front that are compromised with the three studied objectives are selected in Sect. 7.3, and a scalability study is performed in Sect. 7.4. Finally, the best compromised solutions from the designer of the protocol point of view are suggested in Sect. 7.5.

7.1 Algorithms Performance Comparison

We compare in this section the performance of the two studied multi-objective algorithms on the optimization of AEDB for the three different network densities considered. To evaluate the quality of the Pareto front approximations computed by the algorithms, several metrics measuring different aspects of the fronts are used, as it is common in the literature. Specifically, we rely on I_ϵ^+, I_Δ, and I_{HV}, measuring the accuracy of the front, the diversity of solutions, and both of them, respectively. They are defined next:

- **Epsilon (I_ϵ^+).** It measures the smallest distance needed to translate every solution in the front so that it dominates the optimal Pareto front of the problem. More formally, given $\mathbf{z^1} = (z_1^1, \ldots, z_n^1)$ and $\mathbf{z^2} = (z_1^2, \ldots, z_n^2)$, where n is the number of objectives:

  $$I_{\epsilon+}^1(A) = inf\left\{\epsilon \in \mathbb{R} | \forall \mathbf{z^2} \in \mathcal{PF}^* \exists \mathbf{z^1} \in A : \mathbf{z^1} \prec_\epsilon \mathbf{z^2}\right\} \quad (2)$$

  where, A is the front to evaluate, $\mathcal{PF}^*$ is the optimal Pareto front for this problem, and $\mathbf{z^1} \prec_\epsilon \mathbf{z^2}$ if and only if $\forall 1 \leq i \leq n : z_i^1 < \epsilon + z_i^2$.
  Fronts with small I_ϵ^+ values are desirable.
- **Spread (I_Δ).** It quantifies the diversity of solutions in the front by means of how well they are spread along the front. It is defined as:

  $$I_\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} \left| d_i - \bar{d} \right|}{d_f + d_l + (N-1)\bar{d}} \quad , \quad (3)$$

  where d_i is the Euclidean distance between consecutive solutions, $\bar{d}$ is the mean of these distances, and d_f and d_l are the Euclidean distances to the *extreme* solutions of the optimal Pareto front in the objective space. This indicator takes value zero for an ideal distribution, which has a perfect spread of the solutions in the Pareto front.
- **Hypervolume (I_{HV}).** This indicator calculates the volume, in the objective space, covered by members of a non-dominated set of solutions Q, for problems where all objectives are to be minimized [40]. Mathematically, for each solution $i \in Q$, a hypercube v_i is constructed with a reference point W and the solution i as the diagonal corners of the hypercube. The reference point can simply be found by constructing a vector of worst objective function values. Thereafter, a union of all hypercubes is found and its hypervolume (HV) is calculated as:

  $$I_{HV} = \text{volume}\left(\bigcup_{i=1}^{|Q|} v_i\right) . \quad (4)$$

  The higher the value of I_{HV}, the better the approximated Pareto front is.

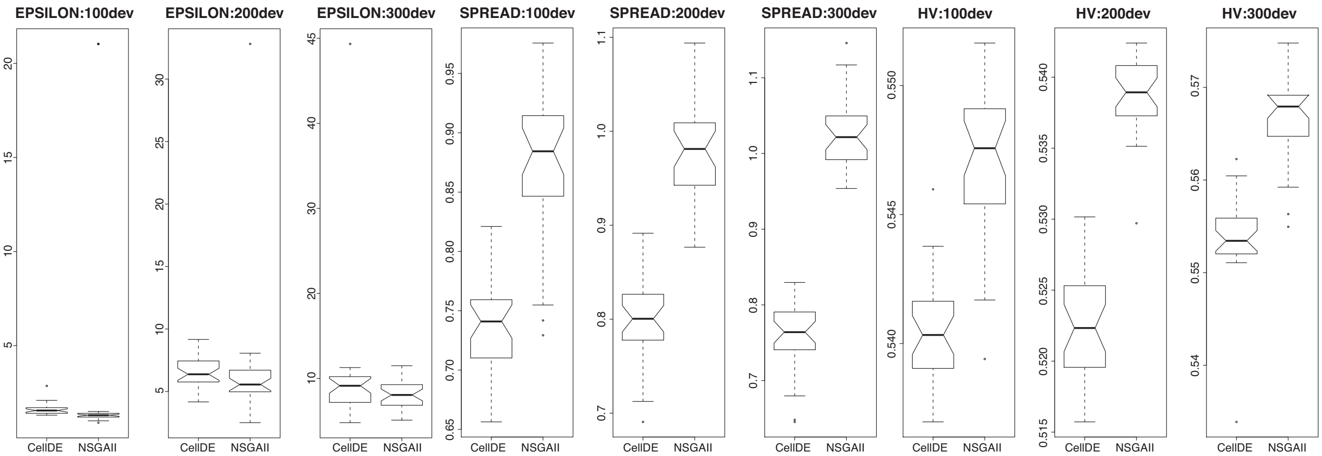
It can be observed in Eq. 2 that the I_ϵ^+ indicator makes use of the optimal Pareto front. Because it is unknown for the considered problem, we build a *reference Pareto front* from the solutions reported by the two algorithms in the 30 independent runs (they are shown in Fig. 2). In order to avoid possible bias in the computation of these indicators due to the different dimensions of the problem objectives, this reference Pareto front is also used to normalize the approximated fronts.

7.1.1 CellDE versus NSGAII

In Fig. 1, we compare the performance of CellDE versus NSGAII for the three problem densities studied according to the three suggested metrics. In order to get strong statistical evidence, these results are computed after performing 30 independent runs of every algorithm for each problem density. In the displayed boxplots, the bottom and top of the boxes represent the lower and upper quartiles of the data distribution, respectively, while the line between them is the median. The whiskers are the lowest datum still within 1.5 IQR of the lower quartile, and the highest datum still within 1.5 IQR of the upper quartile. The circles are data not included between the whiskers. Finally, the notches in the boxes display the variability of the median between samples. If the notches of two boxes are not overlapped, then it means that there are statistical significant differences in the data with 95% confidence.

The results in Fig. 1 show significant differences between the algorithms in all cases but the I_ϵ^+ indicator for the 200 and 300 devices network. We can see that NSGAII is providing more accurate results than CellDE, as the boxplots for I_ϵ^+ and I_{HV} show. Differences are small for I_ϵ^+, and more important in the case of I_{HV}. However, CellDE provides the decision maker with a much broader choice of tradeoff solutions, as the I_Δ plots demonstrate.

Now, we will pay attention on how the algorithms scale with the problem size. The relative performance of the two algorithms is similar for all problem sizes: NSGAII is better for I_ϵ^+ and I_{HV}, and worse for I_Δ. For I_ϵ^+, the boxes are overlapped for the three network densities, meaning that the solutions in the first quartile of CellDE are better than those in the third quartile of NSGAII. Moreover, no significant differences were found between the algorithms for the densest networks. Regarding I_Δ, we can see that the performance of CellDE with respect to NSGAII improves with the problem density, as it can be appreciated by the overlapping degree of the whiskers in the boxplots: very high for the 100 devices problem, almost null for 200 devices one, and far from being overlapped for the densest net-



(a) I_{ϵ}^{+} (b) I_{Δ} (c) I_{HV}

Fig. 1 Comparison of the quality of the fronts computed by the algorithms

work. Finally, we observe that the difference between the two algorithms is lower for the small network than for the other two in the case of I_{HV} too. However, the difference between the algorithms does not increase between 200 and 300 devices problems in this case (indeed, it slightly decreases).

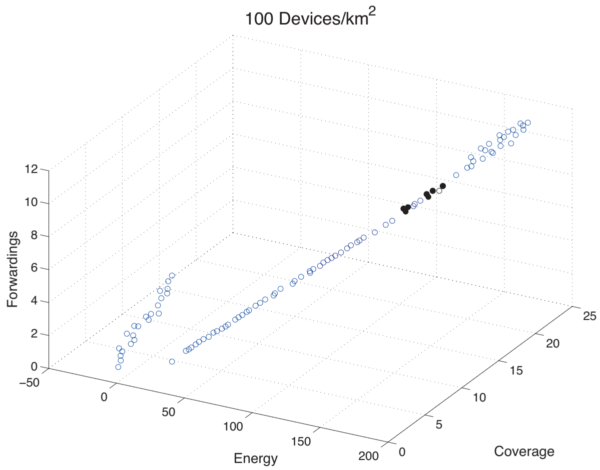
7.2 Studying the Optimization Results

We analyze in this section the quality of the obtained results, and we validate them with the performance of the three original AEDB configurations (three different values of *neighbors_Threshold* parameter were proposed according to the network density). Additionally, we select the five best configurations for every density, and evaluate them when scaling the network size and density.

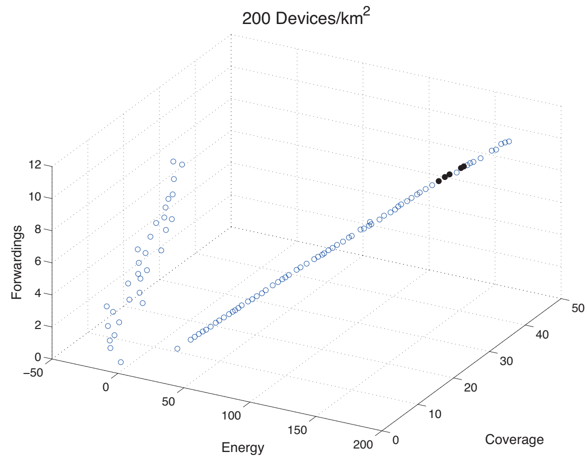
As mentioned before, all the different solutions obtained for each optimization algorithm were considered to build one single Pareto front approximation with the best non-dominated solutions found for every network density. They are displayed in Fig. 2. The maximum size for these fronts was set to 100 solutions, so when more than 100 non-dominated solutions are available, the best 100 ones, according to the crowding distance used in NSGAII, are selected.

In the Pareto front approximations shown in Fig. 2, it stands out that the fronts have two clear sets of solutions in the three scenarios. For the lowest energy values in the approximated range $[-20, 20]$ dBm, solutions provide very low coverage and high number of forwardings, following a linear relationship between these two objectives in which the coverage value is similar to the number of forwardings. These are typically solutions in which devices are only broadcasting the message to their closest one, and therefore the number of forwardings is very close to the number of devices receiving the message (i.e., the coverage). However, for higher energy values over 20 dBm, the shape of the Pareto front changes, and we can see a clearly defined front of solutions in which coverage values are growing much faster than the number of forwardings. This region of the front is the one in which we are more interested, since it is providing high coverage at a reasonable number of forwardings and energy requirements.

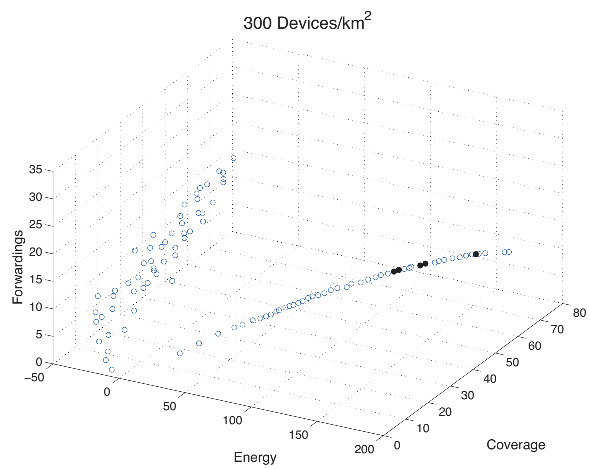
We compared these Pareto front approximations to the solution obtained with the original configurations of AEDB for the three network densities. Looking for fair comparisons, AEDB with the initial settings was executed on 10 different networks using the same seeds as in the optimization algorithms. The average values for each of the objectives is compared to the solutions of the Pareto front. In case, at least one objective of the solution is better than AEDB and better or equal for the rest of the objectives, the solution is said to be dominant. AEDB was executed with values 8, 10, and 12 for *neighbors_Threshold* parameter (as in the original paper) and the best solution among those three ones was compared to the ones obtained in the Pareto front . We found 11, 6 and 1 solutions dominating the original configuration of AEDB (i.e., providing better results for the three objectives) for the three different configurations: 100, 200 and 300 devices/km^2, respectively.



(a) 100 Devices/km^2



(b) 200 Devices/km^2



(c) 300 Devices/km^2

Fig. 2 The reference Pareto fronts obtained after merging all the Pareto front approximations obtained. Black filled circles are the solutions that were chosen in this work according to their performance

7.3 Selecting solutions from the Pareto front

Considering the solutions in the Pareto front approximations, none is better than another (they are all non-dominated). However, from the point of view of the protocol designer, a decision must be taken to choose the most appropriate one according to the expected performance. In this work, we selected some solutions from the Pareto front applying the following restrictions: we consider that the coverage must be at least 80% of the total number of devices, and the number of forwardings should be less than 30%. Moreover, from those solutions that already have less than 30% forwardings and more than 80% coverage, we calculate the percentage of the energy saved. The five solutions with higher energy saved are the ones selected for each network density. In the case of the solutions found for the 300 devices configuration, we restricted the percentage of forwardings to 15% as there are many solutions achieving 80% coverage. All the solutions chosen are represented as black filled circles in Fig. 2, and they were obtained by the NSGAII optimization algorithm.

We give below some general hints of the values of the variables in the selected configurations:

- For the 100 devices configuration: the value of the *borders_Threshold* varies between -90.57 and 91.58 *dBm* for the five solutions. The highest value of the *margin_Threshold* is 0.74 *dBm* and most of the solutions have around value 22 for the *neighbors_Threshold* except one that is 47.
- For the 200 devices configuration: the value of the *borders_Threshold* is around -93.5 *dBm* in general.

The *margin_Threshold* ranges between 0.22 and 0.83 *dBm*. For the *neighbors_Threshold* there is not a common pattern: the values are different in all cases, varying from 11 to 42.
- For the 300 devices configuration: the value of the *borders_Threshold* also varies around -93.5 *dBm* but the values are a bit higher than for the 200 configuration in general. The *margin_Threshold* gets higher values than in the other configurations being the highest value 2.109 *dBm*. However, the values obtained for the *neighbors_Threshold* are lower being the smallest 5 and the highest 28 (only one solution has such a high value).

7.4 Scalability analysis of the selected solutions

For better analyzing these solutions and studying their scalability, we execute each of those 15 solutions in 6 different densities and 3 different scenarios for 100 times in order to get confident results. The densities considered are: 50, 100, 200, 300, 400 and 500 devices/km^2. The three different scenarios are:

1. 500 m × 500 m → 0.25 km^2 (original scenario);
2. 500 m × 1000 m → 0.5 km^2 (medium scenario);
3. 1000 m × 1000 m → 1 km^2 (largest scenario).

Being the first one the original scenario used in the optimization process, the second one double the size of the first, and the third one is four times bigger than the original scenario.

7.4.1 Performance of the selected solutions

All the tables of results concerning these experiments are compiled in the Appendix A. Below, we analyze the general behaviour of each solution in the different densities and scenarios. We first study the differences between the set of solutions of each density used by the optimization algorithms, i.e. solutions obtained for the 100, 200 and 300 devices/km^2 configurations.

The results obtained regarding the percentage of the coverage achieved are shown in tables 4, 5 and 6. We can observe that the set of solutions of the 100 devices configuration is the one that behaves always better, for any scenario or density. Comparing this set of solutions in terms of the different scenarios we can see that the coverage achieved is higher for the 500 m × 500 m (the one used for the optimization), and similar for the other two. Even though the third scenario has double size and the second scenario is four times bigger than the original, the coverage is not half or one quarter. In fact, the set of solutions is scaling well and for 100 devices there is only a maximum of 18.94% and 13.62% difference with the second and third scenarios respectively. And for 50 devices the maximum difference is 28.58% in case of the biggest scenario and 17.52% for the medium scenario. The other two configurations (200 and 300 devices) do not behave as good as the 100 devices configuration in terms of the coverage achieved.

The percentage of forwardings per device reached is shown in tables 7, 8, and 9. It is computed considering the number of devices reached is 100% of the potential forwarders. Therefore, it is the percentage of those devices that received the broadcasted message and forwarded it. The solutions of 300 devices send less forwardings in general. Among the three different scenarios, the behaviour is similar. The solutions scale very well considering the number of forwardings. As expected, the number of forwardings decreases as the density increases.

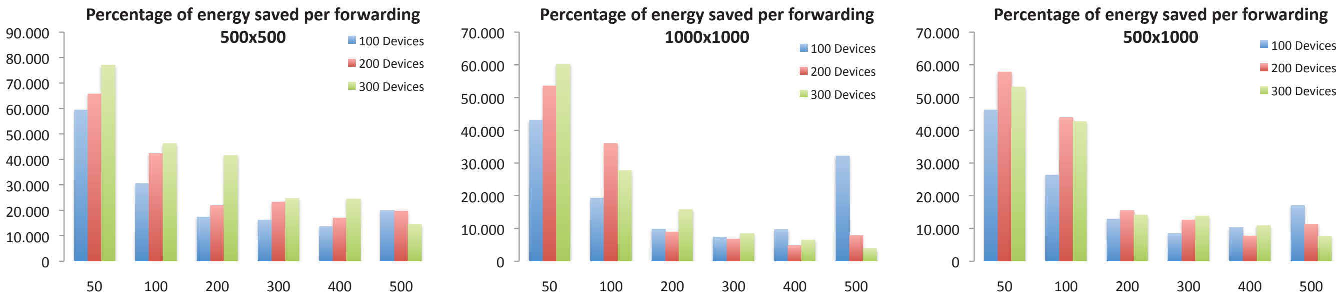
In terms of the energy savings, we show in tables 13, 14 and 15 the percentage of energy saved per forwarding. The energy saved is calculated as:

$$EgSaved = \frac{\#forwardings * DefTx - EgUsed}{\#forwardings} \qquad (5)$$

That is, the difference between the energy used in case all the nodes sending the message are using the default transmission power and the actual energy used by the protocol (in miliWatts), divided by the number of forwardings.

In order to better understand the behavior of the protocol for the different configurations, we include a graph (see Fig. 3) representing the mean value of the percentage of the energy saved for the 5 solutions in each density and scenario. This was obtained by averaging the values shown in tables 13, 14, and 15.

We can observe in Fig. 3 that, for the highest densities in the three scenarios, the solutions obtained for the 100 devices configuration are behaving better because, in general, they are using less energy (see tables 10, 11 and 12) and more forwardings (tables 7, 8, 9). For the sparsest densities, the 300 devices configuration is the one that saves more energy but we must remark that it is also the one with lowest coverage. As mentioned in Sect. 7.3, the 300 devices configuration has around -93.5 *dBm* for the *borders_Threshold* (small forwarding area), low *neighbors_Threshold* over 5 (always reduce the transmission power), and 2.109 *dBm* for the *margin_Threshold*. All these values mean that even in sparse networks, as the *neighbors_Threshold* is very low, the protocol reduces the transmission power, thus, saving more energy but reaching less devices. In dense networks, the high value of the *margin_Threshold* makes



(a) 500×500 scenario (b) 1000×1000 scenario (c) 500×1000 scenario

Fig. 3 Energy saved per forwarding for each of the scenarios studied

Table 3 Domain of the variables of the chosen solutions

	min. delay	*max. delay*	*borders_Threshold*	*margin_Threshold*	*neighbors_Threshold*
Sol1	0.09275885919476065	0.8193490129680125	−90.5793140592024	0.3923234188788317	24.665973963478013
Sol2	0.09275885919476065	0.9170220922018384	−90.5793140592024	0.20311624654997962	21.82887963879373
Sol3	0.41699935691906936	0.6144875063114115	−90.672110892589	0.07548924788699646	21.678919262045863

the energy saving very low (see values for 500 devices in Table 15).

We have also included in the Appendix A the tables regarding the broadcasting time. Tables 16, 17 and 18 show the time needed to broadcast a message in the network for the six densities at hands and for each of the three configurations studied in this work. As we explained before, the broadcast time is not optimized in this work but it is considered as a constrain by the optimization algorithms. We explained in Section 4, that during the optimization process, the algorithms consider a solution is not feasible if its broadcast time is higher than 2 seconds. This value was obtained from previous work [34]. The results show that none of the solutions are over 2 s when studying the original scenario of 500m × 500m (except Sol2 in the 300 devices configuration for the highest density). Therefore, the solutions are scaling well when dealing with lower or higher densities. As expected, for the other two larger scenarios (i.e., 500m × 1000m and 1000m × 1000m), this restriction generally failed for all solutions of the three optimized configurations. In general the larger the scenario, the longer the broadcasting time.

7.5 Best scalable solutions

As we previously mentioned, the result of any multi-objective technique is not an specific value but a set of them that makes the dissemination algorithm promote one objective or another. As the designer of the broadcasting protocol, we are looking for a value for each threshold that generally makes the algorithm behaving better than other values in any kind of scenario for the three objectives. For example, obtaining high savings in energy but poor coverage is not a good solution from the designer point of view. Therefore, in this work, we try to find a tradeoff between the different objectives.

We suggest to use the solutions obtained for the 100 devices configuration. We consider that the higher number of forwardings and lower energy savings are sacrificed to guarantee higher coverage in sparse networks where the dissemination of the message is difficult. For the highest densities, the number of forwardings are reasonable (highest difference with 300 devices configuration is 5.1%) and the energy savings show the same or even better performance than any other configuration.

Considering now the set of solutions selected in the 100 devices configuration, we can see that Sol1, Sol2 or Sol3 show, in general, the best overall behavior. The values of the variables of those three solutions are presented in Table 3.

As we can see, the values of the variables are similar. The main difference is found in the values of the delay between Sol3 and the other two. The interval is smaller, but the minimum delay is 0.32424 units higher. That means, the broadcast time will take longer as the other solutions have a probability of having the value of the random delay lower than 0.417. This is confirmed in Table 16.

Also notice that the value of the *margin_Threshold* of Sol3 is lower than Sol1 or Sol2, but it is not reflected in the energy savings (obtaining higher values). The reason is because it is compensated with the slightly higher value of the *borders_Threshold*.

8 Conclusion and Future Work

In this work, we have optimized the Adaptive Enhanced Distance Based broadcasting algorithm (AEDB) using two different multi-objective optimization algorithms from the state of the art: CellDE and NSGAII. The set of parameters of AEDB have been optimized in terms of three different objectives: (1) the energy used for the dissemination process; (2) the coverage achieved by the message, and (3) the number of forwardings used to get the mentioned coverage. From a previous study [34], we can establish an upper bound for the broadcast time, 2 s. Therefore, we have included a restriction in our optimization process. The set of parameters proposed by the optimization algorithm is only a feasible solution if the broadcasting time is lower than 2 s. Otherwise, the solution is discarded.

We optimized AEDB in a square scenario of 500 m side, for three different densities: 100, 200 and 300 devices/km^2, for each of the optimization algorithms. The solutions obtained of each optimization algorithm were merged in a unique *Pareto front* for each density. A comparison between the performance of both optimization algorithms was done, paying attention on how they scale with the network density.

From the point of view of the optimization, there is no solution better than another among those in the Pareto front. However, from the point of view of the protocol designer, a decision must be taken in order to have a compromise for each objective, i.e., a solution with minimum energy used but very low coverage (or high coverage and high number of forwardings) has no sense for a broadcasting algorithm. Therefore, five solutions of each *Pareto front* were selected for further analysis.

The scalability capabilities of those solutions were studied. For that, they were executed in 6 densities: 50, 100, 200, 300, 400 and 500 devices/km^2. And in 3 different scenarios. The original one with 0.25 km^2, another with double size (0.5 km^2), and a third one four times larger (1 km^2). All the solutions were executed in 100 different networks in order to get confident results.

A deep study was performed in order to analyze the scalability of the different solutions in terms of: (1) the percentage of the coverage achieved; (2) the percentage of forwardings per device reached in the broadcasting process; and (3) the percentage of energy used and saved both in miliWatts. Finally, the broadcast time was also included in the analysis.

From all the results, we suggested that the set of solutions that generally behave and scale better were the ones found in the Pareto front approximation obtained for 100 devices/km^2 density. Among the five solutions selected, 3 were identified as the best overall ones.

It is necessary to highlight that except for the neighbors_Threshold, the other values obtained are quite similar to the original ones used in AEDB (i.e., AEDB was already quite well fine tuned). Therefore, it is pointed out as a decisive parameter on the performance of the protocol.

As future work, we plan to make the parameters adaptive, so that, the protocol itself will change the value of the neighbors_threshold or any other value in terms of, for example, the network density.

A Appendix

The tables obtained for the 6 different densities in each of the three different scenarios studied are presented in this appendix. We are showing only the tables regarding: (1) the percentage of the coverage achieved; (2) the percentage of the number of forwardings per device reached, (3) the energy used per forwarding in miliWatts; (4) the percentage of the energy saved per forwarding also in miliWatts; and finally (5) the broadcast time.

The results obtained using the solutions of the optimization algorithms for 100, 200, and 300 devices/km^2 for the metrics mentioned above are shown in that order.

Acknowledgements This work was completed with the support of Luxembourg FNR GreenIT project (C09/IS/05).

References

1. Abdullah, J., Parish, D.J.: Node connectivity index as mobility metric for GA based QoS routing in MANET. In: Mobility 2007: Proceedings of the 4th international conference on mobile technology, applications and systems and the 1st international symposium on Computer human interaction in mobile technology, pp. 104–111. ACM (2007)
2. Alba, E., Bouvry, P., Dorronsoro, B., Luna, F., Nebro, A.: A cellular multi-objective genetic algorithm for optimal broadcasting strategy in metropolitan MANETs. In: Nature Inspired Distributed Computing (NIDISC) sessions of the (IPDPS) Workshop, p. 192b (2005)
3. Alba, E., Dorronsoro, B.: Cellular Genetic Algorithms. Operations Research/Compuer Science Interfaces. Springer-Verlag Heidelberg (2008)
4. Alba, E., Dorronsoro, B., Luna, F., Nebro, A., Bouvry, P., Hogie, L.: A cellular multi-objective genetic algorithm for optimal broadcasting strategy in metropolitan MANETs. Computer Communications **30**(4), 685–697 (2007)
5. Bäck, T., Fogel, D., Michalewicz, Z. (eds.): Handbook of Evolutionary Computation. Oxford University Press (1997)
6. Cagalj, M., Hubaux, J.P., Enz, C.: Minimum-energy broadcast in all-wireless networks: Np-completeness and distribution issues. In: MobiCom '02: Proceedings of the 8th annual international conference on Mobile computing and networking, pp. 172–182. ACM (2002)

Table 4 Percentage of the coverage achieved for the 100 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	$45.33_{\pm 25.77}$	$74.96_{\pm 23.01}$	$95.44_{\pm 7.25}$	$98.00_{\pm 2.45}$	$98.82_{\pm 0.61}$	$99.13_{\pm 0.38}$
500_500_Sol2	$47.25_{\pm 28.13}$	$73.56_{\pm 23.41}$	$95.36_{\pm 6.87}$	$98.41_{\pm 0.77}$	$98.76_{\pm 1.83}$	$99.10_{\pm 0.48}$
500_500_Sol3	$48.08_{\pm 27.06}$	$77.24_{\pm 24.19}$	$92.42_{\pm 13.73}$	$98.08_{\pm 1.46}$	$98.91_{\pm 0.55}$	$99.17_{\pm 0.19}$
500_500_Sol4	$41.17_{\pm 25.37}$	$68.84_{\pm 23.46}$	$90.96_{\pm 14.48}$	$96.44_{\pm 8.37}$	$98.63_{\pm 1.76}$	$99.02_{\pm 0.95}$
500_500_Sol5	$48.50_{\pm 24.91}$	$74.00_{\pm 24.10}$	$93.20_{\pm 11.82}$	$97.91_{\pm 2.68}$	$98.51_{\pm 2.02}$	$99.13_{\pm 0.30}$
1000_1000_Sol1	$25.40_{\pm 19.49}$	$65.23_{\pm 30.64}$	$97.95_{\pm 3.90}$	$99.40_{\pm 0.84}$	$99.71_{\pm 0.16}$	$99.80_{\pm 0.00}$
1000_1000_Sol2	$23.60_{\pm 17.25}$	$65.00_{\pm 32.67}$	$98.48_{\pm 2.05}$	$99.54_{\pm 0.49}$	$99.72_{\pm 0.23}$	$99.78_{\pm 0.14}$
1000_1000_Sol3	$22.64_{\pm 19.20}$	$63.11_{\pm 32.64}$	$96.62_{\pm 12.52}$	$99.39_{\pm 1.40}$	$99.70_{\pm 0.29}$	$99.80_{\pm 0.02}$
1000_1000_Sol4	$19.20_{\pm 15.80}$	$49.90_{\pm 30.00}$	$95.33_{\pm 5.71}$	$99.14_{\pm 1.07}$	$99.54_{\pm 0.53}$	$99.74_{\pm 0.20}$
1000_1000_Sol5	$19.92_{\pm 17.59}$	$63.45_{\pm 29.96}$	$97.88_{\pm 2.92}$	$99.38_{\pm 0.68}$	$99.67_{\pm 0.25}$	$99.74_{\pm 0.22}$
500_1000_Sol1	$30.12_{\pm 22.22}$	$62.96_{\pm 29.34}$	$95.83_{\pm 10.01}$	$98.90_{\pm 1.11}$	$99.44_{\pm 0.18}$	$99.56_{\pm 0.20}$
500_1000_Sol2	$29.60_{\pm 22.80}$	$67.82_{\pm 27.63}$	$96.63_{\pm 7.24}$	$98.96_{\pm 1.06}$	$99.47_{\pm 0.25}$	$99.57_{\pm 0.19}$
500_1000_Sol3	$30.56_{\pm 20.39}$	$65.10_{\pm 27.59}$	$94.88_{\pm 12.61}$	$99.11_{\pm 0.64}$	$99.40_{\pm 0.38}$	$99.56_{\pm 0.18}$
500_1000_Sol4	$29.12_{\pm 20.79}$	$55.22_{\pm 28.47}$	$93.27_{\pm 13.07}$	$98.15_{\pm 2.63}$	$99.38_{\pm 0.33}$	$99.51_{\pm 0.35}$
500_1000_Sol5	$31.88_{\pm 21.52}$	$62.82_{\pm 26.59}$	$90.68_{\pm 16.12}$	$98.34_{\pm 5.70}$	$99.32_{\pm 0.58}$	$99.52_{\pm 0.41}$

Table 5 Percentage of the coverage achieved for the 200 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	$38.33_{\pm 20.75}$	$51.40_{\pm 23.53}$	$76.82_{\pm 19.92}$	$81.39_{\pm 22.09}$	$88.90_{\pm 20.43}$	$94.86_{\pm 11.60}$
500_500_Sol2	$38.17_{\pm 23.03}$	$54.36_{\pm 24.46}$	$76.72_{\pm 22.54}$	$91.23_{\pm 13.65}$	$96.86_{\pm 4.66}$	$96.32_{\pm 10.42}$
500_500_Sol3	$38.17_{\pm 19.72}$	$49.32_{\pm 22.65}$	$71.74_{\pm 21.68}$	$84.61_{\pm 20.78}$	$92.08_{\pm 13.12}$	$95.23_{\pm 10.33}$
500_500_Sol4	$37.42_{\pm 21.86}$	$48.92_{\pm 21.65}$	$71.50_{\pm 23.72}$	$81.28_{\pm 21.43}$	$88.14_{\pm 15.90}$	$96.51_{\pm 5.29}$
500_500_Sol5	$36.75_{\pm 21.26}$	$50.32_{\pm 20.49}$	$64.42_{\pm 24.23}$	$77.08_{\pm 21.09}$	$84.79_{\pm 18.74}$	$90.14_{\pm 17.65}$
1000_1000_Sol1	$14.22_{\pm 10.48}$	$23.81_{\pm 16.20}$	$55.31_{\pm 26.90}$	$80.56_{\pm 24.35}$	$94.35_{\pm 11.78}$	$97.98_{\pm 8.84}$
1000_1000_Sol2	$14.52_{\pm 11.03}$	$28.52_{\pm 20.37}$	$72.33_{\pm 27.01}$	$92.63_{\pm 15.61}$	$97.94_{\pm 9.29}$	$99.21_{\pm 1.19}$
1000_1000_Sol3	$13.42_{\pm 8.49}$	$25.88_{\pm 16.61}$	$58.50_{\pm 27.37}$	$86.96_{\pm 18.65}$	$94.56_{\pm 13.23}$	$98.35_{\pm 4.01}$
1000_1000_Sol4	$16.48_{\pm 10.90}$	$21.46_{\pm 14.07}$	$46.02_{\pm 27.42}$	$77.92_{\pm 26.35}$	$87.98_{\pm 22.80}$	$95.92_{\pm 10.48}$
1000_1000_Sol5	$14.36_{\pm 9.09}$	$19.07_{\pm 11.36}$	$37.30_{\pm 23.65}$	$68.61_{\pm 26.31}$	$81.80_{\pm 24.06}$	$94.49_{\pm 13.94}$
500_1000_Sol1	$22.00_{\pm 13.50}$	$33.34_{\pm 18.59}$	$61.23_{\pm 25.18}$	$78.61_{\pm 24.57}$	$88.39_{\pm 19.21}$	$93.28_{\pm 13.65}$
500_1000_Sol2	$27.20_{\pm 14.95}$	$38.40_{\pm 21.73}$	$70.32_{\pm 23.10}$	$90.15_{\pm 16.40}$	$97.04_{\pm 4.87}$	$97.10_{\pm 11.04}$
500_1000_Sol3	$23.68_{\pm 15.30}$	$30.02_{\pm 16.49}$	$55.91_{\pm 28.05}$	$80.22_{\pm 21.47}$	$85.52_{\pm 24.30}$	$96.62_{\pm 6.81}$
500_1000_Sol4	$21.40_{\pm 12.49}$	$31.26_{\pm 16.76}$	$51.52_{\pm 26.96}$	$76.83_{\pm 22.87}$	$82.68_{\pm 25.24}$	$94.47_{\pm 13.19}$
500_1000_Sol5	$24.04_{\pm 12.85}$	$31.74_{\pm 16.80}$	$43.11_{\pm 23.18}$	$61.67_{\pm 26.56}$	$79.22_{\pm 24.30}$	$90.36_{\pm 17.54}$

Table 6 Percentage of the coverage achieved for the 300 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	$33.67_{\pm 19.57}$	$44.88_{\pm 19.26}$	$58.82_{\pm 22.33}$	$71.49_{\pm 22.72}$	$85.70_{\pm 16.91}$	$88.70_{\pm 17.15}$
500_500_Sol2	$35.50_{\pm 20.05}$	$45.64_{\pm 17.98}$	$56.62_{\pm 25.01}$	$77.36_{\pm 22.45}$	$82.51_{\pm 21.36}$	$86.21_{\pm 21.34}$
500_500_Sol3	$31.17_{\pm 21.04}$	$47.72_{\pm 20.24}$	$60.42_{\pm 26.78}$	$78.57_{\pm 21.18}$	$83.26_{\pm 21.33}$	$89.95_{\pm 16.23}$
500_500_Sol4	$32.42_{\pm 21.38}$	$44.00_{\pm 18.92}$	$57.34_{\pm 19.40}$	$74.39_{\pm 22.11}$	$74.58_{\pm 23.62}$	$81.06_{\pm 23.38}$
500_500_Sol5	$35.42_{\pm 21.92}$	$53.52_{\pm 27.18}$	$80.82_{\pm 23.51}$	$92.28_{\pm 15.37}$	$96.46_{\pm 8.65}$	$97.70_{\pm 5.36}$
1000_1000_Sol1	$12.40_{\pm 7.98}$	$15.58_{\pm 9.79}$	$31.50_{\pm 19.74}$	$52.81_{\pm 28.90}$	$73.20_{\pm 28.57}$	$91.43_{\pm 13.97}$
1000_1000_Sol2	$12.02_{\pm 7.25}$	$18.46_{\pm 10.69}$	$37.14_{\pm 23.03}$	$60.43_{\pm 29.94}$	$81.98_{\pm 22.25}$	$92.02_{\pm 17.90}$
1000_1000_Sol3	$12.74_{\pm 8.10}$	$17.64_{\pm 11.45}$	$37.60_{\pm 20.74}$	$64.77_{\pm 29.67}$	$88.43_{\pm 17.96}$	$92.65_{\pm 17.80}$
1000_1000_Sol4	$12.50_{\pm 7.45}$	$17.54_{\pm 9.74}$	$30.17_{\pm 21.31}$	$48.16_{\pm 27.15}$	$68.32_{\pm 28.96}$	$80.93_{\pm 22.57}$
1000_1000_Sol5	$15.38_{\pm 11.64}$	$28.32_{\pm 21.89}$	$72.88_{\pm 29.52}$	$91.90_{\pm 19.45}$	$97.75_{\pm 9.08}$	$99.60_{\pm 0.52}$
500_1000_Sol1	$22.76_{\pm 12.89}$	$25.14_{\pm 14.20}$	$36.93_{\pm 19.89}$	$59.93_{\pm 26.01}$	$71.97_{\pm 24.99}$	$81.99_{\pm 23.92}$
500_1000_Sol2	$24.32_{\pm 13.27}$	$28.40_{\pm 16.59}$	$43.18_{\pm 23.31}$	$62.42_{\pm 27.30}$	$76.56_{\pm 25.12}$	$88.70_{\pm 18.48}$
500_1000_Sol3	$21.76_{\pm 13.31}$	$25.08_{\pm 15.80}$	$47.34_{\pm 23.18}$	$66.15_{\pm 27.62}$	$75.78_{\pm 26.21}$	$89.39_{\pm 19.94}$
500_1000_Sol4	$23.44_{\pm 13.84}$	$27.92_{\pm 14.91}$	$37.62_{\pm 19.67}$	$56.90_{\pm 23.75}$	$66.13_{\pm 22.80}$	$78.93_{\pm 23.90}$
500_1000_Sol5	$26.08_{\pm 14.87}$	$37.94_{\pm 23.80}$	$71.40_{\pm 25.60}$	$90.65_{\pm 17.67}$	$97.18_{\pm 8.24}$	$98.09_{\pm 8.72}$

7. Cagalj, M., Hubaux, J.P., Enz, C.C.: Energy-efficient broadcasting in all-wireless networks. Wirel. Netw. **11**(1-2), 177–188 (2005)
8. Chen, X., Faloutsos, M., Krishnamurthy, S.V.: Power adaptive broadcasting with local information in ad hoc networks. In: Conference on Network Protocols, p. 168. IEEE Computer Society (2003)
9. Chiang, T., Liu, C., Huang, Y.: A near-optimal multicast scheme for mobile ad hoc networks using a hybrid genetic algorithm. Expert Systems with Applications **33**(3), 734–742 (2007)
10. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation **6**(2), 182–197 (2002)
11. Di Caro, G.A., Ducatelle, F., Gambardella, L.M.: AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks. Eur. Trans. on Telecom. **16**(5), 443–455 (2005)
12. Durillo, J., Nebro, A., Luna, F., Alba, E.: Solving three-objective optimization problems using a new hybrid cellular genetic algorithm. In: Parallel Problem Solving from Nature (PPSN X), *LNCS*, vol. 5199, pp. 661–670. Springer (2008)
13. Durillo, J.J., Nebro, A.J.: jMetal: A java framework for multi-objective optimization. Advances in Engineering Software **42**, 760–771 (2011)
14. Durillo, J.J., Nebro, A.J., Luna, F., Alba, E.: A study of master-slave approaches to parallelize NSGA-II. In: Nature Inspired Distributed Computing (NIDISC) Workshop of the (IPDPS), p. 11 (2008)

Table 7 Percentage of the number of forwardings used per device reached (i.e., coverage) for the 100 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	52.76	35.70	21.96	16.26	12.73	10.69
500_500_Sol2	53.26	34.64	22.57	16.38	12.80	10.97
500_500_Sol3	50.09	35.42	22.55	16.71	12.99	11.53
500_500_Sol4	43.12	30.33	19.15	14.18	11.33	9.47
500_500_Sol5	46.91	32.32	21.22	14.97	12.03	9.82
1000_1000_Sol1	49.53	34.42	21.68	15.44	12.21	4.98
1000_1000_Sol2	48.81	35.48	21.77	15.52	12.58	11.11
1000_1000_Sol3	47.44	35.65	21.78	16.01	12.85	5.36
1000_1000_Sol4	41.15	29.28	19.17	13.91	10.84	9.11
1000_1000_Sol5	44.38	32.69	20.55	14.69	11.64	9.47
500_1000_Sol1	48.61	34.40	21.88	16.00	12.75	10.59
500_1000_Sol2	46.89	35.98	22.23	15.99	12.88	11.19
500_1000_Sol3	51.83	35.73	22.51	16.43	13.06	11.35
500_1000_Sol4	41.21	30.39	19.68	14.19	11.23	9.37
500_1000_Sol5	48.31	33.24	20.70	15.27	11.84	9.72

Table 8 Percentage of the number of forwardings used per device reached (i.e., coverage) for the 200 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	34.35	20.31	13.72	9.94	8.64	7.08
500_500_Sol2	35.37	22.08	14.52	11.66	9.19	7.71
500_500_Sol3	35.81	19.87	13.58	10.46	8.51	7.28
500_500_Sol4	33.41	18.32	13.03	10.14	7.96	7.04
500_500_Sol5	33.33	18.12	11.80	9.41	7.57	6.52
1000_1000_Sol1	29.96	19.57	13.40	10.45	8.43	7.02
1000_1000_Sol2	32.78	22.09	15.13	11.45	9.12	7.56
1000_1000_Sol3	28.32	20.60	13.96	10.85	8.63	7.36
1000_1000_Sol4	29.25	19.25	12.80	10.18	8.18	6.91
1000_1000_Sol5	26.60	17.41	11.73	9.57	7.78	6.66
500_1000_Sol1	29.82	19.26	14.06	10.41	8.58	7.15
500_1000_Sol2	32.79	22.19	15.05	11.59	9.23	7.72
500_1000_Sol3	31.25	19.12	13.56	10.66	8.74	7.47
500_1000_Sol4	28.04	19.77	12.99	10.00	8.36	7.08
500_1000_Sol5	29.62	19.41	11.20	9.27	7.97	6.69

Table 9 Percentage of the number of forwardings used per device reached (i.e., coverage) for the 300 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	27.48	14.80	10.47	8.50	7.18	6.30
500_500_Sol2	25.12	15.25	10.67	9.20	7.42	6.32
500_500_Sol3	32.09	16.93	11.82	9.16	7.59	6.45
500_500_Sol4	31.11	15.27	10.50	8.48	6.70	6.16
500_500_Sol5	36.94	22.65	15.84	11.86	9.38	8.07
1000_1000_Sol1	24.35	14.76	10.87	8.84	7.36	6.34
1000_1000_Sol2	26.62	15.55	11.32	9.25	7.63	6.59
1000_1000_Sol3	25.75	15.93	11.99	9.35	7.90	6.68
1000_1000_Sol4	24.16	15.45	10.71	8.42	7.18	6.17
1000_1000_Sol5	30.56	22.67	15.76	11.90	9.40	7.89
500_1000_Sol1	25.31	15.19	10.32	8.83	7.34	6.31
500_1000_Sol2	26.64	16.97	11.30	9.26	7.69	6.63
500_1000_Sol3	27.94	17.70	11.47	9.65	7.83	6.68
500_1000_Sol4	26.28	17.26	9.91	8.48	7.03	6.11
500_1000_Sol5	30.52	21.98	15.29	12.02	9.52	8.10

15. García, S., Luque, C., Cervantes, A., Galván, I.: Multiobjective algorithms hybridization to optimize broadcasting parameters in mobile ad-hoc networks. In: Proceedings of the 10th International Work-Conference on Artificial Neural Networks Part I: Bio-Inspired Systems: Computational and Ambient Intelligence, *Lecture Notes in Computer Science*, vol. 5517, pp. 728–735. Springer-Verlag Heidelberg (2009)
16. García, S., Luque, C., Cervantes, A., Galván, I.: Multiobjective algorithms hybridization to optimize broadcasting parameters in mobile ad-hoc networks. In: Proceedings of the 10th International Work-Conference on Artificial Neural Networks: Part I: Bio-Inspired Systems: Computational and Ambient Intelligence, pp. 728–735 (2009)
17. García-Nieto, J., Alba, E.: Automatic parameter tuning with metaheuristics of the AODV routing protocol for vehicular ad-hoc networks. In: Applications of Evolutionary Computation, EvoApplications 2010, *LNCS*, vol. 6025, pp. 21–30. Springer-Verlag Heidelberg (2010)
18. García-Nieto, J., Toutouh, J., Alba, E.: Automatic tuning of communication protocols for vehicular ad hoc networks using metaheuristics. Eng. App. of Artificial Intelligence

Table 10 Energy used per forwarding (miliWatts) for the 100 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	$19.50_{\pm 15.18}$	$28.29_{\pm 10.96}$	$31.59_{\pm 8.06}$	$37.93_{\pm 5.07}$	$34.89_{\pm 4.49}$	$31.10_{\pm 4.33}$
500_500_Sol2	$11.78_{\pm 8.68}$	$25.10_{\pm 10.71}$	$33.78_{\pm 8.03}$	$29.77_{\pm 3.31}$	$34.02_{\pm 4.13}$	$32.15_{\pm 4.24}$
500_500_Sol3	$20.53_{\pm 7.06}$	$29.55_{\pm 12.56}$	$30.06_{\pm 7.98}$	$31.03_{\pm 4.78}$	$35.72_{\pm 4.94}$	$29.29_{\pm 3.72}$
500_500_Sol4	$11.11_{\pm 12.03}$	$23.68_{\pm 10.19}$	$34.55_{\pm 7.71}$	$31.31_{\pm 8.31}$	$31.28_{\pm 3.92}$	$35.97_{\pm 3.80}$
500_500_Sol5	$18.10_{\pm 10.10}$	$32.45_{\pm 10.06}$	$35.12_{\pm 7.13}$	$37.67_{\pm 4.15}$	$36.98_{\pm 3.64}$	$31.51_{\pm 3.80}$
1000_1000_Sol1	$21.78_{\pm 17.26}$	$32.80_{\pm 25.20}$	$36.54_{\pm 8.60}$	$36.28_{\pm 9.05}$	$37.76_{\pm 9.78}$	$16.75_{\pm 3.83}$
1000_1000_Sol2	$25.46_{\pm 17.36}$	$29.83_{\pm 26.01}$	$36.99_{\pm 13.56}$	$36.44_{\pm 6.39}$	$34.18_{\pm 5.83}$	$29.62_{\pm 6.72}$
1000_1000_Sol3	$20.03_{\pm 18.64}$	$31.39_{\pm 23.01}$	$34.24_{\pm 16.81}$	$37.04_{\pm 9.44}$	$34.28_{\pm 6.20}$	$15.46_{\pm 5.06}$
1000_1000_Sol4	$26.95_{\pm 17.29}$	$34.09_{\pm 26.29}$	$37.44_{\pm 15.26}$	$37.52_{\pm 7.64}$	$36.96_{\pm 10.48}$	$36.52_{\pm 8.54}$
1000_1000_Sol5	$19.80_{\pm 13.94}$	$33.43_{\pm 20.54}$	$35.01_{\pm 14.29}$	$38.09_{\pm 7.67}$	$37.41_{\pm 10.71}$	$37.30_{\pm 10.17}$
500_1000_Sol1	$18.39_{\pm 15.10}$	$32.36_{\pm 19.25}$	$37.42_{\pm 10.48}$	$37.31_{\pm 8.05}$	$35.53_{\pm 5.54}$	$33.54_{\pm 6.71}$
500_1000_Sol2	$15.81_{\pm 10.34}$	$26.82_{\pm 17.82}$	$34.02_{\pm 10.37}$	$36.63_{\pm 5.77}$	$35.24_{\pm 6.22}$	$32.30_{\pm 5.52}$
500_1000_Sol3	$23.36_{\pm 15.83}$	$29.94_{\pm 17.61}$	$32.77_{\pm 14.23}$	$34.68_{\pm 7.35}$	$36.55_{\pm 7.74}$	$30.22_{\pm 6.94}$
500_1000_Sol4	$27.82_{\pm 12.51}$	$29.30_{\pm 21.23}$	$34.43_{\pm 14.74}$	$38.80_{\pm 8.29}$	$35.80_{\pm 6.29}$	$34.93_{\pm 6.73}$
500_1000_Sol5	$22.20_{\pm 15.13}$	$28.97_{\pm 15.21}$	$35.69_{\pm 12.72}$	$35.64_{\pm 7.68}$	$36.27_{\pm 6.27}$	$35.02_{\pm 5.83}$

Table 11 Energy used per forwarding (miliWatts) for the 200 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	$14.42_{\pm 10.23}$	$24.50_{\pm 9.26}$	$25.93_{\pm 6.71}$	$25.48_{\pm 10.16}$	$34.48_{\pm 10.35}$	$31.64_{\pm 6.95}$
500_500_Sol2	$12.77_{\pm 8.07}$	$34.82_{\pm 10.76}$	$31.20_{\pm 8.84}$	$39.25_{\pm 7.73}$	$38.37_{\pm 4.81}$	$31.34_{\pm 7.00}$
500_500_Sol3	$17.58_{\pm 6.99}$	$20.84_{\pm 7.21}$	$35.65_{\pm 8.27}$	$33.40_{\pm 9.90}$	$36.87_{\pm 10.17}$	$34.45_{\pm 8.44}$
500_500_Sol4	$13.89_{\pm 8.33}$	$17.24_{\pm 6.28}$	$30.60_{\pm 8.60}$	$26.44_{\pm 9.29}$	$26.31_{\pm 10.62}$	$32.69_{\pm 6.18}$
500_500_Sol5	$9.99_{\pm 7.46}$	$18.14_{\pm 5.37}$	$32.80_{\pm 7.59}$	$28.97_{\pm 9.04}$	$30.17_{\pm 8.00}$	$30.35_{\pm 7.22}$
1000_1000_Sol1	$11.93_{\pm 12.31}$	$29.86_{\pm 16.89}$	$37.23_{\pm 21.15}$	$35.52_{\pm 24.74}$	$38.70_{\pm 21.18}$	$37.34_{\pm 15.40}$
1000_1000_Sol2	$15.74_{\pm 10.21}$	$27.69_{\pm 18.37}$	$38.87_{\pm 23.08}$	$39.09_{\pm 17.78}$	$38.90_{\pm 13.01}$	$38.05_{\pm 8.95}$
1000_1000_Sol3	$24.65_{\pm 7.86}$	$26.60_{\pm 13.18}$	$34.27_{\pm 25.62}$	$36.33_{\pm 20.79}$	$37.57_{\pm 17.01}$	$35.15_{\pm 12.90}$
1000_1000_Sol4	$15.55_{\pm 14.95}$	$21.61_{\pm 12.58}$	$36.31_{\pm 22.19}$	$38.08_{\pm 21.69}$	$38.40_{\pm 20.37}$	$36.20_{\pm 14.49}$
1000_1000_Sol5	$25.01_{\pm 7.33}$	$22.20_{\pm 10.37}$	$35.45_{\pm 17.87}$	$37.32_{\pm 24.73}$	$36.90_{\pm 23.95}$	$37.51_{\pm 17.08}$
500_1000_Sol1	$16.38_{\pm 5.00}$	$19.71_{\pm 9.86}$	$33.23_{\pm 15.62}$	$32.40_{\pm 19.44}$	$32.96_{\pm 12.81}$	$37.07_{\pm 10.38}$
500_1000_Sol2	$14.86_{\pm 8.55}$	$23.97_{\pm 10.93}$	$34.63_{\pm 13.83}$	$37.04_{\pm 14.95}$	$39.18_{\pm 9.57}$	$38.14_{\pm 13.65}$
500_1000_Sol3	$20.72_{\pm 7.86}$	$31.84_{\pm 7.77}$	$32.99_{\pm 20.34}$	$37.77_{\pm 13.92}$	$38.92_{\pm 14.78}$	$32.42_{\pm 6.92}$
500_1000_Sol4	$12.33_{\pm 12.81}$	$18.32_{\pm 7.97}$	$33.21_{\pm 16.63}$	$34.12_{\pm 15.15}$	$37.66_{\pm 14.80}$	$37.30_{\pm 9.38}$
500_1000_Sol5	$20.10_{\pm 5.21}$	$18.41_{\pm 6.29}$	$34.88_{\pm 14.19}$	$33.52_{\pm 17.73}$	$35.86_{\pm 20.70}$	$32.73_{\pm 16.05}$

Table 12 Energy used per forwarding (miliWatts) for the 300 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	$7.90_{\pm 9.50}$	$24.37_{\pm 4.03}$	$20.07_{\pm 10.78}$	$30.35_{\pm 9.42}$	$26.93_{\pm 7.23}$	$39.61_{\pm 7.08}$
500_500_Sol2	$7.59_{\pm 10.59}$	$27.07_{\pm 3.72}$	$19.11_{\pm 6.65}$	$28.09_{\pm 8.10}$	$26.63_{\pm 9.01}$	$36.64_{\pm 8.74}$
500_500_So3	$7.19_{\pm 17.32}$	$15.45_{\pm 4.65}$	$28.91_{\pm 8.24}$	$28.97_{\pm 7.46}$	$29.19_{\pm 9.64}$	$29.40_{\pm 7.53}$
500_500_Sol4	$7.62_{\pm 16.84}$	$23.22_{\pm 4.68}$	$18.57_{\pm 7.28}$	$33.47_{\pm 7.50}$	$39.29_{\pm 9.14}$	$27.63_{\pm 11.13}$
500_500_Sol5	$15.73_{\pm 6.19}$	$17.61_{\pm 8.18}$	$30.23_{\pm 10.85}$	$29.80_{\pm 11.12}$	$28.96_{\pm 6.63}$	$37.92_{\pm 5.47}$
1000_1000_Sol1	$14.10_{\pm 7.29}$	$19.97_{\pm 11.16}$	$37.07_{\pm 15.68}$	$39.87_{\pm 25.31}$	$37.04_{\pm 24.90}$	$39.65_{\pm 18.45}$
1000_1000_Sol2	$15.98_{\pm 6.60}$	$33.92_{\pm 8.93}$	$32.11_{\pm 19.29}$	$38.13_{\pm 23.36}$	$35.03_{\pm 23.44}$	$37.06_{\pm 21.97}$
1000_1000_Sol3	$16.61_{\pm 7.70}$	$32.37_{\pm 9.93}$	$28.71_{\pm 16.94}$	$34.30_{\pm 25.52}$	$39.74_{\pm 21.14}$	$39.81_{\pm 22.55}$
1000_1000_Sol4	$15.47_{\pm 4.46}$	$28.04_{\pm 11.01}$	$30.71_{\pm 21.27}$	$31.71_{\pm 24.77}$	$36.57_{\pm 24.35}$	$38.59_{\pm 20.31}$
1000_1000_Sol5	$17.51_{\pm 9.11}$	$30.38_{\pm 22.27}$	$39.77_{\pm 25.76}$	$38.94_{\pm 21.61}$	$38.79_{\pm 12.69}$	$37.26_{\pm 7.43}$
500_1000_Sol1	$14.51_{\pm 4.83}$	$32.12_{\pm 6.45}$	$34.45_{\pm 15.40}$	$38.84_{\pm 15.34}$	$34.92_{\pm 18.50}$	$39.15_{\pm 17.30}$
500_1000_Sol2	$18.20_{\pm 4.26}$	$21.64_{\pm 6.83}$	$36.88_{\pm 15.35}$	$35.31_{\pm 14.36}$	$37.55_{\pm 18.44}$	$37.32_{\pm 16.69}$
500_1000_Sol3	$15.53_{\pm 5.00}$	$16.93_{\pm 12.76}$	$29.48_{\pm 13.96}$	$34.60_{\pm 14.27}$	$38.43_{\pm 14.20}$	$39.27_{\pm 13.31}$
500_1000_Sol4	$14.11_{\pm 5.11}$	$20.49_{\pm 7.95}$	$32.14_{\pm 13.79}$	$28.92_{\pm 12.22}$	$31.07_{\pm 15.02}$	$30.50_{\pm 19.04}$
500_1000_Sol5	$31.30_{\pm 6.69}$	$23.45_{\pm 14.78}$	$38.88_{\pm 19.18}$	$34.80_{\pm 12.40}$	$36.33_{\pm 7.83}$	$38.73_{\pm 9.27}$

Table 13 Percentage of the energy saved per forwarding (miliWatts) for the 100 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	51.24	29.27	21.03	5.18	12.78	22.24
500_500_Sol2	70.56	37.26	15.54	25.59	14.95	19.61
500_500_Sol3	48.68	26.11	24.85	22.44	10.69	26.78
500_500_Sol4	72.22	40.80	13.62	21.72	21.81	10.06
500_500_Sol5	54.76	18.89	12.19	5.83	7.54	21.23
1000_1000_Sol1	45.55	18.00	8.64	9.29	5.61	58.12
1000_1000_Sol2	36.36	25.42	7.51	8.90	14.56	25.95
1000_1000_Sol3	49.93	21.53	14.40	7.39	14.30	61.35
1000_1000_Sol4	32.62	14.78	6.40	6.21	7.60	8.71
1000_1000_Sol5	50.51	16.43	12.48	4.78	6.46	6.74
500_1000_Sol1	54.01	19.09	6.44	6.72	11.17	16.15
500_1000_Sol2	60.46	32.94	14.95	8.43	11.90	19.24
500_1000_Sol3	41.59	25.15	18.06	13.30	8.64	24.45
500_1000_Sol4	30.45	26.75	13.91	3.00	10.49	12.69
500_1000_Sol5	44.50	27.59	10.78	10.90	9.32	12.45

Table 14 Percentage of the energy saved per forwarding (miliWatts) for the 200 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	63.95	38.76	35.19	36.30	13.81	20.89
500_500_Sol2	68.07	12.95	22.01	1.87	4.08	21.64
500_500_Sol3	56.06	47.91	10.88	16.49	7.84	13.87
500_500_Sol4	65.27	56.90	23.49	33.91	34.23	18.28
500_500_Sol5	75.03	54.64	17.99	27.58	24.58	24.13
1000_1000_Sol1	70.17	25.35	6.92	11.20	3.25	6.65
1000_1000_Sol2	60.65	30.77	2.82	2.28	2.74	4.88
1000_1000_Sol3	38.38	33.50	14.33	9.18	6.07	12.14
1000_1000_Sol4	61.12	45.97	9.23	4.81	4.00	9.49
1000_1000_Sol5	37.49	44.49	11.37	6.71	7.76	6.22
500_1000_Sol1	59.05	50.73	16.93	18.99	17.59	7.33
500_1000_Sol2	62.85	40.08	13.42	7.39	2.04	4.66
500_1000_Sol3	48.19	20.39	17.53	5.57	2.70	18.96
500_1000_Sol4	69.18	54.21	16.98	14.70	5.85	6.75
500_1000_Sol5	49.75	53.98	12.79	16.20	10.35	18.18

Table 15 Percentage of the energy saved per forwarding (miliWatts) for the 300 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	80.26	39.09	49.83	24.13	32.67	0.97
500_500_Sol2	81.02	32.32	52.22	29.78	33.43	8.40
500_500_Sol3	82.03	61.36	27.73	27.57	27.04	26.49
500_500_Sol4	80.96	41.95	53.57	16.32	1.78	30.92
500_500_Sol5	60.67	55.98	24.43	25.51	27.59	5.20
1000_1000_Sol1	64.76	50.07	7.33	0.33	7.39	0.88
1000_1000_Sol2	60.05	15.20	19.72	4.68	12.43	7.34
1000_1000_Sol3	58.49	19.08	28.23	14.26	0.64	0.48
1000_1000_Sol4	61.33	29.89	23.22	20.72	8.58	3.52
1000_1000_Sol5	56.22	24.05	0.57	2.64	3.03	6.86
500_1000_Sol1	63.73	19.71	13.88	2.91	12.71	2.12
500_1000_Sol2	54.49	45.91	7.80	11.71	6.12	6.69
500_1000_Sol3	61.19	57.66	26.30	13.50	3.93	1.84
500_1000_Sol4	64.72	48.78	19.64	27.71	22.33	23.74
500_1000_So5	21.74	41.39	2.81	13.01	9.18	3.16

Table 16 Broadcast time (nanoseconds) for the 100 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	6,6154E008 ±6,5157E008	1,3099E009 ±6,5754E008	1,4179E009 ±5,7336E008	1,1462E009 ±4,2729E008	8,9515E008 ±2,7830E008	9,1945E008 ±2,7952E008
500_500_Sol2	7,7402E008 ±7,0499E008	1,3798E009 ±7,4075E008	1,4874E009 ±4,7012E008	1,2039E009 ±3,5424E008	1,0575E009 ±2,8721E008	1,0559E009 ±2,8283E008
500_500_Sol3	7,2361E008 ±6,2011E008	1,6432E009 ±8,0836E008	1,7656E009 ±5,2148E008	1,6909E009 ±3,6623E008	1,6640E009 ±3,1593E008	1,7074E009 ±3,0946E008
500_500_Sol4	6,2314E008 ±6,7361E008	1,6055E009 ±9,1984E008	1,8259E009 ±7,4059E008	1,5942E009 ±5,2176E008	1,4997E009 ±3,9238E008	1,3610E009 ±3,8485E008
500_500_Sol5	5,4429E008 ±4,5984E008	1,0291E009 ±5,2180E008	1,2588E009 ±4,7901E008	1,1579E009 ±2,9422E008	1,0984E009 ±2,7771E008	1,0194E009 ±2,3924E008
1000_1000_Sol1	1,3333E009 ±1,2498E009	3,1248E009 ±1,6045E009	2,8321E009 ±5,9579E008	2,2839E009 ±4,5698E008	1,9492E009 ±3,9714E008	8,2658E008 ±1,5656E008
1000_1000_Sol2	1,4491E009 ±1,2790E009	3,4549E009 ±1,9259E009	3,0808E009 ±6,2124E008	2,5054E009 ±4,8589E008	2,2131E009 ±3,4848E008	2,2314E009 ±3,5992E008
1000_1000_Sol3	1,3911E009 ±1,2918E009	4,0641E009 ±2,2789E009	4,0530E009 ±9,9518E008	3,6081E009 ±5,3206E008	3,8044E009 ±4,9928E008	2,1550E009 ±3,8050E008
1000_1000_Sol4	1,1990E009 ±1,1615E009	3,5028E009 ±2,2853E009	4,3003E009 ±1,0163E009	3,2706E009 ±6,3141E008	2,8835E009 ±5,7864E008	2,7360E009 ±4,1883E008
1000_1000_Sol5	8,6453E008 ±9,5478E008	2,8506E009 ±1,5637E009	2,8817E009 ±6,8224E008	2,5613E009 ±4,3662E008	2,2964E009 ±3,8654E008	2,2621E009 ±3,2175E008
500_1000_Sol1	8,0690E008 ±7,9467E008	1,9209E009 ±1,1444E009	2,2875E009 ±6,7315E008	1,8635E009 ±4,9219E008	1,5096E009 ±3,4822E008	1,5124E009 ±3,2009E008
500_1000_Sol2	9,6769E008 ±1,1539E009	2,3584E009 ±1,2235E009	2,6191E009 ±8,9059E008	2,0392E009 ±5,6597E008	1,7045E009 ±4,7228E008	1,7138E009 ±3,5770E008
500_1000_Sol3	1,0418E009 ±8,7463E008	2,4724E009 ±1,2656E009	3,2953E009 ±8,6244E008	3,0209E009 ±6,4367E008	2,9158E009 ±5,5982E008	2,9255E009 ±5,2940E008
500_1000_Sol4	8,7580E008 ±9,0799E008	2,1287E009 ±1,3190E009	3,1361E009 ±9,9381E008	2,6859E009 ±8,3037E008	2,4299E009 ±6,0552E008	2,1783E009 ±4,7697E008
500_1000_Sol5	8,1295E008 ±7,0898E008	1,6994E009 ±8,9772E008	2,2036E009 ±7,5509E008	1,9222E009 ±4,3763E008	1,8257E009 ±3,4859E008	1,7702E009 ±3,7212E008

32(5), 795–805 (2010)
19. Gendreau, M., Potvin, J.Y. (eds.): Handbook of Metaheuristics, *International Series in Operations Research & Management Science*, vol. 146. Springer (2010)
20. Gomez, J., Campbell, A.T.: Variable-range transmission power control in wireless ad hoc networks. IEEE Trans-

Table 17 Broadcast time (nanoseconds) for the 200 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	3,8473E008 ±3,5955E008	7,0653E008 ±5,3501E008	1,2367E009 ±5,7901E008	1,2160E009 ±5,5660E008	1,2838E009 ±4,8499E008	1,3175E009 ±3,8031E008
500_500_Sol2	3,3366E008 ±3,7953E008	6,7061E008 ±4,7525E008	9,8390E008 ±6,0591E008	9,8617E008 ±4,4023E008	8,7748E008 ±3,4996E008	7,8587E008 ±3,1416E008
500_500_Sol3	2,6836E008 ±2,7974E008	4,8495E008 ±3,8184E008	7,7340E008 ±3,7093E008	9,0347E008 ±4,0174E008	9,0723E008 ±3,6177E008	8,8091E008 ±3,5860E008
500_500_Sol4	4,4468E008 ±4,6683E008	7,7261E008 ±5,0474E008	1,3790E009 ±6,8913E008	1,5884E009 ±6,8037E008	1,7597E009 ±5,5813E008	1,8604E009 ±5,3811E008
500_500_Sol5	2,0153E008 ±2,5981E008	3,9766E008 ±2,7527E008	5,4718E008 ±4,1590E008	6,5278E008 ±3,5166E008	6,5218E008 ±2,6928E008	6,7129E008 ±3,0243E008
1000_1000_Sol1	5,5419E008 ±5,2270E008	1,1016E009 ±7,9488E008	2,6040E009 ±1,3856E009	3,0484E009 ±9,9830E008	3,1454E009 ±8,2776E008	2,8905E009 ±5,8808E008
1000_1000_Sol2	4,5357E008 ±5,6815E008	1,2241E009 ±1,0471E009	2,3816E009 ±1,1164E009	2,1594E009 ±6,7732E008	1,7313E009 ±4,9498E008	1,4629E009 ±3,1132E008
1000_1000_Sol3	3,3547E008 ±3,3832E008	9,3129E008 ±6,7096E008	1,8607E009 ±8,8969E008	2,1473E009 ±6,8109E008	2,0904E009 ±5,6618E008	1,8898E009 ±4,0811E008
1000_1000_Sol4	8,1496E008 ±6,5592E008	1,4076E009 ±1,1208E009	2,8766E009 ±1,8131E009	4,1745E009 ±1,5639E009	4,0799E009 ±1,3725E009	4,0103E009 ±9,4025E008
1000_1000_Sol5	3,0186E008 ±2,9279E008	5,2312E008 ±3,8334E008	1,0488E009 ±7,5293E008	1,5907E009 ±6,8288E008	1,6533E009 ±6,4644E008	1,5618E009 ±4,6857E008
500_1000_Sol1	3,9663E008 ±3,2049E008	8,4166E008 ±6,1614E008	1,7439E009 ±8,1038E008	2,0709E009 ±8,6187E008	2,0848E009 ±6,5696E008	2,2743E009 ±6,9881E008
500_1000_Sol2	4,9451E008 ±4,4850E008	8,2345E008 ±6,5631E008	1,4756E009 ±7,0885E008	1,5406E009 ±6,0786E008	1,3806E009 ±4,8222E008	1,2884E009 ±4,5390E008
500_1000_Sol3	3,3486E008 ±3,5190E008	5,3345E008 ±3,8792E008	1,1007E009 ±6,6910E008	1,5087E009 ±5,7664E008	1,5301E009 ±6,0122E008	1,4779E009 ±4,2202E008
500_1000_Sol4	5,4797E008 ±4,8582E008	1,0685E009 ±7,3099E008	1,8265E009 ±1,2154E009	2,5653E009 ±1,0178E009	2,7664E009 ±1,1283E009	3,0535E009 ±7,5592E008
500_1000_Sol5	2,9512E008 ±2,6466E008	5,1555E008 ±4,3231E008	6,8394E008 ±4,7181E008	9,7871E008 ±5,3208E008	1,1203E009 ±5,5042E008	1,1674E009 ±4,2527E008

Table 18 Broadcast time (nanoseconds) for the 300 devices optimized configurations

	50	100	200	300	400	500
500_500_Sol1	2,4370E008 ±3,2218E008	4,8394E008 ±4,1833E008	8,0567E008 ±5,6656E008	8,8758E008 ±5,1117E008	1,0372E009 ±5,0100E008	1,0101E009 ±4,6429E008
500_500_Sol2	4,5934E008 ±5,4414E008	8,9021E008 ±6,2886E008	1,3705E009 ±9,1303E008	1,9096E009 ±8,6020E008	1,9622E009 ±7,7064E008	2,0932E009 ±9,1611E008
500_500_Sol3	4,0056E008 ±4,4921E008	8,3080E008 ±4,9611E008	1,1859E009 ±7,4509E008	1,5581E009 ±6,4637E008	1,5576E009 ±6,4913E008	1,6584E009 ±6,6446E008
500_500_Sol4	3,9501E008 ±4,4984E008	6,6917E008 ±4,6996E008	1,0750E009 ±5,5631E008	1,4236E009 ±6,8233E008	1,4213E009 ±7,1921E008	1,5923E009 ±6,3581E008
500_500_Sol5	4,6171E008 ±5,3161E008	1,0135E009 ±7,4636E008	1,5221E009 ±8,2773E008	1,5607E009 ±6,0374E008	1,4186E009 ±5,9295E008	1,2863E009 ±4,0319E008
1000_1000_Sol1	4,2980E008 ±4,5043E008	6,7359E008 ±6,3772E008	1,4251E009 ±9,1059E008	1,9249E009 ±1,0629E009	2,2417E009 ±9,9529E008	2,3394E009 ±7,3389E008
1000_1000_Sol2	7,9543E008 ±6,9023E008	1,3505E009 ±9,6943E008	2,9138E009 ±1,8516E009	4,3003E009 ±2,3198E009	4,9831E009 ±1,6411E009	4,6277E009 ±1,3018E009
1000_1000_Sol3	6,7438E008 ±6,8036E008	1,0828E009 ±9,5325E008	2,5991E009 ±1,6092E009	3,5805E009 ±1,8187E009	4,1442E009 ±1,3096E009	3,8973E009 ±1,1496E009
1000_1000_Sol4	5,3604E008 ±4,6847E008	9,8223E008 ±6,3039E008	1,7442E009 ±1,3274E009	2,8747E009 ±1,6867E009	3,5981E009 ±1,7472E009	3,8231E009 ±1,2681E009
1000_1000_Sol5	7,6854E008 ±9,0763E008	1,7120E009 ±1,4397E009	3,4070E009 ±1,6814E009	3,4964E009 ±1,1201E009	2,8966E009 ±6,2255E008	2,5658E009 ±4,7560E008
500_1000_Sol1	3,9571E008 ±4,3867E008	5,2176E008 ±4,6797E008	8,5055E008 ±6,3590E008	1,4086E009 ±8,1541E008	1,5623E009 ±8,1779E008	1,5048E009 ±6,5074E008
500_1000_Sol2	7,9610E008 ±6,6290E008	1,1812E009 ±8,9912E008	1,7858E009 ±1,1731E009	2,7796E009 ±1,5296E009	3,2281E009 ±1,3317E009	3,5302E009 ±1,1674E009
500_1000_Sol3	5,8195E008 ±5,8967E008	8,9663E008 ±7,9297E008	1,7125E009 ±9,4261E008	2,3231E009 ±1,2440E009	2,5175E009 ±1,0664E009	2,7898E009 ±9,5770E008
500_1000_Sol4	5,7973E008 ±4,8739E008	8,7965E008 ±5,9160E008	1,1981E009 ±8,0013E008	1,9644E009 ±1,0828E009	2,2335E009 ±9,3333E008	2,5888E009 ±1,1191E009
500_1000_Sol5	6,7928E008 ±6,0651E008	1,0986E009 ±9,0160E008	2,3604E009 ±1,1008E009	2,5768E009 ±9,3751E008	2,3050E009 ±6,4480E008	2,0901E009 ±6,2535E008

actions on Mobile Computing **6**(1), 87–99 (2007)

21. Groenevelt, R.B., Altman, E., Nain, P.: Relaying in mobile ad hoc networks: The brownian motion mobility

model. Journal of Wireless Networks pp. 561–571 (2006)
22. Hogie, L., Guinand, F., Bouvry, P.: A heuristic for efficient broadcasting in the metropolitan ad hoc network. In: 8th Int. Conf. on Knowledge-Based Intelligent Information and Engineering Systems, pp. 727–733 (2004)
23. Huang, C., Chuang, Y., Hu, K.: Using particle swarm optimization for QoS in ad-hoc multicast. Engineering Applications of Artificial Intelligence **22**(8), 1188–1193 (2009)
24. Khan, I., Javaid, A., Qian, H.L.: Distance-based dynamically adjusted probabilistic forwarding for wireless mobile ad hoc networks. In: Wireless and Optical Communications Networks, 2008. WOCN '08. 5th IFIP International Conference on, pp. 1–6 (2008)
25. Lacage, M., Henderson, T.R.: Yet another network simulator. In: WNS2 '06: Proceeding from the 2006 workshop on ns-2: the IP network simulator, p. 12 (2006)
26. León, C., Miranda, G., Segura, C.: Optimizing the configuration of a broadcast protocol through parallel cooperation of multi-objective evolutionary algorithms. In: Proc. of the 2nd Int. Conf. on Advanced Engineering Computing and Applications in Sciences, pp. 135–140 (2008)
27. León, C., Miranda, G., Segura, C.: Optimizing the broadcast in MANETs using a team of evolutionary algorithms. In: 6th International Conference on Large-Scale Scientific Computing (LSSC07), *Lecture Notes in Computer Science*, vol. 4818, pp. 569–576. Springer-Verlag Heidelberg (2009)
28. Li, X., Nguyen, T.D., Martin, R.P.: Using adaptive range control to optimize 1-hop broadcast coverage in dense wireless networks. In: SenSys, pp. 314–315 (2003)
29. Liang, W., Brent, R., Xu, Y., Wang, Q.: Minimum-energy all-to-all multicasting in wireless ad hoc networks. Trans. Wireless. Comm. **8**(11), 5490–5499 (2009)
30. Ni, S.Y., Tseng, Y.C., Chen, Y.S., Sheu, J.P.: The broadcast storm problem in a mobile ad hoc network. In: 5th annual ACM/IEEE international conference on Mobile computing and networking, pp. 151–162 (1999)
31. Reumerman, H., Runi, M.: Distributed power control for reliable broadcast in inter-vehicle communication systems. In: 2nd International Workshop on Intelligent transportation (2005)
32. Ruiz, P., Bouvry, P.: Distributed energy self-adaptation in ad hoc networks. In: In proc. of IEEE Int. workshop on Management of Emerging Networks and Services (MENS), in conjunction with IEEE Globecom, pp. 539–543. Miami, USA (2010)
33. Ruiz, P., Bouvry, P.: Enhanced distance based broadcasting protocol with reduced energy consumption. In: Workshop on Optimization Issues in Energy Efficient Distributed Systems (OPTIM), part of the 2010 International Conference on High Performance Computing and Simulation (HPCS), pp. 249–258 (2010)
34. Ruiz, P., Dorronsoro, B., Bouvry, P.: Optimization and performance analysis of the AEDB broadcasting algorithm. In: International Workshop on Wireless Mesh and Ad Hoc Networks, in conjunction with International Conference on Computer Communication Networks, ICCN, pp. 1–6. Maui, Hawaii (2011)
35. Ruiz, P., Dorronsoro, B., Valentini, G., Pinel, F., Bouvry, P.: Optimisation of the enhanced distance based broadcasting protocol for manets. J. of Supercomputing. Special Issue on Green networks (to appear). DOI 10.1007/s11227-011-0564-x
36. Sapienza, T.J.: Optimizing quality of service of wireless mobile ad-hoc networks using evolutionary computation. In: Proceedings of the 4th annual workshop on Cyber security and information intelligence research (CSIIRW), pp. 1–5. ACM (2008)
37. Toutouh, J., Nesmachnow, S., Alba, E.: Fast energy-aware OLSR routing in VANETs by means of a parallel evolutionary algorithm. Cluster Computing **to appear** (2012)
38. Wu, S.L., Tseng, Y.C., Lin, C.Y., Sheu, J.P.: A multi-channel mac protocol with power control for multi-hop mobile ad hoc networks. Comput. J. **45**(1), 101–110 (2002)
39. Yen, Y.S., Chan, Y.K., Chao, H.C., Park, J.H.: A genetic algorithm for energy-efficient based multicast routing on MANETs. Computer Communications **31**, 2632–2641 (2008)
40. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. IEEE Transactions on Evolutionary Computation **3**(4), 257–271 (1999)