

# Distributed Energy Self-Adaptation in ad hoc Networks

Patricia Ruiz  
University of Luxembourg  
Luxembourg  
Email: patricia.ruiz@uni.lu

Pascal Bouvry  
University of Luxembourg  
Luxembourg  
Email: pascal.bouvry@uni.lu

**Abstract**—Nowadays energy management is a key feature in many different fields, specially in mobile ad hoc networks where devices heavily rely on the battery life, thus, the network survival is absolutely related to the energy consumption of nodes. In this work, we present a broadcasting algorithm AEDB that not only tries to reduce the network but also the device resources. AEDB is an extension of EDB which is a distance based and also energy aware broadcasting algorithm. The new proposed scheme, AEDB, regulates the transmission power of the device in order to decrease the energy consumption with no detriment of the performance of the algorithm. This is done by managing the transmission power of the device in terms of the number of one hop neighbors for disseminating a message. Results show that by adaptively managing the transmission power, the energy the device uses is highly reduced obtaining at the same time, good performance in terms of both the coverage achieved and the broadcast time.

## I. INTRODUCTION

Mobile ad hoc networks, also called MAETS, are spontaneously created between neighboring devices without the need of any existing infrastructure. Due to the exceptional amount of smart phones, PDAs, tablets, etc. that exist nowadays, the idea of creating a network at any place, and at any moment is becoming a hot topic where both industry and academia are involved. But it is still necessary to do a lot of work before MANETs become a reality because of the many challenging aspects surrounding them, e.g. the appearance and disappearance of devices, battery life, mobility of devices, limited transmission range, etc.

The technology used in these MANETs is wireless, as devices move from one point to another. In this environment, the most appropriate schemes to use are dissemination algorithms because of the intrinsic broadcast nature of the wireless medium. Moreover, most of the applications consider broadcasting as a fundamental low level operation, and even some other communication protocols assume the existence of a broadcasting service. The main problem associated to these algorithms is the broadcast storm problem [1], where all devices resend the received message what may lead to the congestion of the network. Many researchers try to minimize the network resources and the broadcast time, maximizing at the same time the number of devices in the network receiving the message [2].

The main drawback 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 [3], [4], [5], [6], [7].

We are considering the enhanced distance based broadcasting protocol (EDB) presented in [8], that is based on the original distance based broadcasting algorithm (DB) [1]. EDB reduces the network resources by using the distance between the source and the destination nodes as decision parameter for rebroadcasting the packet, but it also reduces the energy consumption for sending the broadcast message.

In this work, we are presenting AEDB that is an extension of EDB but with energy management capabilities, i.e., AEDB adapts its transmission power depending on the number of neighbors located in the furthest area covered by its transmission range.

The rest of the paper is organized as follows. Next section provides a small state of the art in energy aware broadcasting algorithms for MANETs. In Section III, the description of the original EDB is given. Section IV presents the broadcasting algorithm AEDB. The simulation parameters and the results obtained are explained in Sections V and VI, respectively. Finally, we conclude the paper in Section VII.

## II. RELATED WORK

As we previously mentioned, many researchers are focusing in reducing the energy consumption, especially in ad hoc networks where devices rely on batteries and the massive disconnection of devices would lead to the network disappearance. Below, we mention some of the most relevant works that already exist in the literature.

Cartigny et al. proposed in [9] an algorithm where the node is allowed to choose a transmission power level for performing a rebroadcast based on local information (about the two hop neighborhood of the node). In this work, a node constructs a restricted neighborhood graph and attempts to choose its transmission power level only to reach nodes within its restricted neighborhood graph. This approach relies on using the distance between different nodes. If we do not assume GPS, the distance is estimated using the received power.

In [10], the transmission power needed to reach the two hop neighborhood is calculated using the beacons. The source

node examines if it worths excluding the furthest node from the one hop neighborhood and reducing the transmission range to reach the new furthest neighbor. This is done if the latter can be used as relay node to successfully cover the two hop neighborhood.

It was shown in [4] that an approach using a variable transmission range can outperform another one using common range in terms of capacity and power saving.

An approach to estimate the local density using an analytical model is used in [3]. According to this value the transmission range is set.

Studies on energy efficient algorithms for finding the minimum-energy broadcast tree (MEBT) have been proposed [5], [6], and also in [7] a shared multicast tree built in a distributed fashion with minimum energy is presented, where the transmission power is either fixed or adjustable.

Also in [8] an energy aware broadcasting algorithm is used to estimate according to the reception signal strength the maximum transmission power needed to reach the furthest node during the broadcasting process. This estimate is calculated using the beacons that nodes periodically exchange.

### III. ENHANCED DISTANCE BASED BROADCASTING ALGORITHM

In this section we briefly explain the mechanism of the enhanced distance based broadcasting algorithm (EDB) proposed in [8] (for a more detailed explanation see [8]), that we took as starting point for this work.

The idea is that only devices that are far enough from the source node will rebroadcast the message, as devices closer would not contribute much to the performance of the coverage achieved by the dissemination process. This idea was first presented in [1].

For calculating the distance between the source and the destination nodes, the most common technique is either assuming a GPS service or considering the signal strength. In EDB no GPS service is assumed, thus, the signal strength of the received packet is used to estimate how far two nodes are. Once the distance is calculated, it is necessary to predefine a value from which on nodes are considered to be far enough from the source node, thus they retransmit. This threshold is not in terms of distance (m) but power (dBm), and it is called *borders\_Threshold*.

Apart from that, EDB also tries to reduce the energy consumption of the devices. It considers that for sending a message it is only necessary to reach the neighbors in range, that is, if neighbors are close to the source node it is possible to reduce the default transmission power ( $r$ ) in order to reach only the furthest node (see Figure 1).

In EDB a crosslayer design was implemented to take the received signal strength from the physical layer, and bubble it to the upper layers in order to decide in terms of its value whether to rebroadcast or not.

Every device sends a *hello* message (or beacon) to alert devices within a close area about their presence. We assume

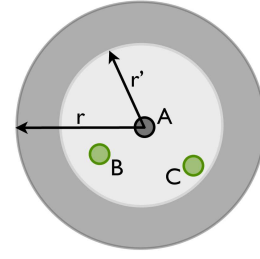


Fig. 1. Reducing the transmission power of the node.

that all nodes send the *hello message* with the same transmission power. Thus, a node receiving a beacon will be able to estimate the loss that packet suffered during the transmission, using the reception power detected at the physical layer and the default transmission power.

Every node keeps and updates the reception power of each of its neighbors in a list. When a device wants to send a broadcast message, it will be able to estimate the loss the packet will suffer (as we assume a packet traversing in a direction will experiment the same loss as another one traversing in the opposite direction).

If a node can estimate the loss the packet is going to suffer, it will be able to reduce its transmission power and use only the necessary one to get the furthest one hop neighbor. Thus, reducing the transmission power for sending broadcast messages directly decreases the energy consumption of the device, without degrading the performance of the broadcasting process as loosing the connection with any neighbor it is not considered. The pseudocode of EDB is shown in Algorithm 1.

---

#### Algorithm 1 Pseudocode of enhanced DB.

---

**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:**  $p_{min}$ : the minimum signal strength received for  $m$  from any  $s$ .  
**Data:**  $borders\_Threshold$ : the signal strength threshold.

```

1: if  $m$  is received for the first time then
2:   calculate  $p$ ;
3:   update  $p_{min}$ ;
4:   if  $p_{min} > borders\_Threshold$  then
5:      $r \rightarrow$  drop message  $m$ ;
6:   else
7:     waiting = true;
8:     wait time rand/randDist/fixedDist;
9:     goto 17;
10:  end if
11: else if waiting then
12:   calculate  $p$ ;
13:   if  $p > p_{min}$  then
14:     update  $p_{min}$ ;
15:   end if
16: end if
17: if  $p_{min} > borders\_Threshold$  then
18:    $r \rightarrow$  drop message  $m$ ;
19: else
20:   estimate power to reach furthest neighbor
21:   transmit  $m$ ;
22: end if
23: waiting = false;

```

---

#### IV. ADAPTIVE ENHANCED DISTANCE BASED BROADCASTING ALGORITHM

As EDB does not consider the possibility of missing any neighbor when reducing the transmission power, it was checked in [8] that the energy saved is reducing as the density of the network increases. That is logical, as the higher the density the more probable the existence of neighbors close to the transmission limit range, thus, almost no energy reduction is performed.

In that work, different configurations of the delay were studied. As result, choosing a random value between  $[0, 1]$  (s) showed the best behavior, therefore, this is the configuration we use for this implementation.

The idea arisen in this work is that, usually when the network is very dense the connectivity is 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 as the dissemination is easier. 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.

The node will dynamically manage the transmission power used in terms of the node density. The number of nodes located in the forwarding area is considered in order to decide how dense the network is. If this amount is higher than a predefined threshold called *neighbors\_Threshold*, then the network is considered dense, and the node decreases its transmission power for disseminating the message. The new furthest neighbor is one of the nodes located in the forwarding area but the one closest to the source node. In Figure 2, the idea is represented graphically. The number of neighbors in the forwarding area (dark grey zone) is higher than *neighbors\_Threshold*, thus, the transmission power is adapted to reach the new furthest neighbor, in this case node 'B' (in red). The pseudocode of the protocol is shown in Algorithm 2.

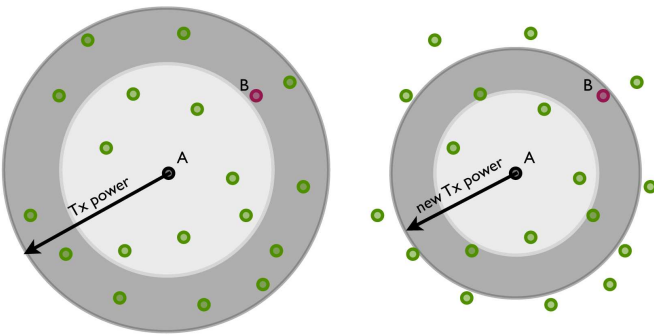


Fig. 2. Selecting the new transmission power of the node.

As we mentioned before, the threshold to decide whether the network is dense enough to discard some neighbors or not is called *neighbors\_Threshold*, and its value is studied in this work.

#### Algorithm 2 Pseudocode of 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:**  $p_{min}$ : the minimum signal strength received for  $m$  from any  $s$ .  
**Data:**  $borders\_Threshold$ : the signal strength threshold.  
**Data:**  $neighbors\_Threshold$ : # neighbors in the forwarding area.

```

1: if  $m$  is received for the first time then
2:   calculate  $p$ ;
3:   update  $p_{min}$ ;
4:   if  $p_{min} > borders\_Threshold$  then
5:      $r \rightarrow$  drop message  $m$ ;
6:   else
7:     waiting = true;
8:     wait time rand;
9:     goto 17;
10:  end if
11: else if waiting then
12:   calculate  $p$ ;
13:   if  $p > p_{min}$  then
14:     update  $p_{min}$ ;
15:   end if
16: end if
17: if  $p_{min} > borders\_Threshold$  then
18:    $r \rightarrow$  drop message  $m$ ;
19: else
20:   if # border_neighbors > neighbors_Threshold then
21:     estimate power to reach closest neighbor to borders_Threshold
22:   else
23:     estimate power to reach furthest neighbor
24:   end if
25:   transmit  $m$ ;
26: end if
27: waiting = false;

```

---

#### V. SIMULATION PARAMETERS

For simulating the proposed algorithm we are using the discrete event simulator ns3 [11], [12], written in C++ and with an optional Python scripting API. In order to validate the performance of the algorithm proposed, we are studying the behavior of the original EDB and comparing it to AEDB in four different aspects:

- 1) The coverage achieved: that is, the total number of devices that receive the broadcasting message.
- 2) The energy used for disseminating the message in terms of *dBm*.
- 3) The complexity of the message also called the network resources, measured in terms of the number of nodes that rebroadcast the message.
- 4) The broadcasting time, i.e., the period between the source node starts the dissemination and the last node receives it.

As we are dealing with mobile ad hoc networks, it is necessary to set a mobility model for the devices. In this case we are using the *random walk* also known as *brownian motion mobility model* [13], in which nodes move with a speed and direction randomly chosen during a fixed amount of time, in our case 20 seconds. If a node hits one of the limits of the simulation area, it rebounds on the boundary with a reflexive angle and speed.

The simulation environment used is a square area of 2000 m side ( $4\text{Km}^2$ ). The speed of the nodes can vary from 0 to 2m/s (between 0 and 7.2Km/h). Table I shows the parametrization

used for the protocol. These values were the same as the proposed ones in the original EDB (see [8] for a detailed explanation).

TABLE I  
PARAMETERIZATION USED

<i>Number of devices</i>	500-1200
<i>Speed</i>	[0, 2] m/s
<i>Size of the area</i>	2000 m × 2000 m
<i>Transmission power</i>	16.02 dBm
<i>end_Threshold</i>	-95 dBm
<i>borders_Threshold</i>	-90 dBm
<i>margin_Forwarding</i>	0.5 dBm
<i>neighbors_Threshold</i>	12
<i>Delay interval</i>	[0, 1] s
<i>Direction change</i>	every 20 s

In [8], the graph for the energy saved showed that it was increasing up to networks with densities varying between 500 and 600 devices. However, for a network with 700 devices it still showed a small reduction in the energy consumption (7.55%). Therefore, in this work we are studying networks whose number of devices range from 500 to 1200 for the same simulation area. Different values for the *neighbor\_Threshold* have been studied. We present here, only the most relevant ones: 8 and 12.

## VI. SIMULATION RESULTS

In order to obtain reliable results in our experiments, we are doing 100 independent simulation runs for each proposal of the algorithm. Table II presents the average of the values obtained after 100 runs for each of the network densities studied (from 500 to 1200 devices). These results are shown for each of the proposed settings of the *neighbor\_Threshold* (8 and 12). The parameters showed are: (1) the total energy used, (2) the coverage achieved, (3) the number of rebroadcasts performed, and finally (4) the broadcast time.

TABLE II  
RESULTS OF THE EXPERIMENTS.

	500	600	700	800	900	1000	1100	1200	
EDB	<i>Eg. used</i>	1711.25 ±680.27	2407.14 ±519.84	2705.25 ±480.16	2909.76 ±72.27	2950.64 ±420.90	3071.00 ±68.11	3132.35 ±75.74	3170.74 ±68.64
	<i>Coverage</i>	345.69 ±132.84	538.18 ±111.59	666.78 ±116.36	792.46 ±8.19	879.12 ±122.69	997.25 ±2.87	1097.88 ±2.02	1198.28 ±1.44
	<i>Forwarding</i>	114.02 ±45.75	159.67 ±34.71	178.61 ±31.89	191.35 ±4.88	193.37 ±27.73	200.73 ±4.52	204.27 ±5.07	206.45 ±4.57
	<i>Bc time</i>	7.89 ±2.66	7.81 ±2.02	6.47 ±1.62	5.55 ±1.11	5.04 ±1.19	4.58 ±0.88	4.12 ±0.75	3.78 ±0.70
AEDB	<i>Eg. used</i>	1098.10 ±740.44	1687.53 ±807.05	2117.30 ±743.52	2338.38 ±729.43	2704.97 ±606.89	2824.66 ±774.63	3196.27 ±599.47	3399.14 ±511.90
	<i>Coverage</i>	223.84 ±145.70	382.55 ±178.74	527.59 ±179.58	632.22 ±189.67	779.18 ±166.05	857.68 ±227.74	1016.76 ±184.67	1134.21 ±164.64
	<i>Forwarding</i>	75.84 ±51.60	120.02 ±57.71	155.75 ±54.90	179.15 ±55.62	214.51 ±48.09	230.88 ±63.39	268.45 ±50.24	291.00 ±43.80
	<i>Bc time</i>	6.84 ±3.44	8.04 ±3.04	8.83 ±2.26	9.05 ±1.99	9.27 ±1.56	8.96 ±1.82	9.06 ±1.66	9.06 ±1.41
8 neigh.	<i>Eg. used</i>	1590.74 ±805.38	2354.35 ±663.53	2626.89 ±614.41	2859.41 ±338.14	3015.63 ±74.54	3153.75 ±69.55	3266.72 ±73.67	3349.61 ±340.59
	<i>Coverage</i>	320.21 ±157.81	525.70 ±145.60	649.32 ±150.21	776.77 ±88.16	892.55 ±8.25	992.43 ±8.66	1094.71 ±5.37	1183.66 ±118.23
	<i>Forwarding</i>	106.17 ±54.26	157.45 ±44.61	176.71 ±41.58	195.33 ±23.31	211.20 ±5.68	227.25 ±5.78	244.16 ±5.75	258.51 ±26.62
	<i>Bc time</i>	7.61 ±3.16	7.81 ±2.25	6.88 ±1.95	6.62 ±1.33	6.67 ±1.24	6.59 ±1.16	6.51 ±1.14	6.42 ±1.23

The first set of results in Table II correspond to EDB. It can be seen that from 800 devices on, the network is very dense as almost all the nodes receive the message. The lowest coverage is found in the sparsest network with 69,13% of devices reached, while the densest achieved 99.86%. The percentage

of the number of forwarding nodes decreases as the density increases. The highest value for the number of rebroadcasts is 26.61%, obtained in the network composed of 600 devices, and the lowest 17.20% for 1200 nodes. The broadcasting process takes never more than 7.89 seconds, and it was checked that the denser the network, the lower the duration.

We first observe the coverage achieved by each possible solution, as the main idea is to reduce the transmission power, but maintaining the good of performance EDB. For the results obtained when the *neighbors\_Threshold* was set to 12, we observe that the percentage of the coverage achieved ranged from 64.04% to 99.51%, values that are similar to the ones obtained in the original EDB. Moreover, we did statistical tests comparing the coverage achieved by each proposal (EDB and AEDB with value 12) and EDB was better with statistical differences only for the networks with 1000, 1100 y 1200 devices, where the coverage obtained but AEDB is already satisfactory, around 99%. Because of the lack of space we only present one of the graphs obtained. This study has been done using the *boxplot* function from *Matlab*. In the displayed *boxplot*, 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 crosses 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 is statistical significant difference in the data with 95% confidence.

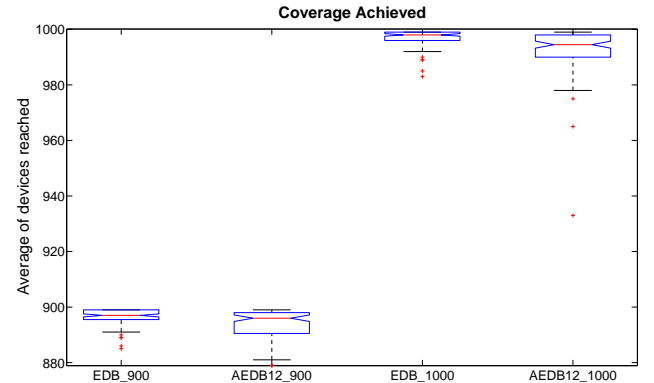


Fig. 3. Statistical study of the coverage for 900 and 1000.

In Figure 3, we can see the *boxplots* related to the coverage achieved by the original EDB for 900 and 1000 densities and AEDB (with the value 12 for the threshold) for the same densities. As explained before, AEDB and EDB do not have significant difference for 900, but EDB performs better for 1000 with statistical significant.

Regarding the time needed to broadcast the message, Table II shows that it is higher for the denser networks, but it must be noticed that never more than 2.64 seconds. The number of forwarding nodes is, as expected, increased. But

the maximum percentage obtained for this configuration is smaller than the maximum for the original 26.24%. For the 3 first densities this configuration uses less nodes for forwarding than the original, but for the densest network (1200 devices), the number of rebroadcast is increased at maximum in 4.33% (total percentage for EDB is 21.54% in the same environment).

The values obtained with *neighbors\_Threshold* equal to 8 gives lower values for the coverage, varying from 44.77% to 94.52%. The percentage of forwarding nodes is also smaller compared to the original EDB for the first four densities but it is a 5.83% higher for 1100 devices (up to 24.40%). The time needed for covering the network is also higher, exactly 5.28 seconds longer.

In order to make a statement about the benefits of reducing the transmission power when the density is high, we need to calculate the average of the transmission power each device uses in the broadcasting process. When reducing the transmission power, more rebroadcast are generally needed to cover the same area. The number of hops increases so does the broadcast time (those values are shown in Table II). We are not interested in the total energy used for the dissemination process in the whole network, but in reducing the energy each device spends in order to make longer its battery life. So, it is necessary to provide the average energy used per device. This is presented in Table III, and also the percentage of the savings obtained in every density compared to the original EDB.

TABLE III  
AVERAGE OF ENERGY USED PER FORWARDING.

	500	600	700	800	900	1000	1100	1200
<b>EDB</b>	15.01	15.08	15.15	15.21	15.26	15.30	15.33	15.36
<b>AEDB_8</b>	14.48	14.06	13.59	13.05	12.61	12.23	11.91	11.68
% Eg. saved AEDB_8	11.49	20.93	30.18	39.19	45.67	50.68	54.50	57.15
<b>AEDB_12</b>	14.98	14.95	14.87	14.64	14.28	13.88	13.38	12.96
% Eg. saved AEDB_12	0.69	2.95	6.24	12.30	20.20	27.89	36.17	42.46

The values presented in Table III are in logarithmic scale, so after converting them to the linear one, we can see that when setting the *neighbors\_Threshold* to 8, the energy used is at minimum reduced 11.49% up to 57.15%. In the case of establishing the threshold to 12 neighbors, the saving is lower but still very considerable, from 0.69% to 42.46%. We must highlight that EDB already saves energy. If we consider the default transmission power (not performing any reduction as EDB does) the percentage of energy saved goes as far as 63.19% and 50.57% for the AEDB\_8 and AEDB\_12 settings respectively.

On one hand, the energy saved when using 12 as *neighbors\_Threshold* is smaller but the coverage achieved is similar to the one obtained with EDB and the time is only increased in 2.64 seconds for covering a network of 1200 devices. On the other hand, setting the threshold to 8 gives less power consumption, decreasing 5.34% the coverage and increasing in 5.28 seconds the total time of the broadcast process.

## VII. CONCLUSIONS

In this work we present AEDB, an adaptive energy aware distance based broadcasting protocol that allows each device to locally manage the transmission power used for each broadcasting message sent.

We take as starting point EDB, that already reduces the transmission power in order to reach its furthest neighbor. The problem in EDB is that, as it does not consider loosing any neighbor in the process, when the network is very dense no reduction is performed or a very low one.

In order to save energy even in denser networks, in this paper we study the possibility of reducing the transmission power of each node even allowing the loss of some neighbors as the network connectivity might not be really affected when dealing with high densities. As the network density is directly related to the number of neighbors, to reduce the signal strength the node must have a minimum number of neighbors. Moreover, these neighbors must be in the forwarding area in order to be sure that the message will be forwarded.

The results show that any of the two values proposed for the threshold highly reduce the energy consumption. Depending on the necessities we can choose between maintaining a similar the coverage but reducing energy up to 42.46% (using *neighbors\_Threshold* equal to 12), or save more energy (up to 57.15%) but decreasing the coverage in a 5.34% in average and taking 5.28 seconds longer for disseminating.

As a future work, we plan to make a deeper study on the values of the thresholds in the protocol by using a multi-objective optimization that ensures, minimum energy, maximum coverage and minimum broadcast time. And also, to provide the protocol with different sets of parameters so that the node will be able to locally choose between different the settings depending on the circumstances.

## REFERENCES

- [1] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *ACM/IEEE Int. Conf. on Mobile computing and networking*, 1999, pp. 151–162.
- [2] E. Alba and B. Dorransoro, *Cellular Genetic Algorithms*, ser. Operations Research/Computer Science Interfaces. Springer-Verlag, 2008.
- [3] X. Li, T. D. Nguyen, and R. P. Martin, "Using adaptive range control to optimize 1-hop broadcast coverage in dense wireless networks," in *SenSys*, 2003, pp. 314–315.
- [4] J. Gomez and A. T. Campbell, "Variable-range transmission power control in wireless ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 1, pp. 87–99, 2007.
- [5] M. Cagalj, J.-P. Hubaux, and C. Enz, "Minimum-energy broadcast in all-wireless networks: Np-completeness and distribution issues," in *Int. Conf. on Mobile Computing and Networking*, 2002, pp. 172–182.
- [6] M. Cagalj, J.-P. Hubaux, and C. C. Enz, "Energy-efficient broadcasting in all-wireless networks," *Wirel. Net.*, vol. 11, no. 1, pp. 177–188, 2005.
- [7] W. Liang, R. Brent, Y. Xu, and Q. Wang, "Minimum-energy all-to-all multicasting in wireless ad hoc networks," *Trans. Wireless Comm.*, vol. 8, no. 11, pp. 5490–5499, 2009.
- [8] P. Ruiz and P. Bouvry, "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), 2010, pp. 249–258.
- [9] J. Cartigny, D. Simplot, and I. Stojmenovic, "Localized minimum-energy broadcasting in ad-hoc networks," pp. 2210 – 2217, 2003.
- [10] X. Chen, M. Faloutsos, and S. V. Krishnamurthy, "Power adaptive broadcasting with local information in ad hoc networks," in *Conference on Network Protocols*. IEEE Computer Society, 2003, p. 168.
- [11] M. Lacage and T. R. Henderson, "Yet another network simulator," in *WNS2, workshop on ns-2: the IP network simulator*, 2006, p. 12.
- [12] "ns3 project website, <http://www.nsnam.org/>."
- [13] R. B. Groenevelt, E. Altman, and P. Nain, "Relaying in mobile ad hoc networks: The brownian motion mobility model," *Journal of Wireless Networks*, pp. 561–571, 2006.