Routing Algorithms for Wireless Sensor Networks Using Ant Colony Optimization

Christian Domínguez-Medina and Nareli Cruz-Cortés

Center for Computing Research,
National Polytechnic Institute (CIC-IPN), Mexico
chrisdom27@hotmail.com, nareli@cic.ipn.mx

Abstract. Wireless Sensor Networks have become an active research topic in the last years. The routing problem is a very important part in this kind of networks that need to be considered in order to maximize the network life time. As the size of the network increases, routing becomes more complex due the amount of sensor nodes in the network. Sensor nodes in Wireless Sensor Networks are very constrained in memory capabilities, processing power and batteries. Ant Colony Optimization based routing algorithms have been proposed to solve the routing problem trying to deal with these constrains. We present a comparison of two Ant Colony-based routing algorithms, taking into account current amounts of energy consumption under different scenarios and reporting the usual metrics for routing in wireless sensor networks.

Keywords: Wireless Sensor Networks (WSN), network life time, routing algorithms, Ant Colony Optimization.

1 Introduction

Wireless Sensor Networks (WSN) consist of a large number of embedded sensors having the capability to communicate among them via wireless links deployed in an area that should be monitored. WSN are very effective in many fields such as intrusion detection, weather monitoring, security and tactical surveillance, distributed computing, detecting ambient conditions such as temperature, movement, sound, light, or the presence of certain objects, inventory control, and disaster management [8]. Nowadays, these sensor nodes are equipped with a small processor, constrained memory and constrained wireless communication capabilities. Furthermore, the sensor nodes are very limited in terms of their battery capabilities. For this reason, it is crucial to handle their energies properly [6]. A node in the WSN measures some phenomena from the environment, then it sends the measured data through others nodes to the base station. This base station could be connected to an application or the Internet. The strategy to build the path to the base station is known as a routing algorithm. The algorithms inspired by some biological phenomena have become popular in some Artificial Intelligence communities mainly because they have demonstrated to be competitive options to solve some hard problems from engineering and science. Specifically, a family of

G. Sidorov et al. (Eds.): MICAI 2010, Part II, LNAI 6438, pp. 337–348, 2010.

Ant Colony Optimization (ACO) algorithms [4] have been successfully applied to solve some routing problems in wired and wireless networks. We selected two ant-based routing algorithms and programmed and executed them in order to compare their performance under some metrics. The main contribution of this paper is the calculation of these metric values based on realistic amounts of energy consumption. Our goal is to give some light on the real earnings and feasibility of using this kind of algorithms. The remainder of the paper is organized as follows. In Section 2 the related work is presented. Section 3 describes the ACO algorithms. The Experiments are explained in Section 4. The results are described in Section 5, and finally in Section 6 some conclusions are drawn.

2 Related Work

WSN can be considered ad-hoc networks. However, protocols for Mobile Ad-hoc Networks (MANETs) can not be successfully applied to them because of WSN's special features [2]. There are a wide range of routing protocols that have been used to solve the problem of routing in WSN, for example a hierarchical clustering algorithm for sensor networks, called "Low Energy Adaptive Clustering Hierarchy" (LEACH) [7] and the protocol called "Power-Efficient Gathering in Sensor Information Systems" (PEGASIS) [10], they are two of the most used routing protocols. LEACH is a cluster-based protocol, which includes distributed cluster formation. It randomly selects a few sensor nodes as clusterheads and rotate this role to evenly distribute the energy load among the sensors in the network. The basic idea of the protocol PEGASIS is that in order to extend network lifetime, nodes need only communicate with their closest neighbors and they take turns in communicating with the base station. When the round of all nodes communicating with the base station ends, a new round will start, and so on. This reduces the power required to transmit data per round as the power draining is spread uniformly over all nodes.

Moreover, very recently, some researchers have proposed routing protocols based on the ACO algorithm, among them we can mention the following:

In [1] a Quality of Service (QoS) routing solution is proposed, it is named ACO based QoS routing algorithm (ACO-QoSR), it searches for the best paths that satisfied the QoS requirements by using intelligent artificial ants. ACO-QoSR algorithm is the tradeoff between a certain guaranteed QoS requirements and acceptable computational complexity.

In [17] is proposed a pheromone based energy-aware directed diffusion algorithm (PEADD) for WSN that extend the network lifetime by using pheromone and enhances the network reliability by maintaining remaining energy distribution relatively uniform among sensor nodes.

In [5] are shown the properties and review the main instances of network routing algorithms whose bottom-up design has been inspired by collective behaviors of social insects such as ants and bees.

In [12] authors proposed an algorithm based on ACO for flat architectures and localization awareness. This proposal tries to maximize the network lifetime and deal, react and adapt itself to changes in the network.

In [15] an Energy Delay Based on ACO (E&D) whose main goal is to find the optimal routing not only to maximize the lifetime of the network but also to provide real-time data transmission services.

In [14] the authors proposed the called Ant Colony Optimization-Based Location-Aware Routing (ACLR) which is a flat and location awareness algorithm. It fuses the residual energy and the global and local location information of nodes, to define the probability to select the next hop for the ants.

In [2] is proposed an Energy-Efficient Ant-Based Routing Algorithm (EEABR) for flat and location awareness architectures. In this proposal, the ants look for less energy consuming paths meanwhile reducing the size of the ants during the communication among nodes. Other similar ACO-based routing algorithms for WSN are published in [9] where authors shown crucial biologically inspired mechanisms and the associated techniques for resolving routing in WSN, including ant based and genetic approaches.

In [11] a novel routing approach using an ACO algorithm is proposed for WSN consisting of stable or limited mobile nodes. This approach is also implemented to a small sized hardware component as a router chip.

In [16] a routing algorithm for data aggregation based on ACO (ACAR) is presented. The main idea of this algorithm is optimization of data aggregation route by some cooperation agents called ants using the three heuristic factors about energy, distant and aggregation gain.

Dorigo and Di Caro show a method in [3], which is called Ant-Net, based on ant's colonies. In this method the information that the ants provide appears in each node like a routing table and a data structure that involve information about local traffic and delay quantities. Below presents the basic ACO algorithm and the two algorithms we are going to compare.

3 Ant Colony Optimization-Based Routing Algorithms for WSN

The main characteristic of an ACO routing algorithm consists in the continual acquisition of routing information through path sampling by using small control packages called *ants*. The ants are placed inicially in the source node s_o with the task of find out paths through the other nodes to the destination node s_b. An ant going from the source node to the destination node, collects information about the quality of the path, and it uses this information to update the pheromone levels of the intermediate nodes, reinforcing the pheromone of the good paths, creating a form of distributed reinforcement learning based on stigmergy [5].

3.1 General Outline of the ACO Based Algorithms

Let us assume that a WSN consists of m static and identical wireless sensors (nodes). The nodes are uniformly distributed in a flat region. The communication area covered by each node is represented by a circle whose radius is r.

A WSN is formally described as a weighted undirected graph *G(V, E, L)*. Where

- V is the set of sensor nodes, $V = \{s_1, s_2, s_3, ..., s_m\}$.
- L is the set of weights.
- E is the set of edges, $E \subset V \times V \times L$, for example, for any $s_i, s_j \in V$, $i \neq j$, $(s_i, s_j, \psi_{ij}(t)) \in E$, where $\psi_{ij}(t))$ is the cost to deliver a data package from s_i to s_j in the time t, in this case $\psi_{ij}(t)$ is the pheromone between s_i and s_j.

Any node s_i has a set of neighbors, defined by:

$$N(s_i) = \{s_j | s_j \in V, d_{ij} \leq r\} \tag{1}$$

r is the wireless communication coverage of the nodes. d_{ij} is the distance between s_i and s_j where the coordinates of the node s_i are x_i and y_i, and the coordinates of the node s_j are x_j and y_j and is calculated by:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{2}$$

The algorithm is composed of two phases. In the first phase it starts with a set of ants placed in the source node s_o. When the ant k is at the node s_i at instant t, the next-hop node $s_j \in N(s_i)$ will be selected randomly with a probability $P_{ij}^k(t)$ calculated by:

$$P_{ij}^k(t) = \frac{[\psi_{ij}(t)]^\alpha \times [\xi_{ij}(t)]^\beta}{\sum_{s_l \in N(s_i)} [\psi_{il}(t)]^\alpha \times [\xi_{il}(t)]^\beta} \tag{3}$$

Where ξ_{ij} is the location function defined by:

$$\xi_{ij} = \frac{1}{d_{ij}} \tag{4}$$

$\psi_{ij}(t)$ is the level of pheromone between node s_i and node s_j in the time t. α and β are the adjustable weights of $\psi_{ij}(t)$ and $\xi_{ij}(t)$, respectively.

When ant k reaches the destination node s_b phase two begins. The ant k goes back following the same route, depositing an increment of pheromone on that. This increment of pheromone is defined as follows:

$$\psi_{ij}(t+1) = (1 - p(t)) \times \psi_{ij}(t) + \Delta\psi_{ij} \tag{5}$$

Where $p(t)$ refers to the pheromone evaporating rate in the time t and $\Delta\psi_{ij}$ is the pheromone increment on the route between s_i and s_j in the current round travel.

$$\Delta\psi_{ij} = \sum_{k=1}^{n} \Delta\psi_{ij}^k \tag{6}$$

$\Delta\psi_{ij}^k$ is the pheromone that ant k laid on the route between the node s_i and the node s_j in the current round travel giving by.

$$\Delta\psi_{ij}^k = \begin{Bmatrix} \frac{1}{L^k} \text{ if } k \text{ passed from } s_i \text{ to } s_j \\ 0 \qquad\qquad \text{otherwise} \end{Bmatrix} \tag{7}$$

L^k refers to the length of the route founded by ant k. When ant k has returned to the suorce node s_o the ant k is eliminated.

Below we describe the routing algorithms that we are going to compare.

3.2 Ant Colony Optimization-Based Location-Aware Routing for Wireless Sensor Networks (ACLR)

ACLR [14] tries to find an equilibrium between the lifetime and the delay of the transmissions. In ACLR not all the s_i's neighbors are candidates to be selected to be the next-hop, only the nearer neighbors to the destination node are candidates. This set is defined by:

$$C(s_i) = \{s_j | s_j \in N(s_i), d_{jb} \leq d_{ib}\}. \tag{8}$$

Each node in the WSN has a memory block in which the residual energy, the location information of the node, its neighbors and the destination node are stored. Each ant is a mobile agent that has a contraindication list to memorize the nodes traversed by itself in a round travel. This contraindication list avoids to select the nodes which have been traversed by the ant. The algorithm ACLR is composed of two phases. In the first phase ants walk from the node s_o to the node s_b and the second phase is when ants return from s_b to s_o. The two phases are described below:

First phase: every ant follows the proposed routing scheme. The ant k starts to look for a route from the source node s_o to the destination node s_b. The new transition probability formula that allows ant k to select the next node, is defined as follows:

$$P_{ij}^k(t) = \frac{[\psi_{ij}(t)]^\alpha \times [\xi_{ij}]^\beta \times [\eta_{ij}(t)]^\gamma}{\sum_{s_l \in C(s_i)} [\psi_{il}(t)]^\alpha \times [\xi_{ij}]^\beta \times [\eta_{il}(t)]^\gamma} \tag{9}$$

This transition probability formula is composed not only for location ξ_{ij} and pheromone $\psi_{ij}(t)$ metrics but an energy metric $\eta_{ij}(t)$, which tries to maximize the network lifetime. The location function proposed by ACLR, ξ_{ij} is defined by:

$$\xi_{ij} = (\frac{d_{ob}}{d_{oi} + d_{ij} + d_{jb}}) \times (1 - \frac{d_{ij}}{\sum_{s_l \in C(s_i)} d_{il}}) \tag{10}$$

Where d_{ob} is the distance between node s_o and node s_b, d_{oi} is the distance between node s_o and node s_i, d_{ij} is the distance between node s_i and node s_j, d_{jb} is the distance between node s_j and node s_b and d_{il} is the distance between node s_i and node s_l. $\eta_{ij}(t)$ is the energy function proposed by ACLR and is defined as follows:

$$\eta_{ij}(t) = \frac{e_j(t)}{\sum_{s_l \in C(s_i)} e_l(t)} \tag{11}$$

If there is not any next-hop neighbor to select, that is $C(s_i)$ is empty, then ant k returns to the previous-hop node. And s_i is added to the contraindication list of the ant k. When ant k reaches the destination node s_b, a route R^k from the source node to the destination node is found by ant k. L^k is the length of R^k.
Second phase: Each ant returns to the source node from the destination node along the route R^k. At the meantime, ant k updates the pheromone on each segment of R^k, following the increment of pheromone defined in formula (5). The ACLR defines $\Delta\psi_{ij}^k$ as follows:

$$\Delta\psi_{ij}^k = \left\{ \begin{array}{cc} \frac{d_{ob} \times Q}{(d_{oi}+d_{ij}+d_{jb})L^k} & \text{if } s_i \text{ to } s_j \in R^k \\ 0 & \text{otherwise} \end{array} \right\} \tag{12}$$

Where Q is a constant, d_{ob}, d_{oi}, d_{ij} and d_{jb} have the same meaning as that of formula (10), respectively. L^k is the length of the route that is found by the ant k in the current round travel.

When the ant k has returned to the suorce node s_o, the ant k is eliminated. When all ants have been eliminated, a new iteration of the algorithm is repeated until it reaches a certain number of iterations.

In the experiments conducted in [14], it should be noted that they use an arbitrary value for energy consumption. They assume that it consumes one unit energy to directly deliver a data package between two nodes. They are not careful on the size of the data packages sent between nodes (ants). This size is vital to calculate the energy consumption in the WSN. Since the data packages they use are of variable size, the energy consumption they calculate is not correct, because it is not the same energy consumption for delivery small data packages that large data packages. To be able to make a fair comparison between algorithms, we must be careful in the way of calculate the energy consumption of each algorithm. Below it is described the following routing algorithm that we will compare.

3.3 Energy-Efficient Ant-Based Routing Algorithm (EEABR)

EEABR [2] is an ACO based routing protocol for WSN, which considers the energy efficiency of the underlying algorithm in order to maximize the networks lifetime. It has been proved that the tasks performed by the sensor nodes that are related with communications (transmitting and receiving data), spend much more energy than those related with data processing and memory management. Since one of the main concerns in WSNs is to maximise the lifetime of the network, it would be preferable that the routing algorithm could perform as much processing as possible in the network nodes, than transmitting all data through the ants to the base station to be processed there. In fact, EEABR tries to minimize the data package size as much as posible.

This algorithm makes a strong difference between forward ants (from the source node s_o to the destination node s_b) and backward ants (from the destination node s_b to the source node s_o). The ant's memory R^k only remember the last two nodes visited, this allows to have a constant ant size. When the ant k pass through the node s, the node s is the responsible of memorize the next information of the ant k: the previous visited node, the forward node, the ant identification and a timer. EEABR is composed by two phases: first phase concerning the forward ants and second phase concerning the backward ants.

First phase (forward ants): when a node s received an ant k, the node s looks into its memory and searches the ant identification in order to avoid the creation of a possible loop. If there is not such record, the node s saves the required information of k and initializes a timer for the ant k. The ant k decides which node is the most convenient to jump to, according to the transition probability formula that is computed as follows:

$$P_{ij}^k(t) = \frac{[\psi_{ij}(t)]^\alpha \times [\eta_{ij}(t)]^\gamma}{\sum_{s_l \in C(s_i)} [\psi_{il}(t)]^\alpha \times [\eta_{il}(t)]^\gamma} \tag{13}$$

Where $\psi_{ij}(t)$ is the level of pheromone between the node s_i and the node s_j, calculated in the same way that (5). α and γ are the adjustable weights of $\psi_{il}(t)$ and $\eta_{il}(t)$, respectively. $\eta_{ij}(t)$ is the energy function defined as follows:

$$\frac{1}{C - e_j} \tag{14}$$

C is the initial energy level of the nodes, and e_j is the energy of sensor j. If the ant k had passed through the node s, then the ant k is eliminated. When the ant k reaches the destination node s_b, a backward ant is created with the forward ant identification as well the forward ant memory.

Second phase (backward ants): the amount of pheromone that the backward ant will lay is calculated according to the EEABR proposed increment of pheromone of the ant k that is defined by:

$$\Delta\psi_{ij}^k = \left\{ \begin{array}{cc} \frac{1}{\left(C - \left[\frac{Emin^k - Fd^k}{Eavg^k - Fd^k}\right]\right) \times \varphi Bd^k} & k \text{ passed } s_i, s_j \\ 0 & \text{otherwise} \end{array} \right\} \tag{15}$$

Where $Emin^k$ is the minimum energy level registered by the forward ant k through its route. $Eavg^k$ is the average energy of the visited nodes so far registered by forward ant k through its route. Fd^k represents the number of nodes that the forward ant k has visited so far. φ is a coefficient and Bd^k is the traveled distance (the number of visited nodes) by the backward ant k until node i.

The timer is used to delete the record that identifies the backward ant, if for any reason the ant does not reach that node within the time defined by the timer. It is important to note that the key point of EEABR algorithm is to minimize the size of data packages transmitted between nodes. Therefore, to make a real and fair comparison between the ACLR and EEABR algorithms, we must be careful in the way of calculate the energy consumption.

4 Experiments

In this Section we present the experimental comparison between the algorithms ACLR and EEABR. To conduct our experiments, we consider the nodes Mica2 Motes of the company Crossbow [1]. We know that the energy required to transmit one bit between two nodes is 4.28 $\mu joules$ and the energy required to receive one bit is 2.36 $\mu joules$ [13] . It is a fact that nodes consume energy simply because of being on, even off or asleep, however, in our simulation the nodes are always on, so this consumption of energy can be disregarded and only take into account the energy consumption to transmit and receive data packages. With these energy consumption values, it can be established that the comparison between the protocols ACLR and EEABR would be more in line with reality and more fair than the one made in their original proposals [14,2]. For this sake, we execute the experiments using three different scenarios that were explained in [2]. We also evaluate three metrics used in [14] to determine the performance of the routing algorithms. These scenarios and metrics are defined as follows:

Scenarios. ***First scenario:*** all nodes start with the same initial energy level, there is only one source node s_o and the destination node s_b is fixed. ***Second scenario:*** all nodes begin with the same initial energy level, the source node s_o changes randomly at each iteration and the destination node s_b is fixed. ***Third scenario:*** the nodes's initial energy level is randomly selected, the source node s_o changes randomly at each iteration and the destination node s_b is fixed.

Metrics. ***Energy Consumption:*** this metric refers to the total used energy in the network in the process of finding the optimal routes from the source node s_o to the destination node s_b. To be fair in the comparision, we use the energy consumption per bit that is shown in [13], for the transmitting and receiving data in each node. As mentioned in the previous section, the energy required to transmit one bit between two nodes is 4.28 $\mu joules$ and the energy required to receive one bit is 2.36 $\mu joules$. ***Latency:*** the time it takes a data package to be sent from the source node s_o to the destination node s_b it is called latency, it is the sum of temporal delays into the network. This metric is commonly calculated as the average of number of nodes visited per route. ***Energy Efficiency:*** it refers to the ratio of the number of data packages received at s_b by the total consumed energy.

Parameter values. The parameters' values are the same for both algorithms, established as follows: The deployed field has a surface area of 300 x 200 (m^2).

[1] "http://www.xbow.com/"

10000 nodes deployed uniformly. Number of ants equal to 20. Wireless communications radius of sensors is r=30 (m). $\alpha = 1, \gamma = 1$. For the algorithm ACLR $\beta = 1$ and $Q = 1$. For the algorithm EEABR $\varphi = 1$ and the pheromone evaporating rate $p(t)$ equal to 0.95. The initial pheromone level for every pair of adjacent nodes is set to $\psi_{ij}(0) = 0.01$.

The energy consumption per bit transmited (as defined by [13]) is 4.28 $\mu joules$, and per bit received is 2.36 $\mu joules$. We performed 50 iterations in each experiment and for both algorithms we ran 20 independent experiments for each scenario.

5 Results

The metric results for the first scenario are shown in Figures 1, 2 and 3 for Energy Consumption, Latency and Energy Efficiency, respectively. It seems that each routing algorithm has an especial focus, clearly the algorithm EEABR outperforms the Energy Consumption in a better way than ACLR, this is a result of its ant sizes, meanwhile the algorithm ACLR outperforms by some (2 or 3 nodes) the EEABR in all the 20 experiments in the Latency metric, wich means that ACLR focuses on minimize the delay in data transmision. In the Energy Efficiency metric, EEABR is more efficient than ACLR.

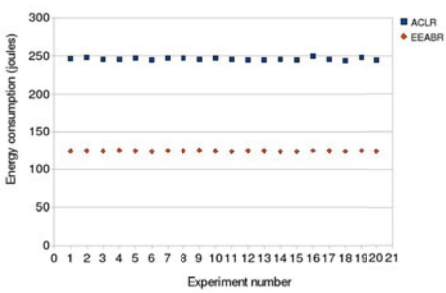

Fig. 1. First scenario. Energy consumption.

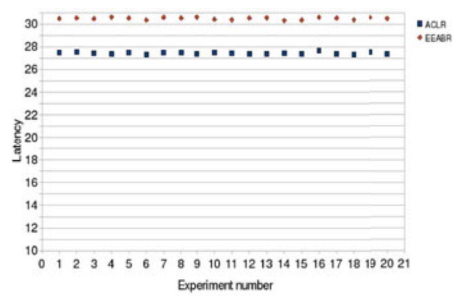

Fig. 2. First scenario. Latency.

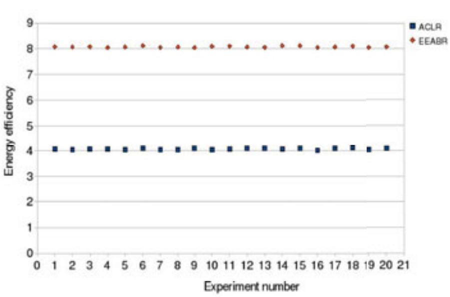

Fig. 3. First scenario. Energy efficiency.

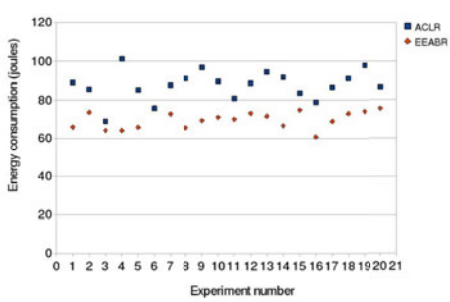

Fig. 4. Second scenario. Energy consumption.

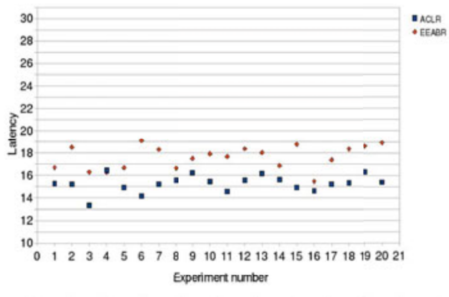

Fig. 5. Second scenario. Latency

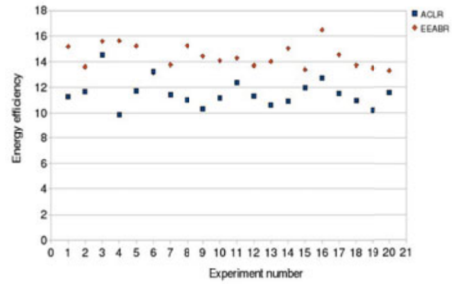

Fig. 6. Second scenario.Energy efficiency

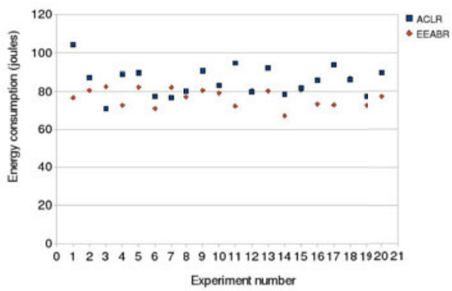

Fig. 7. Third scenario. Energy consumption

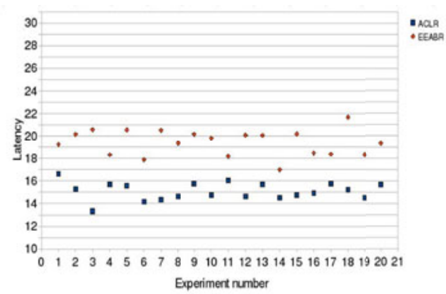

Fig. 8. Third scenario. Latency

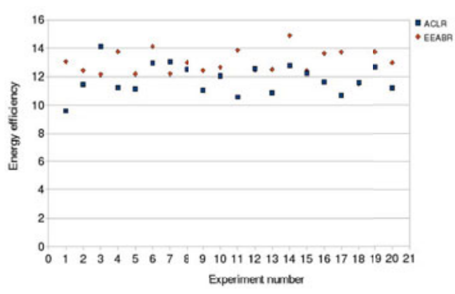

Fig. 9. Third scenario. Energy efficiency

The results for Energy Consumption, Latency and Energy Efficiency metrics are shown in Figures 4, 5 and 6, respectively for the second scenario. For the Energy Consumption metric in this second scenario, the EEABR outperforms ACLR in most of the cases (18 out of 20) with a difference less than the previous scenario. Both algorithms present better values in general than their respective metric in the first scenario. This is because the source node is constantly changing its position, therefore the found paths are more diverse. For the Latency metric case, ACLR outperforms EEABR for most of the experiments. Furthermore, notice that in this second scenario both algorithms perform better than the first scenario. For the Energy Efficiency metric we can observe again that EEABR is better than the ACLR (18 out of 20). Again, the results for this second scenario are consistent with those from the first scenario.

In the third scenario, the metric values for Energy Consumption, Latency and Energy Efficiency metrics are shown in Figures 7, 8 and 9, respectively. We can observe that the Energy Consumption values are a little higher for both algorithms than their respective metric in the second scenario, with similar performance between them. Notice that the latency values for both algorithms are similar in this scenario and similar to the second scenario as well and still ACLR outperforms EEABR. Both algorithms have similar performance for the Energy Efficiency metric in this scenario, being a little bit better EEABR than ACLR (12 out of 20).

6 Conclusion

We can conclude, in general terms, the ACLR and EEABR presented good performances in terms of the defined metrics. However, ACLR provides better results finding shorter routes than EEABR, i.e. two or three nodes less on average. EEABR has shown better performance in Energy Consumption. If the source node is changing its position, both algorithms presented better Energy Consumption than the case when it is fixed. For the scenarios where the source node is changing, the algorithms show better Latency performance than scenarios where the source node is fixed. The algorithms are more efficient, in terms of Energy Efficiency, if the source node is changing for each iteration. For the third scenario, where the nodes's initial energy values are set to different values, the Energy Consumption is very similar for both algorithms, as well the Energy Efficiency. In general, the algorithms present very consistent performance, however EEABR is more efficient maximizing the network's lifetime. This efficiency is due to the usage of slim ants. The results shown that the bio-algorithms to solve the routing problem in Wireless Sensor Networks are viable options.

Acknowledgements

The authors acknowledge support from CONACYT through project 107688 and SIP-IPN 20100273.

References

1. Cai, W., Jin, X., Zhang, Y., Chen, K., Wang, R.: ACO Based QoS Routing Algorithm for Wireless Sensor Networks. In: Ma, J., Jin, H., Yang, L.T., Tsai, J.J.-P. (eds.) UIC 2006. LNCS, vol. 4159, pp. 419–428. Springer, Heidelberg (2006)

2. Camilo, T., Carreto, C., Silva, J.S., Boavida, F.: An Energy-Efficient Ant-Based Routing Algorithm for Wireless Sensor Networks. In: Dorigo, M., Gambardella, L.M., Birattari, M., Martinoli, A., Poli, R., Stützle, T. (eds.) ANTS 2006. LNCS, vol. 4150, pp. 49–59. Springer, Heidelberg (2006)
3. Dorigo, M., DiCaro, G.: Ant Net: A Mobile Agents Approach to Adaptive Routing Technical. IRIDIA Free Brussels University, Belgium (1997)
4. Dorigo, M., Gambardella, L.M.: Ant colony system: A cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary Computation 1(1), 53–66 (1997)
5. Farooq, M., Caro, G.A.: Routing Protocols for Next-Generation Networks Inspired by Collective Behaviors of Insect Societies An Overview. Swarm Intelligence, 101–160 (2008)
6. Heimfarth, T., Janacik, P.: Experiments with Biologically-Inspired Methods for Service Assignment in Wireless Sensor Networks. Biologically-Inspired Collaborative Computing 268, 71–84 (2008)
7. Heinzelman, W., Balakrishnan, H.: Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In: Proceedings of the 33rd Hawaii International Conference on System Sciences (2000)
8. Huo, H., Gao, D., Niu, Y., Gao, S.: ASDP An Action-Based Service Discovery Protocol Using Ant Colony Algorithm in Wireless Sensor Networks. In: Zhang, H., Olariu, S., Cao, J., Johnson, D.B. (eds.) MSN 2007. LNCS, vol. 4864, pp. 338–349. Springer, Heidelberg (2007)
9. Iyengar, S., Wu, H., Balakrishnan, N., Chang, S.: Biologically Inspired Cooperative Routing for Wireless Mobile Sensor Networks. IEEE Systems 1(1), 29–37 (2007)
10. Lindsey, S., Raghavendra, C.: PEGASIS: Power-Efficient Gathering in Sensor Information Systems. IEEE Aerospace Conference Proceedings, 1125–1130 (2002)
11. Okdem, S., Karaboga, D.: Routing in Wireless Sensor Networks Using an Ant Colony Optimization ACO Router Chip. Sensors, 909–921 (2009)
12. Reza, G., Rahman, A., Gueaieb, W., Saddik, A.: Ant Colony-Based Reinforcement Learning Algorithm for Routing in Wireless Sensor Networks. IEEE (1-4244-0589-0) 1–6 (2007)
13. Torres, M.G.: Energy Consumption in Wireless Sensor Networks Usig GSP. Master's thesis, Universidad Pontificia Bolivariana, Medellín, Colombia (2006)
14. Wang, X., Li, Q., Xiong, N., Pan, Y.: Ant Colony Optimization-Based Location-Aware Routing for Wireless Sensor Networks. In: Li, Y., Huynh, D.T., Das, S.K., Du, D.-Z. (eds.) WASA 2008. LNCS, vol. 5258, pp. 109–120. Springer, Heidelberg (2008)
15. Wen, Y., Chen, Y., Qian, D.: An Ant-based approach to Power-Efficient Algorithm for Wireless Sensor Networks. 1546–1550 (2007)
16. Ye, N., Shao, J., Wang, R., Wang, Z.: Colony Algorithm for Wireless Sensor Networks Adaptive Data Aggregation Routing Schema. In: Li, K., Fei, M., Irwin, G.W., Ma, S. (eds.) LSMS 2007. LNCS, vol. 4688, pp. 248–257. Springer, Heidelberg (2007)
17. Zhu, X.: Pheromone Based Energy Aware Directed Diffusion Algorithm for Wireless Sensor Network. In: Huang, D.-S., Heutte, L., Loog, M. (eds.) ICIC 2007. LNCS, vol. 4681, pp. 283–291. Springer, Heidelberg (2007)