

ECO-BAT: A New Routing Protocol for Energy Consumption Optimization Based on BAT Algorithm in WSN

Mohammed Kaddi^{1,*}, Abdallah Banana² and Mohammed Omari¹

¹LDDI Laboratory, Mathematics and Computer Science Department, University of Adrar, Adrar, 01000, Algeria

²Mathematics and Computer Science Department, University of Adrar, Adrar, 01000, Algeria

*Corresponding Author: Mohammed Kaddi. Email: kadimohamed1983@yahoo.fr

Received: 15 June 2020; Accepted: 26 June 2020

Abstract: Wireless sensor network (WSN) has been widely used due to its vast range of applications. The energy problem is one of the important problems influencing the complete application. Sensor nodes use very small batteries as a power source and replacing them is not an easy task. With this restriction, the sensor nodes must conserve their energy and extend the network lifetime as long as possible. Also, these limits motivate much of the research to suggest solutions in all layers of the protocol stack to save energy. So, energy management efficiency becomes a key requirement in WSN design. The efficiency of these networks is highly dependent on routing protocols directly affecting the network lifetime. Clustering is one of the most popular techniques preferred in routing operations. In this work we propose a novel energy-efficient protocol for WSN based on a bat algorithm called ECO-BAT (Energy Consumption Optimization with BAT algorithm for WSN) to prolong the network lifetime. We use an objective function that generates an optimal number of sensor clusters with cluster heads (CH) to minimize energy consumption. The performance of the proposed approach is compared with Low-Energy Adaptive Clustering Hierarchy (LEACH) and Energy Efficient cluster formation in wireless sensor networks based on the Multi-Objective Bat algorithm (EEMOB) protocols. The results obtained are interesting in terms of energy-saving and prolongation of the network lifetime.

Keywords: WSNs; network lifetime; routing protocols; ECO-BAT; bat algorithm; CH; energy consumption; LEACH; EEMOB

1 Introduction

The unbridled need for information and the rapid evolution of microelectronics and wireless technologies have resulted in a massive increase in wireless technology. Also, ad-hoc architecture represents one of the most used implementations in the world of wireless technology. Its characteristic of self-organization and the evolution of the topology allow it to develop several infrastructures such as mobile networks, sensor networks, and lately Internet of Things (IoT).



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Microelectronics, wireless communication, but also miniaturizations were the main players giving birth to the WSN. The development of WSN was originally motivated by military applications (battlefield surveillance, location of the enemy...). Nevertheless, their remarkable performances in terms of reliability and low cost have made it possible to benefit their use in the field of civil application (environmental monitoring, industry, home automation, health...). However, new problems are emerging that make wireless networks less reliable than wired networks. Thus new techniques must be put in place to mitigate this kind of situation.

Energy saving is one of the major challenges to be overcome in order to achieve a successful WSN. Indeed, the sensor nodes are generally powered using a small battery limited in power, and the replacement of it is a very difficult or impossible task. Therefore, the depletion of the energy reserves of the sensor nodes implies the decommissioning of the entire network. Hence, batteries play a significant role in the sensor environment, i.e., the lifetime of the node is determined using batteries. In a sensor network, energy-efficient routing protocols are required. Numerous investigations are carried out to design a sensor network to increase the energy efficiency of nodes [1]. The purpose of this paper is to address the problem of routing in WSN, and particular hierarchical routing based on the cluster principle. The nodes are organizing into clusters, for each cluster one has a cluster leader who is responsible for collecting the information from member nodes and sending them to a master station. The choice of CH and cluster formation is an optimization problem, especially on large networks.

In order to solve this problem, we propose in this paper a meta-heuristic optimization method for clustering based on the bat algorithm. An objective function was used to reach an optimal solution.

The remainder of the paper is organized as follows. In Section 2, the bat algorithm has been detailed. Section 3 describes some similar works. Section 4 details our proposed scheme. In Section 5, we present the ECO-BAT protocol procedure. The results of the comprehensive performance evaluation are presented and discussed in Section 6. Finally, in Section 7, we will conclude with a conclusion and some perspectives.

2 Bat Algorithm

Based on bats' behaviors, Yang [2] developed a new meta-heuristic search algorithm, called the BAT algorithm. The bat algorithm attempts to model the echolocation behavior of micro-bats. It is essentially based on the following scenario: Bats fly randomly to find food, they do not know where their prey is, however they can calculate the distance between each of them. They target using echolocation. Their common goal is not only to reach their prey but also to find the best hunting strategy. The most effective strategy is to follow the bat closest to the flock. This behavior is similar to solving an optimization problem.

The bat algorithm is an optimization method where all bats collaborate in populations to achieve a collective goal. The latter is formulated by a function called the objective function. Each of the flying bats is considered a solution in the search space called a candidate or local solution. This will be evaluated at each iteration of the algorithm by calculating its fitness by the objective function and updated by adjusting its position, its velocity, and its frequency, according to the best position it has reached and that found by the whole population. So finally the algorithm maintains the best solution reached among the whole so-called global solution after a predefined number of iterations.

Formally, Bats fly randomly with a velocity v_i at a position x_i with a fixed frequency f_{min} , varying the wavelength λ and the loudness A_0 to search for prey. They can automatically adjust the wavelength (or frequency) as well as the pulse emission rate $r \in [0,1]$, depending on the proximity of their prey. Although the loudness can vary in several ways, the algorithm assumes that $A \in [A_0, A_{min}]$.

The bat algorithm is essentially composed of the following three steps, which will be repeated until a certain number of iterations are reached:

- Evaluate the fitness of each bat.
- Update the best local and global solutions.
- Update the positions, velocities, and frequencies of each of the bats.

The first two steps are fairly straightforward. The fitness assessment of each of the bats is performed using the objective function. While updating the best local and global solutions is done by comparing the current fitness of each of the bats with the fitness of the best solution it has encountered so far, then choosing the best of them they. As for the search for the global solution, it will be enough to compare all the best local solutions and keep the best.

The third step is the most crucial movement of bats is responsible for the efficiency of the algorithm. The rules defining the new solution and updating the position and velocity of each of the bats in a space of d dimensions are as follows:

$$f_i = f_{min} + (f_{max} - f_{min})\beta \quad (1)$$

$$v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i \quad (2)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (3)$$

where

- f_i is the frequency which belongs to the range $[f_{min}, f_{max}]$ corresponding to the wavelength range $[\lambda_{min}, \lambda_{max}]$.
- $\beta \in [0,1]$ is a random vector drawn from a uniform distribution.
- x_* is the best current global location (solution), which is calculated after comparing all the solutions obtained by each of the n bats.

For local search, once a solution is selected from the best current solutions, a new solution for each bat is generated locally using the random walk:

$$x_{new} = x_{old} + \epsilon A^t \quad (4)$$

where $\epsilon \in [0,1]$ is a random number, while $A^t = \langle A_i^t \rangle$ is the average loudness of all bats at time t .

Besides, the loudness $A_i \in [A_0, A_{min}]$ and the rate r_i of pulse emission must be updated at each iteration of the algorithm. The more the loudness decreases, which indicates that the bat has found its prey, the more the rate of pulse emission increases, and this during the pursuit of the prey. For example and for simplicity, we can use $A_0 = 1$ and $A_{min} = 0$, such that $A_{min} = 0$ indicates that it has found its prey and has temporarily stopped emitting any sound. Indeed we have:

$$A_i^{t+1} = \alpha A_i^t, \quad r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)] \quad (5)$$

where α and γ are constants. For any $0 < \alpha < 1$ and $\gamma > 0$, we have

$$A_i^t \rightarrow 0, \quad r_i^t \rightarrow r_i^0, \quad \text{as } t \rightarrow \infty \quad (6)$$

In order to simplify the implementation, it is possible to assume that $\alpha = \gamma$.

Based on these rules and explanations, the basic steps of the bat algorithm can be summarized as follows:

Algorithm 1: Pseudocode of the bat algorithm

Objective function $f(x)$, $x = (x_1, x_2, \dots, x_d)^T$
Initialize bat population $x_i (i = 1, 2, \dots, n)$ and the velocity of each of the bats v_i
Define pulse frequency f_i at x_i
Initialize pulse rates r_i and loudness A_i
While ($t < \text{maximum number of iterations}$)
Generate new solutions by adjusting frequency,
and updating velocities and location/solutions {see Eq. (1) to Eq. (3)}
 if ($\text{rand} > r_i$)
 Select a solution among the best solutions
 Generate a local solution around the selected best solution
 End if
 if ($\text{rand} < A_i \ \& \ f(x_i) < f(x_*)$)
 Accept new solutions
 Increase r_i and reduce A_i
 End if
Rank the bats and find current best x_*
End while
Display results.

3 Related Works

3.1 Variants of Bat Algorithms

The standard bat algorithm has many advantages, and one of the main advantages is that it allows very fast convergence at a very initial stage. This makes it an efficient algorithm for applications such as classifications and others when quick solutions needed. However, to allow the algorithm to go too quickly into the operating phase we vary A and r too quickly, which can lead to stagnation after the first stage. To improve performance, many methods and strategies have been attempted to increase the diversity of the solution and thus improve performance, which has produced some good and effective variants of the bat algorithm. These variants can be mentioned [3]:

- FLBA (Fuzzy Logic Bat Algorithm [4]: By introducing fuzzy logic into the bat algorithm, it called this variant Fuzzy Logic Bat Algorithm.
- MOBA (Multi-Objective Bat Algorithm) [5]: It is an extension of the bat algorithm for multi-objective optimization, which validated their effectiveness in solving design problems in engineering benchmarks.
- BBA (Binary Bat Algorithm) [6,7]: It developed a discrete version of the bat algorithm to solve classification problems and characteristic choice problems.
- EBA (Evolving Bat Algorithm) [8]: It is a new method in the field of swarm intelligence to solve numerical optimization problems. In order to analyze the improvement of the precision of the search for the best solution and the reduction of the cost of calculation, three well-known and commonly used test functions in the field of swarm intelligence to test the precision and the performances of the algorithm are used in experiments.
- IBA (Improved Bat Algorithm) [9]: It proposed to solve continuous optimization problems by improving three basic Bat Algorithm approaches. The first approach initializes the bat population.

The second approach updates the frequency, velocity, and solution. The third approach updates the sound intensity and pulse emission rate. The innovative aspect of the proposed method is to find better fitness and cost values for the unconstrained and constrained problems respectively.

3.2 Others Related Works

LEACH protocol [10] is a most popular clustering algorithm, LEACH provides convincing energy savings and thus prolongs the life of the network contrasted with static clustering algorithms, and the drawback fundamental of this algorithm is that the CH selection is random. Therefore, it is plausible to choose a CH with low energy, which can die quickly and ultimately reduce network performance. Therefore, a variety of algorithms based on bio-inspired methods have been developed to improve this protocol.

Krishnaveni et al. [11] propose a protocol named EEMOB, where the idea behind the EEMOB algorithm is to increase the coverage rate and to minimize the energy consumption for the nodes and CH. In this model, all nodes are elected as CH depending on the trust level of the node. Cluster formation depends on the distance between the nodes and should be less than the detection range of the bat. The CH is the node that consumes the minimum energy, has a minimum distance, and effectively reaches the threshold of the detection range. EEMOB minimizes intra-cluster communication distance, maximizes power, coverage, and network lifetime over LEACH.

In order to select the optimal number of CH nodes, authors in [12] have proposed a new technique called beats Swarm Algorithm for Wireless Sensor Networks Lifetime Optimization BAT protocol. The proposed technique applies the meta-heuristic algorithm of optimization by the swarm of bats. It optimizes the network as a nonlinear problem to select the optimal number of CH nodes. The objective function used to minimize the intra-cluster compactness with a minimum distance between the nodes in the same cluster. The proposed technique is simulated and applied and compared to the LEACH protocol. The results show that this proposed technique surpasses LEACH. It effectively optimizes the selection of CH nodes ensuring optimal coverage and connectivity based on intra-cluster distances. This reduces power consumption and thus increases the lifespan of each node, which results in a significant extension of the lifespan of the WSN.

The main objective of the protocol called Improved Bat Algorithm Based Energy Efficient Congestion Control Scheme for Wireless Sensor Networks presented in [13] is to develop a model to find the optimized path as a function of the distance between the source and the destination and the residual energy of the node in order to control congestion in WSN at the transport layer. The algorithm was applied to the fitness function to obtain an optimal solution. The simulation results showed an improvement in parameters such as network lifetime and throughput compared to the CODA (Congestion Detection and Avoidance), PSO (Particle Swarm Optimization) and ACO (Ant Colony Optimization) algorithms.

The algorithm proposed in [14] is a clustering algorithm with high energy efficiency for WSN using the bio-inspired Firefly algorithm (FAA) implemented at the BS. A new objective function has been defined to minimize the intra-cluster distance and optimize the energy consumption of the network. The proposed protocol is presented in comparison with the previous LEACH and PSO-C (Centralized Particle Swarm Optimization) protocols.

The protocol HBA (an efficient Hybrid BAT-optimized clustering for wireless sensor networks presented in [15] offers a hybrid BAT algorithm with Differential Evolution (DE) to improve the efficiency of the LEACH protocol. The original BAT algorithm was hybridized with the DE algorithm to optimize an objective function based on the total transmission distance and the number of CHs, the fitness of the nodes increases when the distance decreases and the number of CHs decreases. The algorithm searches for the appropriate number of CHs and their location. The simulations revealed that the HBA has improved throughput, reduced delay, and packet retransmission compared to LEACH.

The construction of a Minimal Connected Dominating Set (MCDS) is good for packet routing and energy efficiency, for this an efficient approximation MCDS construction algorithm E-MCDS (energy-efficient MCDS construction algorithm) is proposed in [16] explicitly consider energy consumption. E-MCDS contains two stages: the CDS construction phase and the pruning phase. The first phase is approximately composed of two independent sets. The performance ratio of E-MCDS is analyzed in both the unit disk graph and disk graphs with bidirectional links. The experimental results show that E-MCDS performs well both in terms of energy efficiency and the size of CDS constructed.

To solve the problem of energy consumption and extend the network lifetime, Zhang et al. [17] propose a Virtual Multiple Input Multiple Output based Cooperative Routing algorithm (VMIMOCR). VMIMOCR chooses cooperative relay nodes based on Virtual Multiple Input Multiple Output Model, and balances energy consumption by reasonable power allocation among transmitters, and decides the forwarding path finally. The simulation results have shown that VMIMOCR can improve network lifetime from 37% to 348% in the medium node density, compared with existing routing algorithms.

To avoid the design discrepancy, Chen et al. [18] develop a new Slot-Assignment and Routing Framework (SARF). The new technique incorporates short-term channel fading statistics to optimize the long term slot assignment, scheduling and routing simultaneously. Hence multi-user diversity can be exploited more efficiently. The simulation results have shown that the performance of the proposed system can be up to 64.0% higher than the emerging scheme for wireless mesh technology proposed in the literature, and also depends less on the topology compared to it.

The following table (see Tab. 1) shows a comparison between these works.

Table 1: Comparison between related works

Protocol	Optimization algorithm	Energy optimization	Congestion control	Distance intra-cluster minimization	Network lifetime extension
LEACH [10]	LEACH	Medium	No	No	Pretty good
EEMOB [11]	Bat	Good	No	No	Good
BAT [12]	Bat	Good	No	Yes	Good
Improved Bat [13]	Improved Bat	Good	Yes	Yes	Good
FAA [14]	Firefly	Very good	No	Yes	Very good
HBA [15]	Bat + Differential Evolution	Very good	No	Yes	Very good
E-MCDS [16]	E-MCDS Construction Algorithm	Very good	No	Yes	Very good
VMIMOCR [17]	VMIMOCR Algorithm	Very good	No	Yes	Very good
SARF [18]	SARF approach + Heuristic Algorithm	Good	No	Yes	/

4 Proposed Scheme

Our proposed protocol called ECO-BAT; it is divided into several execution cycles. Each cycle begins with an initialization phase in which the CHs are elected and the clusters are formed using the bat algorithm.

Followed by a transmission phase where the collected data is transmitted to the CHs which in turn will transmit to the base station (BS).

4.1 Initialization Phase

In the initialization phase of the first cycle (Round) the cluster head election by the BS is random. In the rest of the cycles, the BS executes an iterative version of a bat optimization algorithm to determine the best k CHs of the current cycle. Then the non-CH nodes join the nearest CH to form the clusters.

The overall performance of CHs chosen by the algorithm is evaluated by the use of an objective function called fitness (see Eq. (7)). The smallest value of the fitness function represents the best combination of CHs among others. The Euclidean distance is used to calculate the distance between the CH and the cluster member node.

For the fitness function, the distance and energy are the two elements taken into consideration as follows:

$$\text{Fitness} = \alpha * f_1 + (1 - \alpha) * f_2 \quad (7)$$

where α is a constant of user-defined that calculates the intervention of each of the sub-objectives f_1 and f_2 used. f_1 is the sum of the residual energy ratios of all the nodes of the particular cluster to the residual energy of CH(k).

$$f_1 = \sum_{k=1}^m \frac{\sum_{j=1, j \neq k}^{n_c} E(j)}{E(k)} \quad (8)$$

where

- m is the number of CHs,
- n_c is the number of nodes of the particular cluster,
- $E(j)$ is the residual energy of the node j ,
- $E(k)$ is the remaining energy of the node CH(k).

The sub-objective f_2 is the total sum of the inter-cluster distances.

$$f_2 = \sum_{k=1}^m \sum_{j=1, j \neq k}^{n_c} d(j, k) \quad (9)$$

where $d(j, k)$ is the Euclidean distance between the CH(k) and the node j .

4.2 Transmission Phase

The transmission phase is similar in all the protocol cycles where the nodes transmit its collected data to their CHs, which will execute processing operations on the data (aggregation and compression of the data) in order to transmit them to the BS.

5 ECO-BAT Protocol Procedure

5.1 Step 1: Deployment Phase

1. The nodes and the BS are deployed randomly in the sensation zone;
2. The BS broadcasts a message to collect the information at all the nodes;

3. After receiving this message, the nodes begin to send their information: node Id (Identity), position and initial energy to the BS;
4. The BS uses this information to calculate the distance between its position and the position of the nodes;

5.2 Step 2: First Cycle (Iteration = 1)

5.2.1 Initial Phase

5. The BS randomly selects k CHs for the first round;
6. The selected CH broadcasts a warning message which contains its Id to all neighboring nodes; the CHs use the CSMA (Carrier Sense Multiple Access) MAC protocol in this phase, each of these nodes decides which group to join based on the power of the messages received from the CHs. The closer a node is to the CH, the stronger the signal received, indicating to the latter that it will need less energy to transmit its data to the CH. If two received signals have the same power, then the CH to join is chosen randomly;
7. Each node transmits a joining request to the chosen CH, composed of the node Id, the CH Id, and a header also using a CSMA MAC protocol;
8. Each CH calculates the number of nodes member of its group;

5.2.2 Data Transmission Phase

9. After the cluster is formed, the CH allocates a time interval for each member node to communicate (send/receive) using TDMA scheduling according to the number of member nodes in the cluster. And it is only during the time allocated to them that the nodes can transmit their data to the CH;
10. Once the TDMA program has been set up, the nodes begin to transmit their data to the chosen CH, after receiving the data from the member nodes, the CH performs processing on the data (aggregations and compressions) and then sent to the BS according to unicast communication (single hop);
11. Record the CHs information of first-round (Id, current energy, distance);

5.3 Step 3: Cycle (Round > 1)

5.3.1 Initial Phase

12. Selection of cluster head by executing the adapted algorithm of bats;
 - Initialize the population of bats (N nodes);
 - Initialize the pulsation emission rates r_i and the intensity A_i ;
 - Define the pulse frequency $Q_i \in [Q_{min}, Q_{max}]$;
 - Evaluate the fitness function $f(x^*)$ (see Eq. (7));
 - Generate a random number $rand(0, 1)$;

While ($t < \text{maximum number of iterations}$)

Generate new solutions by adjusting frequencies and updating velocities;

While ($h < \text{number of local solutions to generate}$) (see Eq. (1))

Randomly select a set C of k CHs among all the N eligible nodes;

For each node **do**

Calculate the distances between the node and the CHs;

Assign the node to the CH with a minimum distance;

Calculate the fitness function value $f(C)$ (see Eq. (7));

Save the result;
End For;
End While;
if ($rand > r$) **then**
 Update local solution (x_i^t);
End if
if ($rand < A_i$ & $f(x_i^t) < f(x^*)$) **then**
 Accept the solution;
 Save the result;
End if
End While
 Sort the results;
 $x^* \leftarrow x_i^t$ (The best set of k CHs obtained after the execution of the algorithm);
 13. Cluster formation: Repeat Steps 6 to 8;

5.3.2 Data Transmission Phase

14. Repeat Steps 9 and 10;
 15. End of the cycle: The cycle ends a new selection of CHs will be launched for the next round.
- The following flowchart (see Fig. 1) shows the adapted bats algorithm.

6 Simulation Results

6.1 Radio Model

We use the same model of radio signal propagation introduced in several research studies for energy consumption, which represents a first-order radio model.

$$E_{TX}(L, d) = \begin{cases} L * E_{elec}(L, d) + L * \epsilon_{fs} * d^2, & d < d_0 \\ L * E_{elec}(L, d) + L * \epsilon_{mp} * d^4, & d \geq d_0 \end{cases} \quad (10)$$

where

- $E_{TX}(L, d)$ represents the energy consumed by the transmitter to send a message of L bits over a distance of d meters.
- $E_{elec}(L, d)$ is the energy sufficient to send or receive a single bit over d meters.
- L is the size of a message to be transmitted.
- ϵ_{fs} and ϵ_{mp} is the energy of amplification that depends on the transmitter amplifier model.
- d_0 is the limit distance (threshold) for which the amplification factors change in value.

The distance d_0 is given by:

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \quad (11)$$

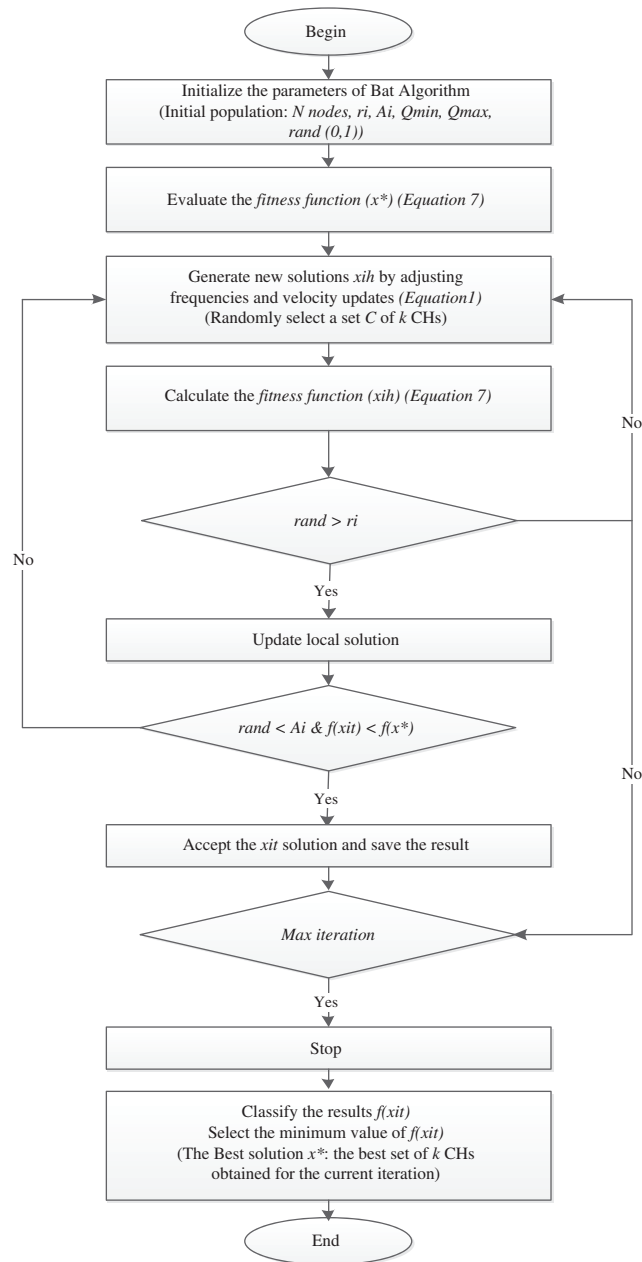


Figure 1: Flowchart of the adapted bat algorithm

In order to receive a data of L bits, the receiver consumes:

$$E_{RX}(L) = L * E_{elec} \quad (12)$$

6.2 Simulation Settings

Tab. 2 below shows the parameters used in the simulation and their values.

Table 2: Simulation parameters

Parameters	Values
Node deployment mode	Random
Deployment area	$100 \times 100 \text{ m}^2$
Number of nodes	200
Coordinate of the BS	(50,50)
Starting energy (E_0)	0.1 J/node
E_{elec}	50 nJ/bit
Data Packet Size	7200 bits
ε_{fs}	10 pJ/bit/m ²
ε_{mp}	0.0013 pJ/bit/m ⁴
Energy of Data Aggregation (EDA)	5 pJ/bit
Value of α	0,5
Pulse emission rate r_i	0,4
Loudness A_i	0,5
f_{min}	0
f_{max}	12

6.3 Evaluation Metrics

In order to demonstrate the effectiveness of our protocol, we have considered the following metrics to evaluate these three protocols.

- Network lifetime: We have calculated the lifetime of the network in terms of the number of iterations since the launch of the protocol up to the First Dead Node (FND), Half Nodes Dead (HND), and Last Node Dead (LND).
- Residual energy: Residual energy is defined as the total sum of the energy available in the sensor network compared to the lifetime of the network.
- Data transmitted to the BS: The CH collects the information sent by the cluster member nodes and transfers this information to the BS. This guarantees an optimal rate of packet transmission on the network.

6.4 Results and Discussion

In this subsection, we present the simulation results obtained. To examine the success of our proposed ECO-BAT protocol, we compared it with the two protocols proposed in the literature: LEACH [10] and EEMOB [11].

Fig. 2 shows a comparison between these three protocols in terms of the network lifetime: FND graph, HND graph, and LND graph for 200 nodes in terms of a round number.

In this figure, we see that the FND in the 28th round, HND in the 60th round, and the LND in the 100th iteration for the LEACH protocol. However, the EEMOB protocol loses its first node after about iteration number 60, 50% of node dead in a round which equals around 72 and the last one dies in a round which equals around 138. But our ECO-BAT protocol surpasses both protocols; it loses its first node in the 78th

round, HND in the 136th round, and the LND in the 207th round. So, these results demonstrate that the proposed protocol gives good results in terms of the network life cycle compared with LEACH and EEMOB protocols.

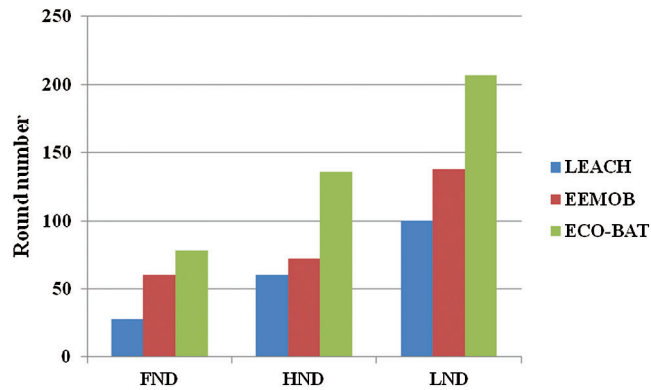


Figure 2: FND, HND and LND comparison over a round number

Fig. 3 is another comparison chart between the same protocols. The abscissa is the number of iterations, and the ordinate is the residual energy.

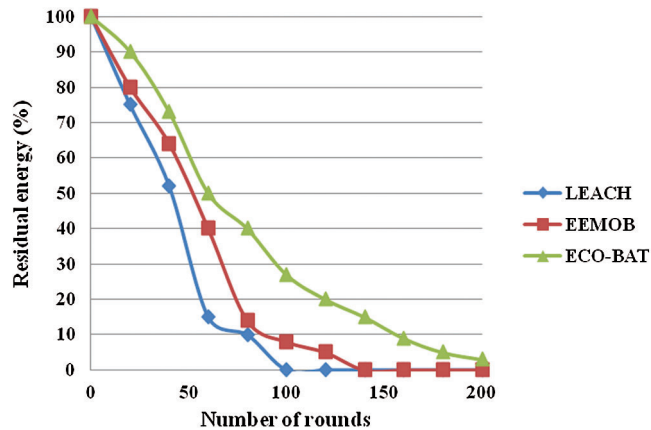


Figure 3: Residual energy over a round number

From Fig. 3, we observe that the residual energy of ECO-BAT decreases regularly compared to LEACH and in a way close to the EEMOB protocol. It effectively comes down to selecting the best combination of CHs with maximum residual energy and minimum intra-cluster distance due to the use of an adapted version of the bat algorithm with an objective fitness function.

Fig. 4 shows the data received by BS in the three protocols versus the number of iterations.

From the results of the simulation shown in Fig. 4, we can see that the proposed ECO-BAT protocol surpassed the LEACH and EEMOB protocol and offers a stable and efficient data transmission rate to BS.

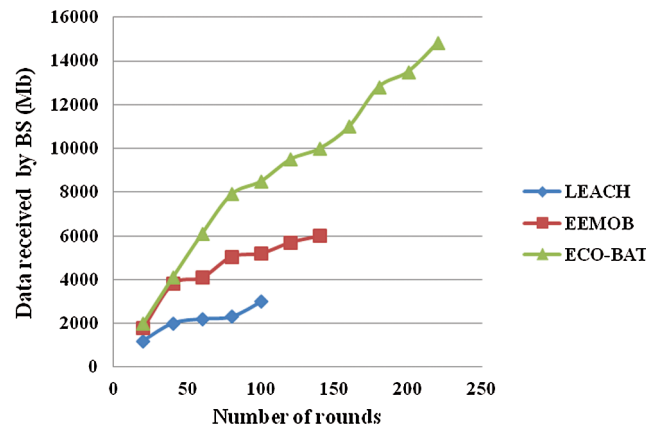


Figure 4: Data received by BS vs. N^o. of rounds

7 Conclusion and Perspectives

In this work, a clustering protocol has been proposed, which optimizes the selection of CHs. The algorithm implemented to take into account the remaining energy of the CH candidates and the maximum distance between the CH and the members of the cluster. It helps in the optimal selection of CHs. As the results of the simulation prove, our contribution ECO-BAT can effectively reduce the energy consumption of the network which gives a longer network lifetime compared to both protocols LEACH and EEMOB.

As perspectives for our future work, we think to direct our research towards the integration of mobility of the nodes in the network. Also, we think to add other parameters in the objective fitness function such as packet loss rate to improve the performance of ECO-BAT protocol and to assign for the constant alpha (α) different values for each test to see the percentage of the intervention of each sub-objectives f_1 and f_2 . We also think to combine the bat algorithm with other heuristic methods to get more optimal results for CH selection as well as cluster formation.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] V. Manikandan, M. Sivaram, A. S. Mohammed and V. Porkodi, "Nature inspired improved firefly algorithm for node clustering in WSNs," *Computers, Materials & Continua*, vol. 64, no. 2, pp. 753–776, 2020.
- [2] X. S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization*, J. R. Gonzalez *et al.* (Eds.), Vol. 284. Berlin, Heidelberg: Springer, pp. 65–74, 2010.
- [3] S. Induja and V. P. Eswaramurthy, "Bat algorithm: An overview and its applications," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 5, no. 1, pp. 448–451, 2016.
- [4] K. Khan, A. Nikov and A. Sahai, "A fuzzy bat clustering method for ergonomic screening of office workplaces," in *Advances in Intelligent and Soft Computing, S3T*, vol. 101. Berlin, Heidelberg: Springer-Verlag, pp. 59–66, 2011.
- [5] X. S. Yang, "Bat algorithm for multi-objective optimization," *International Journal of Bio-Inspired Computation*, vol. 3, no. 5, pp. 267–274, 2011.
- [6] R. Y. M. Nakamura, L. A. M. Pereira, K. A. Costa, D. Rodrigues, J. P. Papa *et al.*, "BBA: A binary bat algorithm for feature selection," in *Graphics, Patterns and Images. 25th Int. Conf., SIBGRAPI 2012*, Ouro Preto, Brazil, pp. 291–297, 2012.

- [7] S. Mirjalili, S. M. Mirjalili and X. Yang, "Binary bat algorithm," *Neural Computing and Applications*, vol. 25, no. 3–4, pp. 663–681, 2014.
- [8] P. W. Tsai, J. S. Pan, B. Y. Liao, M. J. Tsai and V. Istanda, "Bat algorithm inspired algorithm for solving numerical optimization problems," *Applied Mechanics and Materials*, vol. 148, pp. 134–137, 2012.
- [9] Y. Selim and U. K. Ecir, "Improved bat algorithm (IBA) on continuous optimization problems," *Lecture Notes on Software Engineering*, vol. 1, no. 3, pp. 279–283, 2013.
- [10] W. Heinzelman, A. Chandrakasan and H. Balakrishnan, "An application-specific protocol architecture for wireless micro sensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [11] V. Krishnaveni and S. Varadhaganapathy, "Energy efficient cluster formation in wireless sensor networks based on multi objective bat algorithm," *International Journal of Advanced Research in Science and Engineering*, vol. 5, no. 12, pp. 1–9, 2016.
- [12] M. Sharawi, E. Emary, I. A. Saroit and H. El-Mahdy, "Bat swarm algorithm for wireless sensor networks lifetime optimization," *International Journal of Science and Research*, vol. 3, no. 5, pp. 654–664, 2014.
- [13] M. S. Manshahia, M. Dave and S. B. Singh, "Improved bat algorithm based energy efficient congestion control scheme for wireless sensor networks," *Wireless Sensor Network*, vol. 8, no. 11, pp. 229–241, 2016.
- [14] N. V. S. N. Sarma and M. Gopi, "Energy efficient clustering using jumper firefly algorithm in wireless sensor networks," *International Journal of Engineering Trends and Technology*, vol. 10, no. 11, pp. 525–532, 2014.
- [15] K. Seelam, M. Sailaja and T. Madhu, "An efficient hybrid bat-optimized clustering for wireless sensor networks," *International Journal of Electronics, Communication & Instrumentation Engineering Research and Development*, vol. 5, no. 3, pp. 2249–7951, 2015.
- [16] Q. Tang, K. Yang, P. Li, J. Zhang, Y. Luo *et al.*, "An energy efficient MCDS construction algorithm for wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, pp. 102, 2012.
- [17] J. Zhang, D. F. Zhang, K. Xie, H. Qiao and S. M. He, "A VMIMO-based cooperative routing algorithm for maximizing network lifetime," *China Communications*, vol. 14, no. 4, pp. 20–34, 2017.
- [18] W. W. Chen, C. T. Lea, S. M. He and X. Y. Zhe, "Opportunistic routing and scheduling for wireless networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 1, pp. 320–331, 2016.