International Journal of Reconfigurable and Embedded Systems (IJRES)
Vol. 12, No. 3, November 2023, pp. 478~487
ISSN: 2089-4864, DOI: 10.11591/ijres.v12.i3.pp478-487

Design of ant lion optimization-based PEGASIS routing protocol for energy efficiency in networks

Kandrakunta Chinnaiah, Kunjam Nageswara Rao
Department of Computer Science and Systems Engineering, Andhra University College of Engineering (AUCE), Visakhapatnam, India

Article Info

Article history:

Received Jan 16, 2023
Revised Mar 19, 2023
Accepted Apr 17, 2023

Keywords:

Ant lion optimization
Elite antlion
PEGASIS
Residual energy
Wireless sensor networks

ABSTRACT

In sensor networks the main problem facing by many researchers is regarding the energy efficiency. Different protocols are evaluated for communicating between the sensor nodes. The routing protocol when combined with evolutionary algorithms gives best optimal solution for the problem incurred in wireless sensor networks. In this paper, ant lion energy efficient-power efficient gathering in sensor information systems (ALEE-PEGASIS) is used to develop the chain. This technique can achieve a global optimization solution by finding the best cluster head or the leader node for data communication. The techniques help in distributing the paths equally while the transmission of data process is performed. By performing this process, the power consumption near the sensor nodes can be reduced. The proposed technique is compared with other techniques like energy efficient PEGASIS and swarm energy efficient PEGASIS. The parameters used to compare are number of alive nodes, number of dead nodes and residual energy. The performance is observed using MATLAB simulation results.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Kandrakunta Chinnaiah
Department of Computer Science and Systems Engineering
Andhra University College of Engineering (AUCE)
Visakhapatnam-530003, AndhraPradesh, India
Email: chinna.nitc@gmail.com

1. INTRODUCTION

Improvements in wireless communication and digital signal processing have recently led to the creation of the wireless sensor network (WSN) [1]. WSN has been used in a variety of disciplines, including military investigation, medical therapy, environmental monitoring, and industry management. It does, however, have significant limitations, such as a restricted energy source and limited calculation and communication capabilities. As a result, how to extend network lifetime is a crucial and complex topic, which is also the subject of the WSN routing protocol design.

Several routing methods for WSN have been suggested. They are classed as plane protocols or hierarchical protocols based on the network structure. The hierarchical protocol intends to cluster sensor nodes so that cluster leaders may perform data gathering and reduction to conserve energy [2]. The core concept of hierarchical protocol is followed by power efficient gathering in sensor information systems (PEGASIS). All the nodes in the sensor network are grouped together to form a chain by following the greedy approach. The base station (BS) does everything. Beginning from the closest node, it selects the nearest neighbour of each node as the next node on the chain. Once the chain is complete, one node will be picked as the leader in each round of communication. Along this chain, each node merges the received data with its own before transmitting it to the next neighbour. The process is continued till the leader node receives the data. The complete fused data is then transmitted to the BS by the leader [3].

Because of advancements in the WSN innovation field, it has become necessary to investigate new methods or techniques, such as improving routing protocols by utilising many intelligent systems and optimization algorithms, in order to stay abreast of developments that have a positive impact on WSN technologies. These routing techniques are used on small and inexpensive sensor nodes to provide effective communication throughout the whole network. The sensor node architecture is depicted in Figure 1. These sensor nodes are extremely energy sensitive, resulting in restricted energy supply and, as a result, a short network lifetime. To address this issue, we must employ efficient routing algorithms that provide efficient and reliable communication between these nodes.

Several clustering strategies for WSN are proposed to increase energy efficiency, throughput, and network longevity [4]–[6]. Pooja and Singh [7] author proposed one of the prominent cluster-based routing protocol i.e. low-energy adaptive clustering hierarchy (LEACH). Depending on the residual energy at node and the distance between the nodes, the thresholding-based clustering technique need to be selected [8]. This cluster method improves the average residual energy and first dead node time when compared with existing LEACH technique. However, by adding extra parameters while clustering the network, performance may be greatly enhanced. Yao *et al*. [9], an evolutionary algorithm is utilized for sensor network clustering and routing based on each node's residual energy and the distance between sensor nodes and their CH. The utilization of different meta-heuristic techniques in WSN for grouping of nodes and CH selection is proposed by Wang *et al*. [10]. Rejinaparvin and Vasanthanayaki [11], author proposed a particle swarm optimization technique and is termed as enhanced-optimized energy efficient routing protocol (EOEERP). In 2002, PEGASIS [12] proposed a number of other enhanced protocols based on it, such as energy-efficient PEGASIS-based (EEPB) [13], improved energy efficient PEGASIS-based protocol (IEEPB) [14], protocol with double cluster head (PDCH) [15], PEGASIS performance based contract administration (PBCA) [16], PEGASIS-intersection based coverage algorithm (IBCA) [17], multi hops (MH)-PEGASIS [18], and modified PEGASIS [19]. In addition, some heuristic algorithms are used to improve the performance of the network [14], [20]–[22]. To enhance the efficiency optimization techniques introduced in PEGASIS such as ant–PEG [23]. Now further to enhance the energy of the system in this paper ant lion optimization (ALO) in PEGASIS is proposed.

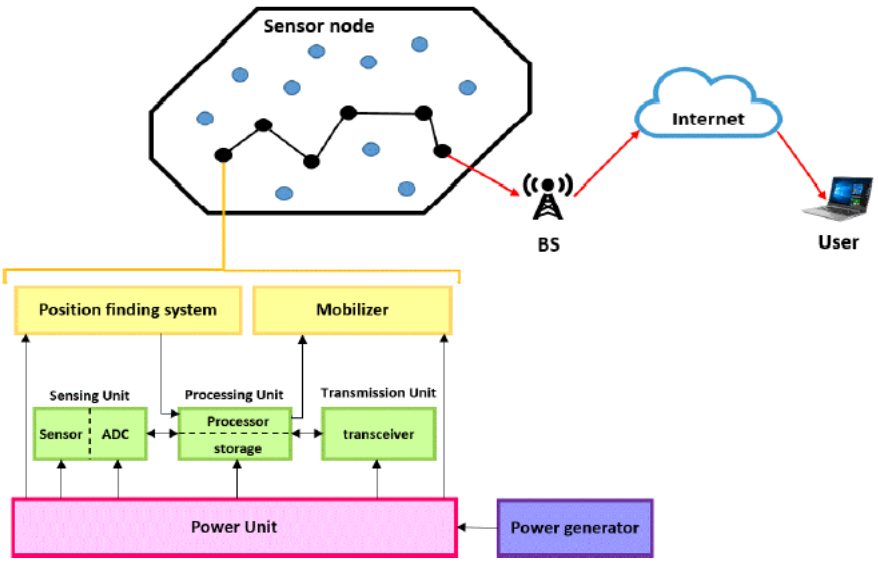

Figure 1. Architecture of sensor node

2. PEGASIS PROTOCOL

The PEGASIS protocol is a chain-based protocol which employs a greedy algorithm. In this method the sensor node creates a chain for transmission of data. For the purpose of routing the sensors nodes in the network together form a chain. The process of formation of chain keeps the chain alive by rebuilding the nodes in place of dead nodes. The transmission of data to the base station is handled by the leader node which is allocated in the network. The principle of PEGASIS is to send and receive data from the neighbour nodes and cluster head is used for sending data to base station [13]. The transmission process of PEGASIS is shown in Figure 2.

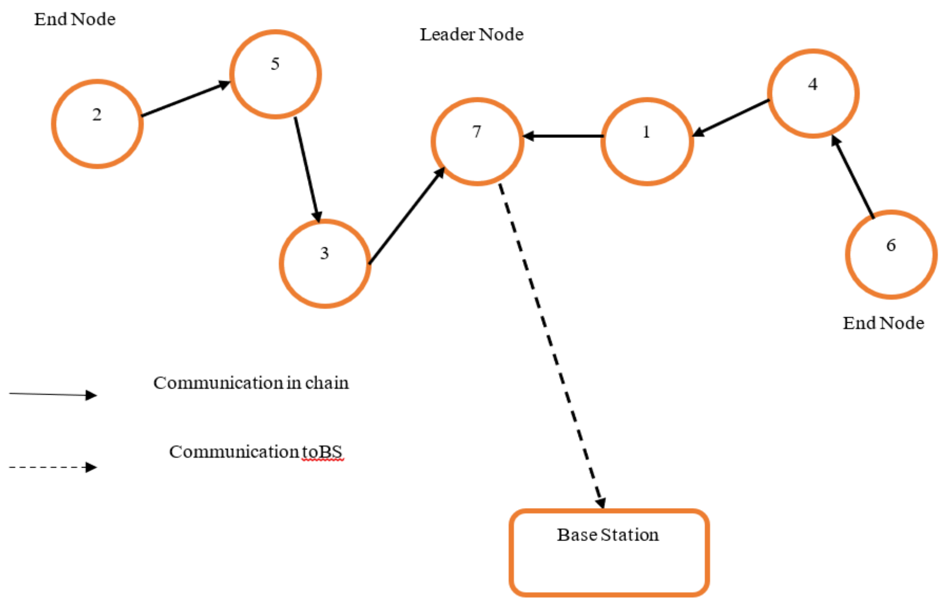

Figure 2. Transmission using PEGASIS

3. PROPOSED ALEE-PEGASIS

We propose a routing protocol ALEE-PEG using an improved energy efficient ALO algorithm for selecting the cluster head and forming the cluster by taking energy constraint into consideration for transmitting the data along the chain of nodes developed using PEGASIS protocol. The flow of proposed methodology is shown in Figure 3. In proposed methodology, arranging required number of sensors and choosing cluster head selection with the help of ALO and forming the cluster. By using that cluster, energy consumption is obtained at every node. If it is achieved minimum energy dissipation, then the process will be stopped.

3.1. Initiating the process

Firstly, need to consider a base station with maximum amount of energy and nodes which are of equivalent energy. The positions and energy of each node need to be identified and the distances between the nodes need to be calculated. Now the optimization technique will be initiated and the detailed description of Ant Lion is said in next session.

3.2. Modified ant lion optimization

The ALO programme simulates the interaction of ant-lions and ants in the trap [24]. Ants are needed to wander around the search space to replicate such interactions, and ant-lions are permitted to hunt them and get fitter using traps. Because ants travel stochastically in nature when looking for food, a random walk is used to represent their movement as (1).

$$X(t) = [0, CS(2r(t_1) - 1), CS(2r(t_2) - 1), \ldots, CS(2r(t_n) - 1] \tag{1}$$

Where the CS is termed as cumulative sum, maximum number of iterations is termed as ‘n’, ants random walk is termed as ‘t’. The ant’s fitness function needs to be evaluated at every iteration. The nodes are selected based on the ants that are arriving towards the ant-lions trap. As shown in (1) shows the random walks of the ants. Every step of optimization is performed based on the position updated with respect to the random walk of ants. The boundaries and limits are specified. Hence the normalization needs to done. The normalized equation which is applied near iteration to justify the random walk within the boundary is given (2).

$$X_i^t = \frac{(X_i^t - a_i) \times (d_i - c_i^t)}{(d_i^t - a_i)} + C_i \tag{2}$$

Thus, calculate the random walk of every ant and updating the position. The trap of ants is performed by antlion by siding the ants towards the position of antlions. Now the antlion relocates the position of best ants during the process of optimization, so that the search space will be saved properly. The movement intensity of ants will be decrease with the increase in number of iterations.

The search space location which is created is utilized by the antlion to guide the ants to the region. Each iteration's finest ant-lion will be kept and compared. The best solution is given by (3),

$$Ant_i^t = \frac{R_A^t + R_E^t + R_T^t}{2} \tag{3}$$

where R_A^t is the random walk of antlion selected using selection mechanism at t^{th}iteration, R_E^t random walk around the final stage at t^{th} iteration, the position of i^{th} ant at t^{th}iteration is Ant_i^t and R_T^t is the random walk of antlion selected using tournament mechanism at t^{th} iteration.

Hence the same process is applied in finding the best node to make it as a cluster head for transmission of data. This process of antlion optimization helps in forming the best chain of clusters and check for achieving minimum energy dissipation. During each round of communication, the node with the highest current energy is chosen as the cluster head. Starting from the end nodes, each node along the created chain and in the direction of the cluster head combines the received data (if any) with its own as one packet to broadcast to the other partner.

The cluster head is formed and the fitness evaluation of the identified cluster head is shown in (4) [25].

$$Fit_i = \beta_1.\left(\sum dis_{ij}\right) + \beta_2.E_{res}(i) + \beta_3.n_{neighbour}(i) + \beta_4.Eud_{i,BS} \tag{4}$$

Here $\beta_1, \beta_2, \beta_3$ *and* β_4 are constants between [0,1]$\sum dis_{ij}$ is the cumulative distance between node i and its neighbours. $n_{neighbour}(i)$ is the neighbour nodes for node i and the Euclidian distance between node i and base station is given by $Eud_{i,BS}$. The fitness function is calculated at every point of iteration for each ant and antlion. Algorithm 1 gives fitness function of ant lion and global best value.

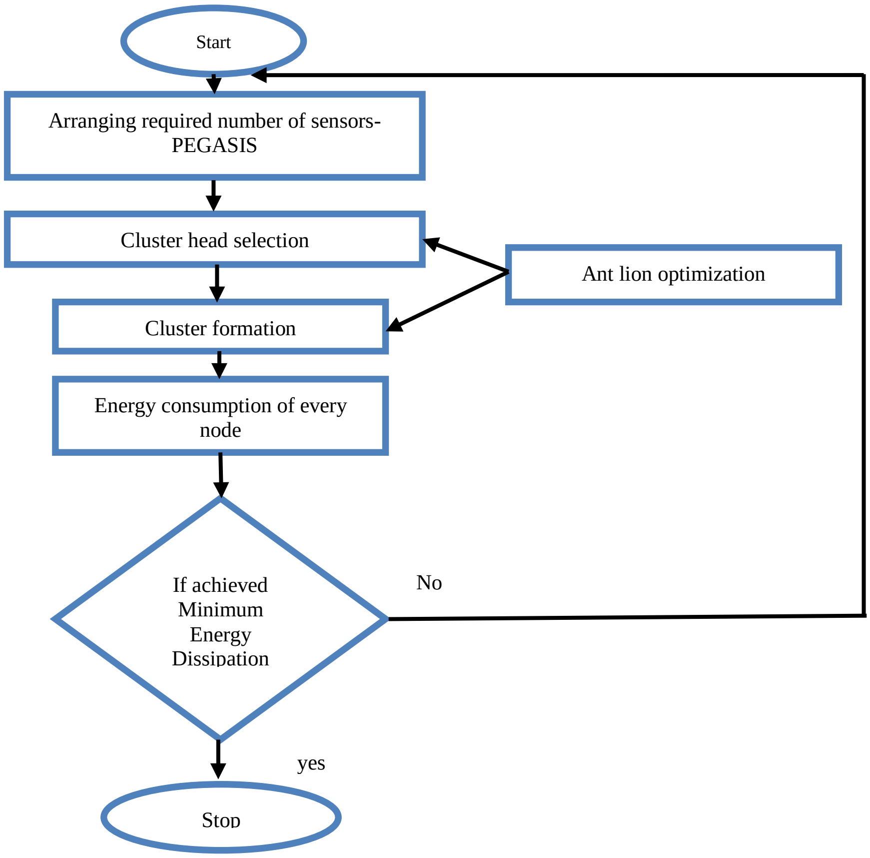

Figure 3. Flow diagram of proposed methodology

Algorithm 1. Ant lion optimization algorithm

```
Step 1: Initialize number of nodes, number of search agents, no of iterations.
Step 2: Initialization of the fitness function using (4).
Step 3: Initialization of ants positions and Antlions position.
Step 4: The sensor nodes are spread randomly with in the area utilized and BS is placed at
        (0,0)co-ordinate.
Step 5: The node distance is calculated among the nodes and its distance to BS.
Step 6: Evaluating the fitness function of antlion and obtain the best one.
Step 7: While (Current Iteration<No. of Generation)
               for each ant in the population
                    The Position is updated towards the Elite Antlion and a randomly
                    selected Antlion using (1) and (3)
                    min-max normalization is used to keep the ants in random walks inside
                    the search space using (4)
                    Calculate the fitness value of each ant using (4) and sort them
               End
            Merge the populations of ants and antlions and sort them
            Create new Antlion population based on the above sorted fitness value
            Select the best Antlion and mark it as Elite based on its fitness value
         End while
Step 8: Optimal value is identified by the global best value.
Step 9: The sensor node with the highest value obtained using (4) is designated as CH, and
its neighbours are joined as cluster members to create a cluster, and the process is
repeated with the other sensor nodes.
Step 10: Ending the process
```

4. RESULTS AND DISCUSSION

The simulation results are obtained using MATLAB tool. The environment consists of 100 nodes with a network size of 100×100 meters. The comparison is done between energy efficient PEGASIS and swarm energy efficient PEGASIS (SSE-PEG). By applying modified swarm optimization technique, the network life time is increased and is shown in below simulation results. The parameters considered for performing simulation operation is shown in Table 1. For the parameters shown in Table 1, the obtained simulation results are shown in Figures 4 and 5.

Table 1. Parameters set for simulation

Parameter	Value
Size of the network	100×100 meters
Nodes in network	100 numbers
Starting energy of node	0.5 J
Total energy for transmitting and receiving of data	50 J
Distance between the available nodes	50 m
Data considered for transferring	2000 bits

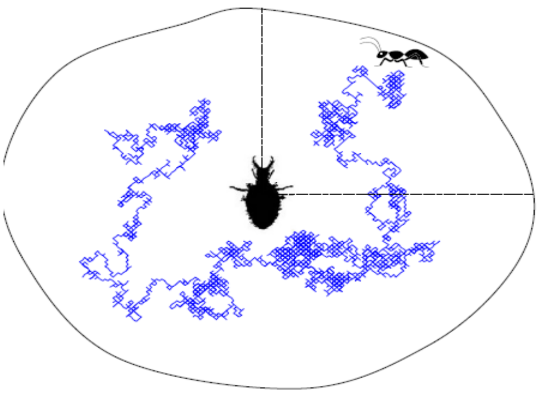

Figure 4. Random walk of an Ant inside the Antlion trap region

4.1. Case α

The performance output shown in Figure 5 is considered for 100 number of nodes. The starting energy of node is 0.5 J and the total energy for transmitting and receiving of data is 50 J. The distance between available nodes is 50 m and data considered for transferring is 2,000 bits. By these results network lifetime is increased.

From Figure 6 it is shown that proposed swarm optimization as good level of energy compared to EE-PEGASIS. If we consider 2,000 rounds for comparison the energy remained using ALEE-PEG is 38 J, modified SEE-PEGASIS is around 10 J, whereas energy remained using EE-PEGASIS is 4 J. The energy becomes zero when 3,500 rounds are done for EE-PEGASIS, whereas the energy remained upto 5,000 rounds using modified SEE-PEGASIS, using ALEE-PEG the energy remained upto 6,500 number of rounds.

From Figure 7 it is shown that using ALO alive nodes at 5,000 rounds is 12 nodes, whereas in swarm optimization the number of alive nodes is 4 nodes and EE-PEGASIS approach the number of alive nodes are 0 nodes. This shows the nodes death rate is slow in proposed techniques compared to the existing technique. From Figure 8 it is shown that all the nodes are dead at around 4,600 nodes using EE-PEGASIS, modified SEE-PEGASIS the nodes are dead around 5,500 rounds, whereas using ALEE-PEG the nodes complete zero after 8,000 rounds. The number of rounds is increased using proposed technique by which the data transmitting rate will be increased.

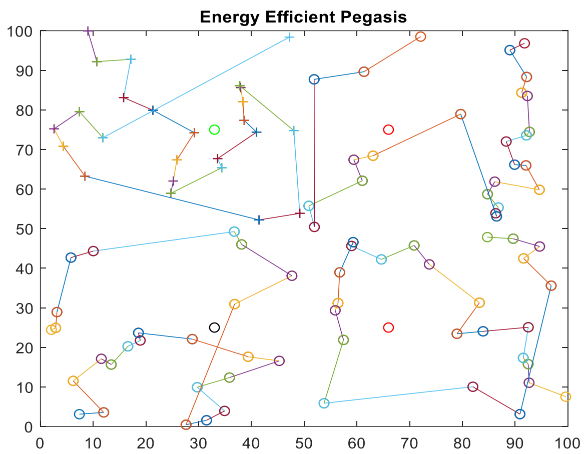

Figure 5. Sensor node distribution

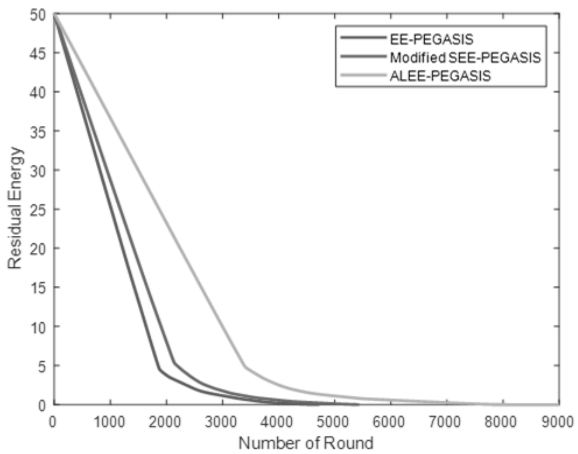

Figure 6. Residual energy (J) vs no of rounds

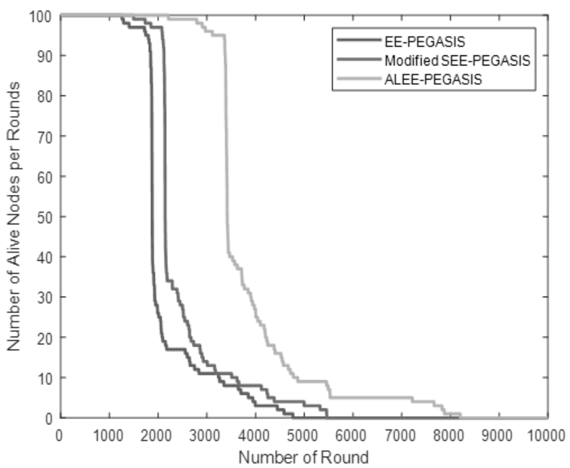

Figure 7. No of alive nodes vs no of rounds

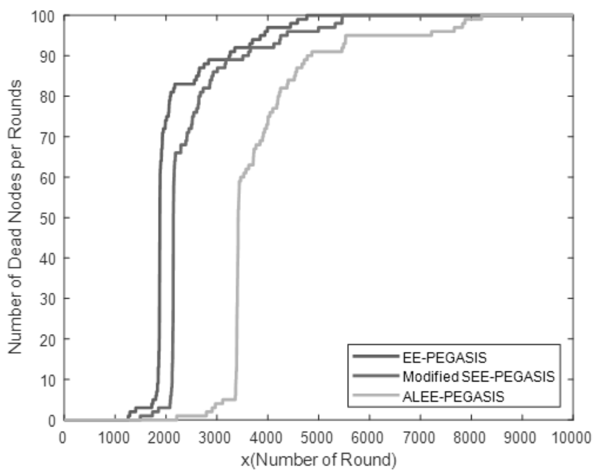

Figure 8. Number of dead nodes vs number of rounds

4.2. Case 2

The performance results obtained using 200 number of nodes are shown below. From Figure 9 it is shown that using ALO alive nodes are 60 nodes after reaching 5,000 rounds whereas in swarm optimization the number of alive nodes is 20 nodes and EE-PEGASIS approach the number of alive nodes are 10 nodes. These shows the alive nodes are more active in proposed techniques compared to the existing technique.

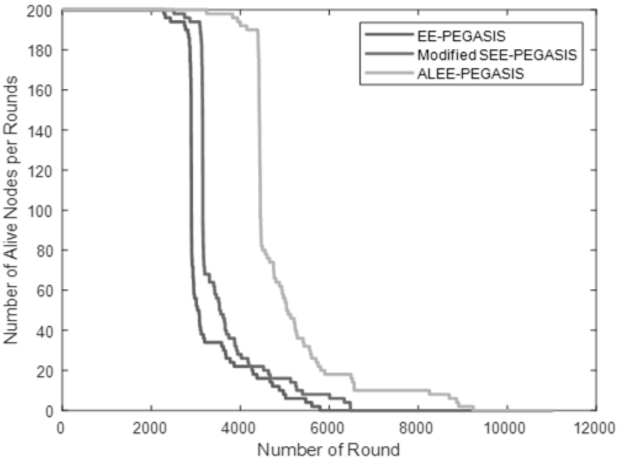

Figure 9. Number of alive nodes vs number of rounds

From Figure 10 it is shown that the nodes are completely dead after 9,200 rounds using ALEE-PEG techniques, whereas using modified swarm optimization the nodes are completely dead after 6,200 round and using EE-PEGASIS the nodes are completely dead after reaching 5,000 rounds. From Figure 11, if we consider 5,000 rounds for comparison the energy remained using ALEE-PEG is 28 J, modified SEE-PEGASIS is around 10 J, whereas energy remained using EE-PEGASIS is 3 J.

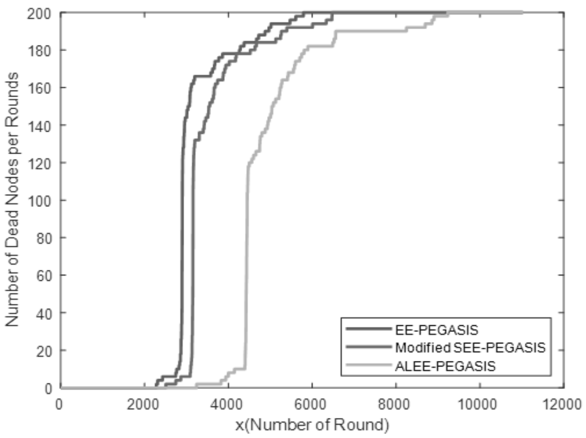

Figure 10. Number of dead nodes vs number of rounds

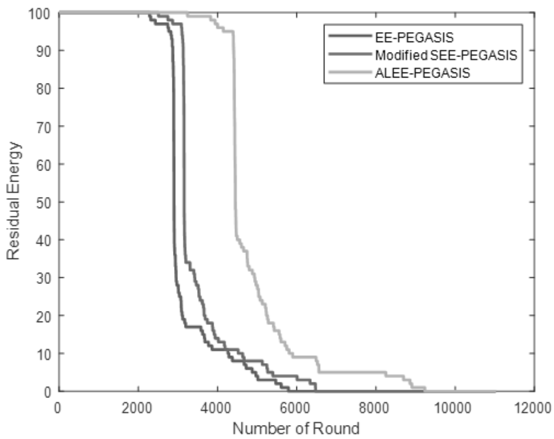

Figure 11. Residual energy vs number of rounds

4.3. Case 3

Considering 500 nodes the results obtained are shown in Figures 12-14. In Table 2 the comparison of results are shown. The parameters shown are residual energy, number of dead nodes and number of alive nodes. The values are tabulated by considering 2,500 number of rounds for 100 nodes.

From Table 2 it is clear that proposed ALEE-PEGASIS retains its energy level for a greater number of rounds and also the number of alive nodes is higher compared to existing method. The ALEE-PEGASIS is having residual energy of 18 J which is more when compared to residual energy of SEE-PEGASIS and EE-PEGASIS. The throughput percentage of ALEE-PEGASIS is better when compared to SEE-PEGASIS and EE-PEGASIS which is 89.7%.

The Table 3 values are tabulated by considering 5,000 number of rounds for 200 nodes. The number of alive nodes of ALEE-PEGASIS is more which is 54 when compared to SEE-PEGASIS and EE-PEGASIS. The number of dead nodes of ALEE-PEGASIS is 146 which is less when compared to SEE-PEGASIS and EE-PEGASIS. The delay time of ALEE-PEGASIS is 12.71 sec which is less when compared to SEE-PEGASIS and EE-PEGASIS.

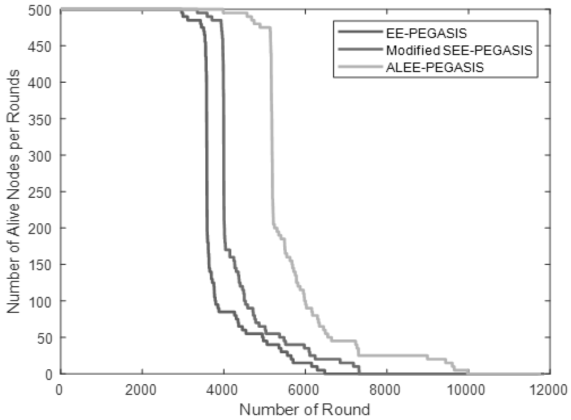

Figure 12. Number of alive nodes vs number of nodes

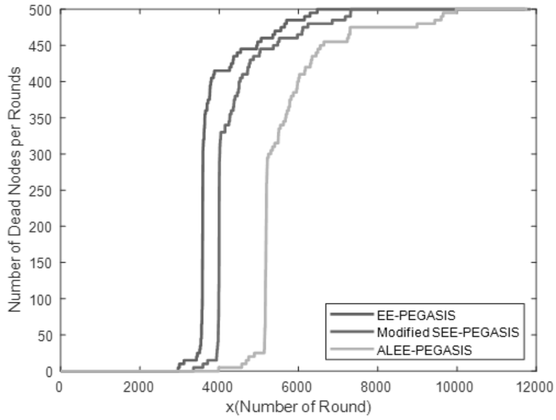

Figure 13. Number of dead nodes vs number of rounds

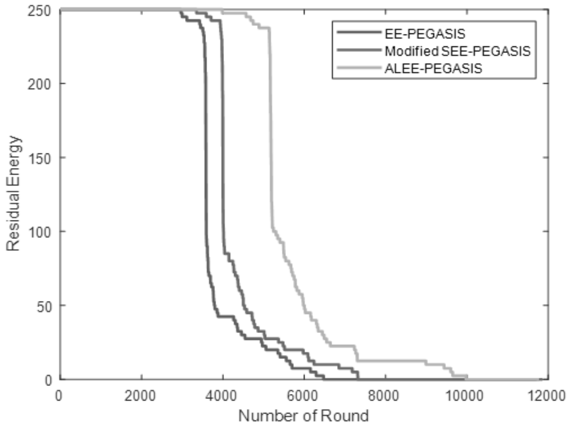

Figure 14. Residual energy vs number of nodes

Table 2. Results comparison for 100 nodes

Parameters	EE-PEGASIS	SEE-PEGASIS	ALEE-PEGASIS
Residual energy	3 J	9 J	18 J
Number of alive nodes	18	30	96
Number of dead nodes	82	70	04
Throughput (%)	82.96	86.3	89.7
Packet delivery ratio (%)	84.79	88.4	91.93
Delay (Sec)	18.87	17.11	13.75

Table 3. Results comparison for 200 nodes

Parameters	EE-PEGASIS	SEE-PEGASIS	ALEE-PEGASIS
Residual energy	4 J	10 J	27 J
Number of alive nodes	10	18	54
Number of dead nodes	190	182	146
Throughput (%)	82.87	86.49	89.84
Packet delivery ratio (%)	84.41	88.69	91.76
Delay (Sec)	17.39	16.44	12.71

The Table 4 values are tabulated by considering 6,000 number of rounds for 500 nodes. The packet delivery ratio of ALEE-PEGASIS is 91.81% which is more when compared to SEE-PEGASIS and EE-PEGASIS. The throughput percentage of ALEE-PEGASIS is 89.82% which is more when compared to SEE-PEGASIS and EE-PEGASIS. The delay time of ALEE-PEGASIS is 11.69 sec which is less when compared to SEE-PEGASIS and EE-PEGASIS.

Table 4. Results comparison for 500 nodes

Parameters	EE-PEGASIS	SEE-PEGASIS	ALEE-PEGASIS
Residual energy	10 J	25 J	65 J
Number of alive nodes	20	45	90
Number of dead nodes	480	455	410
Throughput (%)	82.89	86.65	89.82
Packet delivery ratio (%)	84.97	88.56	91.81
Delay (Sec)	16.93	15.07	11.69

From Table 5, the proposed improved ALO techniques provides high energy efficiency as the number of node present till 8,201 rounds performed by the wireless communication system while transmission of data. The first dead nodes, half dead nodes and last dead nodes at 100, 200 and 500 are shown in Table 5 by using ALEE-PEGASIS, SEE-PEGASIS and EE-PEGASIS. The last dead nodes of ALEE-PEGASIS is 9,976 which is more when compared to SEE-PEGASIS and EE-PEGASIS at 500 rounds. The half dead nodes of ALEE-PEGASIS is 5,194 which is more when compared to SEE-PEGASIS and EE-PEGASIS at 500 rounds. The first dead nodes of ALEE-PEGASIS is 3,986 which is more when compared to SEE-PEGASIS and EE-PEGASIS at 500 rounds. Similarly, ALEE-PEGASIS at 100 and 200 rounds for first dead nodes is 2,795 and 3,243, half dead nodes is 3,419 and 4,451 and for last dead nodes is 8,201 and 9,233 which are more compared to SEE-PEGASIS and EE-PEGASIS.

Table 5. The nodes status based on rounds

Parameters	EE-PEGASIS			SEE-PEGASIS			ALEE-PEGASIS		
	100	200	500	100	200	500	100	200	500
First dead nodes	1,490	2,283	2,966	1,735	2,509	3,352	2,795	3,243	3,986
Half dead nodes	2,079	2,903	3,586	2,345	3,158	4,401	3,419	4,451	5,194
Last dead nodes	4,767	5,791	6,474	5,462	6,475	7,318	8,201	9,233	9,976

5. CONCLUSION

PEGASIS protocol is one of the best protocols which is introduced in wireless sensor networks. Improvements have risen through this mechanism and model. Further optimization techniques are combining with PEGASIS protocol to enhance the network efficiency. In this paper, to build the PEGASIS protocol chain, we provided a routing protocol that used an enhanced ALO. We present our protocol's concept in detail, and show how our protocol outperforms EE-PEGASIS and SEE-PEG using a simulated experiment in MATLAB. In order to enhance the network life time new metaheuristic algorithm is designed. The results obtained using Improved ALEE-PEG is better compared to other techniques in minimizing the energy consumption of the network.

REFERENCES

[1] N. Primeau, R. Falcon, R. Abielmona, and E. M. Petriu, "A review of computational intelligence techniques in wireless sensor and actuator networks," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 4, pp. 2822–2854, 2018, doi: 10.1109/COMST.2018.2850220.

[2] K. Dasgupta, K. Kalpakis, and P. Namjoshi, "An efficient clustering-based heuristic for data gathering and aggregation in sensor networks," in *IEEE Wireless Communications and Networking Conference, WCNC*, 2003, vol. 3, pp. 1948–1953. doi: 10.1109/WCNC.2003.1200685.

[3] S. Lindsey and C. S. Raghavendra, "PEGASIS: power-efficient gathering in sensor information systems," in *IEEE Aerospace Conference Proceedings*, 2002, vol. 3, pp. 1125–1130. doi: 10.1109/AERO.2002.1035242.

[4] Y. Gong and G. Lai, "Low-energy clustering protocol for query-based wireless sensor networks," *IEEE Sensors Journal*, vol. 22, no. 9, pp. 9135–9145, May 2022, doi: 10.1109/JSEN.2022.3159546.

[5] T. Wang, X. Guan, X. Wan, H. Shen, and X. Zhu, "A spectrum-aware clustering algorithm based on weighted clustering metric in cognitive radio sensor networks," *IEEE Access*, vol. 7, pp. 109555–109565, 2019, doi: 10.1109/ACCESS.2019.2929574.

[6] L. Chen, D. Yang, Z. Xu, and C. Chen, "An adaptive routing strategy for cluster-based wireless sensor networks," in *Proceedings of the 2015 27th Chinese Control and Decision Conference, CCDC 2015*, May 2015, pp. 5224–5229. doi: 10.1109/CCDC.2015.7162856.

[7] P. Pooja and S. Singh, "Improved O-LEACH protocol: a clustering based approach in wireless microsensor network," in *Proceedings of the 10th International Conference on Intelligent Systems and Control, ISCO 2016*, Jan. 2016, pp. 1–4. doi: 10.1109/ISCO.2016.7727015.

[8] A. Hossan and P. K. Choudhury, "DE-SEP: distance and energy aware stable election routing protocol for heterogeneous wireless sensor network," *IEEE Access*, vol. 10, pp. 55726–55738, 2022, doi: 10.1109/ACCESS.2022.3177190.

[9] Y. D. Yao, X. Li, Y. P. Cui, J. J. Wang, and C. Wang, "Energy-efficient routing protocol based on multi-threshold segmentation in wireless sensors networks for precision agriculture," *IEEE Sensors Journal*, vol. 22, no. 7, pp. 6216–6231, Apr. 2022, doi: 10.1109/JSEN.2022.3150770.

[10] Z. Wang, H. DIng, B. Li, L. Bao, and Z. Yang, "An energy efficient routing protocol based on improved artificial bee colony algorithm for wireless sensor networks," *IEEE Access*, vol. 8, pp. 133577–133596, 2020, doi: 10.1109/ACCESS.2020.3010313.

[11] J. Rejinaparvin and C. Vasanthanayaki, "Particle swarm optimization-based clustering by preventing residual nodes in wireless sensor networks," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4264–4274, Aug. 2015, doi: 10.1109/JSEN.2015.2416208.

[12] A. N. Wilson, A. Kumar, A. Jha, and L. R. Cenkeramaddi, "Embedded sensors, communication technologies, computing platforms and machine learning for UAVs: a review," *IEEE Sensors Journal*, vol. 22, no. 3, pp. 1807–1826, Feb. 2022, doi: 10.1109/JSEN.2021.3139124.

[13] K. Wang, C. M. Yu, M. L. Ku, L. C. Wang, and W. K. Jia, "Joint shortest chain and fair transmission design for energy-balanced PEGASIS in WSNs," *IEEE Internet of Things Journal*, vol. 10, no. 8, pp. 6803–6817, Apr. 2022, doi: 10.1109/JIOT.2022.3227599.

[14] M. Abo-Zahhad, S. M. Ahmed, N. Sabor, and S. Sasaki, "Mobile sink-based adaptive immune energy-efficient clustering protocol for improving the lifetime and stability period of wireless sensor networks," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4576–4586, Aug. 2015, doi: 10.1109/JSEN.2015.2424296.

[15] S. Sasirekha and S. Swamynathan, "Cluster-chain mobile agent routing algorithm for efficient data aggregation in wireless sensor network," *Journal of Communications and Networks*, vol. 19, no. 4, pp. 392–401, Aug. 2017, doi: 10.1109/JCN.2017.000063.

[16] Y. L. Chen, N. C. Wang, C. L. Chen, and Y. C. Lin, "A coverage algorithm to improve the performance of PEGASIS in wireless sensor networks," in *Proceedings - 2011 12th ACIS International Conference on Software Engineering, Artificial Intelligence Networking and Parallel Distributed Computing, SNPD 2011*, Jul. 2011, pp. 123–127. doi: 10.1109/SNPD.2011.11.

[17] H. Nakahara, A. Monden, and Z. Yucel, "A simulation model of software quality assurance in the software lifecycle," in *Proceedings - 22nd IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, SNPD 2021-Fall*, Nov. 2021, pp. 236–241. doi: 10.1109/SNPD51163.2021.9704927.

[18] Y. L. Chen, Y. C. Lin, and N. C. Wang, "An intersection-based coverage algorithm for PEGASIS architecture in wireless sensor networks," in *Proceedings - International Conference on Machine Learning and Cybernetics*, Jul. 2012, vol. 5, pp. 1727–1731. doi: 10.1109/ICMLC.2012.6359635.

[19] Z. Aliouat and M. Aliouat, "Efficient management of energy budget for PEGASIS routing protocol," in *2012 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, Mar. 2012, pp. 516–521. doi: 10.1109/SETIT.2012.6481966.

[20] X. Sun, P. Hu, Y. Li, H. Kang, Q. Chen, and Y. Shen, "An improved algorithm for multi-swarm particle swarm optimization based on clustering algorithm," in *2018 IEEE 4th International Conference on Computer and Communications, ICCC 2018*, Dec. 2018, pp. 2017–2021. doi: 10.1109/CompComm.2018.8780708.

[21] J. Wang, J. Cao, B. Li, S. Lee, and R. S. Sherratt, "Bio-inspired ant colony optimization based clustering algorithm with mobile sinks for applications in consumer home automation networks," *IEEE Transactions on Consumer Electronics*, vol. 61, no. 4, pp. 438–444, Nov. 2015, doi: 10.1109/TCE.2015.7389797.

[22] Y. Tang, N. Wang, J. Lin, and X. Liu, "Using improved glowworm swarm optimization algorithm for clustering analysis," in *Proceedings - 2019 18th International Symposium on Distributed Computing and Applications for Business Engineering and Science, DCABES 2019*, Nov. 2019, pp. 190–194. doi: 10.1109/DCABES48411.2019.00054.

[23] S. Chauhan and M. J. Nene, "Energy efficent protocols LEACH - GWO vs PEGASIS comparision," in *2022 International Conference on Futuristic Technologies, INCOFT 2022*, Nov. 2022, pp. 1–6. doi: 10.1109/INCOFT55651.2022.10094452.

[24] I. Strumberger *et al.*, "Feature selection by hybrid binary ant lion optimizer with COVID-19 dataset," in *2021 29th Telecommunications Forum, TELFOR 2021 - Proceedings*, Nov. 2021, pp. 1–4. doi: 10.1109/TELFOR52709.2021.9653256.

[25] H. Shen and Q. Liu, "An improved ant lion optimization algorithm and its application," in *ICNSC 2022 - Proceedings of 2022 IEEE International Conference on Networking, Sensing and Control: Autonomous Intelligent Systems*, Dec. 2022, pp. 1–6. doi: 10.1109/ICNSC55942.2022.10004110.

BIOGRAPHIES OF AUTHORS

Kandrakunta Chinnaiah worked as a full-time research scholar in Andhra University, 2015-2020. Presently working in Universal College of Engineering and Technology as Associate Professor in the Department of Computer Science and Engineering, Dokiparru, Perecharla, Medikonduru, Guntur, Andhra Pradesh, India. He completed his B.Tech. in Computer Science and Systems Engineering from GITAM Engineering College, M.Tech. in Computer Science and Technology with Bio-Informatics from Andhra University and MBA from VIPS, Vishakapatnam. His current research includes wireless sensor networks, bioinformatics, data science, cloud computing, artificial intelligence and machine learning, internet of things, and big data analytics. He can be contacted at email: chinna.nitc@gmail.com.

Dr. Kunjam Nageswara Rao working as Professor and Head in the Department of Computer Science and Systems Engineering, University College of Engineering, Andhra University, Visakhapatnam, Andhra Pradesh, India. He completed his B.Tech. in Computer Science and Systems Engineering from GITAM Engineering College, M.Tech. in Computer Science and Technology from Andhra University and Ph.D. in Computer Science Engineering from Andhra University. His current research includes bio-informatics, computer science information systems, health informatics, big data analytics, artificial intelligence and machine learning, algorithm, and design analysis. He can be contacted at email: kunjamnag@gmail.com.