Dragonfly Algorithm for Intensifying Double Chain Routing PEGASIS Protocol in Wireless Sensor Networks

Kondapalli Siva, PG Student, Dpt of Electronics and communication Engineering, Andhra University College of Engineering, Andhra University, Visakhapatnam.

P.V.Sridevi, Professor, Dpt of Electronics and communication Engineering, Andhra University College of Engineering, Andhra University, Visakhapatnam.

Abstract

Wireless sensor networks (WSNs) have become one of the most widely used techniques in various applications such as agriculture, factory surveillance, healthcare, and fire protection systems. This paper propose a novel approach to achieve better WSN improvements in terms of network robustness and data transmission time represented by reduced packet delay times. Dual Chain Dual Sink Node An energy-efficient PEGASIS protocol improved by Dragonfly optimization was designed and compared with the ant colony optimization algorithm. Dragonfly optimization technique is used to identify sensor nodes and cluster heads. The simulation work is mainly performed to generate sensor positions based on the Dragonfly algorithm with PEGASIS. Parameters such as active nodes, energy loss and network lifetime are estimated using Matlab, the results are compared and the proposed approach leads to better energy efficiency of the cluster head.

Keywords:

Wireless Sensor Networks, PEGASIS, Dragonfly algorithm, ACO

1. Introduction

A wireless sensor network consists of small and low-power sensor nodes distributed over a large geographic area, capable of detecting phenomena and transmitting data to receivers or base stations. Generally sensor nodes are responsible for their work, data transfer, computer knowledge and resources. Every divided sensor point of network is ability of aggregating and transmitting information to every sensor points or a base station. Typically, base stations could be motionless or mobile to provide data reporting to users via the Internet or communication support, as shown in Fig. 1[1]. Generally, WSNs are designed to decrease power consumption and increase the life time of network. Information transmitted from a sensor node to a destination usually consists of one or more hops. The largest energy consumption occurs during data transmission from node to the base station [2], [3]. Power saving has always been a major challenge in WSNs.

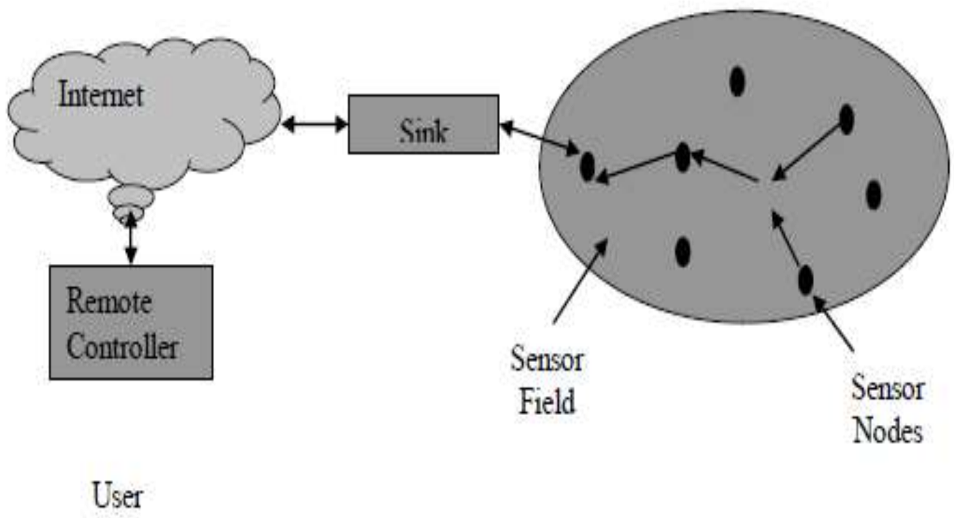


Fig1. Sensor Network Model

The network lifetime of a WSN based on the performance of the battery, this has usually a finite power source. To increase the lifetime of the network, sensor points must have another power sources. One way to increase the network lifetime of a sensor point is to adjust the transmit power of the sensor node during transmission. Setting the transmit power is not always sufficient to optimize power consumption and improve sensor battery performance.

Several protocols can also be used to reduce excessive power consumption when transmitting information to base stations. The routing protocol plays a major role as it reduces the delay by providing more energy efficiency and increase the network life time. One such type of protocol is a PEGASIS, introduced by Stephanie Lindsey [4]. This is based on the chain based structure; each chain has cluster header. The cluster head generally consumes a lot of power and the interval of each circular increment is responsible for sending and receiving messages from each entry belonging to that chain structure. In the PEGASIS, that takes advantage of nearest-neighbor data transmission, saving battery power for WSNs and increasing network lifetime. The paper structured as follows: Section 2 represents the work of this paper. Section 3 provides a quick description of the PEGASIS protocol along with routing optimization. Section 4 represents the performance assement of the proposed work. Eventually, Section 6 presents the conclusion.

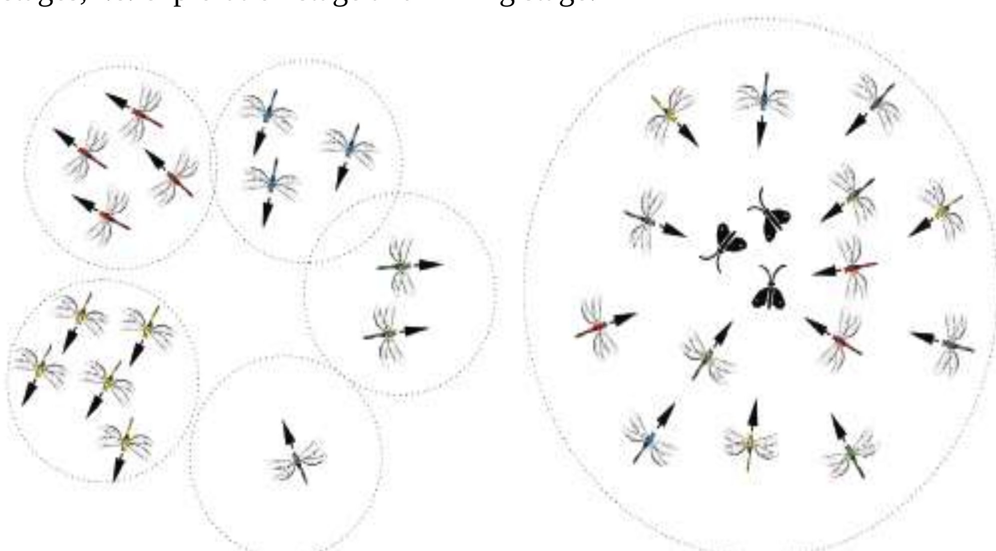
2. Related Work

In the PEGASIS protocol, instead of clusters, chains are built, starting from highest node of the base station. Then the relationship between every node and its closest neighbor sensor point forms a chain structure. The base station collects information from the chain head (CH), which receives information from other sensor points in the network. This procedure initiate a transmission delay as the data must travel around the node to reach the CH. In [5], the PEGA-ACO (PEGASIS using Ant Colony Optimization) protocol was used to reduce the transmission length and total chain length by merging cluster-based structure and chain-based architecture. This reduces network ldelay of concurrently running clusters. In addition, to achieve proper network load balancing, proper cluster head approach and use of smart routing methods for optimal global circuits at shortest distances. A biogeography-based optimization (BBO) is a technique it is proposed in [6] to construct a biogeography-based PEGASIS circuit. This includes the distribution of plants and animals in different environments. The circuit design takes into account that each sensor node creates an energy balance as it slides between the sensor nodes. PEGA-BBO improves energy efficiency and extends network life.

The Artificial Bee Colony algorithm (ABC) in [7] is designed as routing algorithm for optimizing energy levels of nodes. PEGA-ABC is useful to prevent node dead and data packet loss. Strings are created by grouping nodes with multiple cell sinks. This process provides load balancing between nodes that consume less energy for efficient packet transmission. A new method to improve the performance of the PEGASIS is given in [8] it selects two strings and partitioning the network into numerous data fields. Schematic of potential conductor selection depends on the node closest to sink. Generally, the actual PEGASIS used causal chain routing. So, it reduces the transmission length from the head chain to the base station. An iterative technique is also used to transfer data to the receiver.

3. Dragon Fly Optimization

The algorithm triggers the dragonfly swarm behavior. Two types of motion designs are involves during search process, in this one is static movement and another one is dynamic movement. The static movement type is taken into account when going into a little area to hunt insects. The static hunting procedure involves unpredictable moves and local moves. The dynamic movement type is considered a as long search, in this group the flies form groups and move in one direction for longer intervals [14]. Static movement and dynamic movement swarming are shown in Figure 2. This swarming process involves two stages, i.e. exploration stage and mining stage.



(a) (b)

Fig2. (a) Static Movement (b) Dynamic Movement

The performance of the kite ensemble algorithm based on various variables such as separation ‘s’, orientation ‘a’, cohesion ‘c’, inertial mass ‘w’ and food and enemy vectors ‘f’, ‘e’. This parameter assists the dragonfly to move along various paths. The exploration and utilization of these two phases depends on harmony and coherence. High-point alignment parameters and low-point cohesion variables are used to explore the search area and vice versa, i.e., low-point alignment parameters and high-point convergence variables are used to exploit the search area. The variables must be adjusted during the

optimization process to maintain fairness between the two stages. The stage transition i.e. from one stage to another based on the number of repetitions. The variables are mathematically formulated and shown in the equations from Equation 1 to Equation 5 below.
The separation is calculated by Reynolds as mentioned [15]:

$$SP_i = -\sum_{j=1}^{N} P - P_j \quad (1)$$

In the equation 1, the current location of the dragonfly given by P, The Position of j^{th} of the neighboring dragonflies is given byP_j, and the no of dragonflies in the given area is given as *N*, and SP is the separation movement. The following equation (2) [9] is used to calculate the alignment parameters::

$$AP_i = \frac{\sum_{j=1}^{N} Ve_j}{N} \quad (2)$$

Here, the alignment movement of i^{th} individual given as AP_i and velocity given as *Ve.*

The formula for calculating Cohesion given as,

$$CP_i = \frac{\sum_{j=1}^{N} P_j}{N} - P \quad (3)$$

Here, cohesion of i^{th} individual is represented by CP_i , size of closest is determined by *N*, Location of j^{th} fly P_j, present location of fly is given as *P*. The search for food and the attraction of food calculated by using the equation (4) below:

$$Fd_i = P^+ - P \quad (4)$$

Where, food attraction is Fd_i for i^{th} dragonfly, the food source position isP^+, the current location of fly given as P. In this algorithm the best objective purposes the food. The Distraction out way predators are determined by:

$$En_i = P^- + P \quad (5)$$

Here, Distraction of enemy is given by En_i to the i^{th} individual, the current location of enemy is represented by P^- and present location of individual fly is the P. To change the location in the search area two vectors used in that one is the step vector and another one is the position vector and it is denoted by using ΔP and *P*. In this step vector used over here is same as the vector in PSO [16]. The changing of location is same as PSO algorithm. In this the step vector is determined in [17] as follows:

$$\Delta P_{t+1} = (sSP_i + aAP_i + cCP_i + fFd_i + eEn_i) + w\Delta P_t \quad (6)$$

Where w is the weight of inertia and the *t* represents the iteration counter. After that the evaluation of position vector, step vectors are calculated by using the equation 8.

$$P_{t+1} = P_t + \Delta P_{t+1} \quad (7)$$

Here, current iteration is termed as *t*.

To increase the probability of exploring the whole sample space with improvement calculations, a random step should be added to the exploration process. When there is no coherent arrangement to construct the deviance, stochastic behavior, and investigation of false human dragonflies, the dragonflies must use random paths (let L'eva) to fly through the space under investigation.

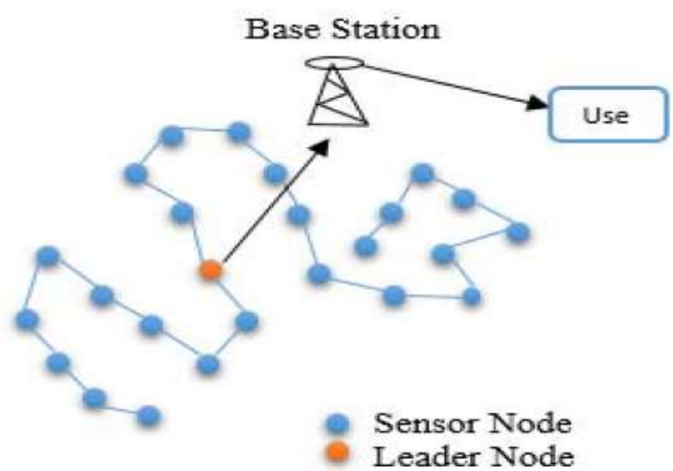
In the Dragonfly algorithm, the number of solutions is generated unsystematically at the beginning of the optimization procedure. A boundary is generated and the values for the step vector and its initial location between the bottom and higher boundaries are determined. The update of the location of each dragonfly and the step vector is done in each iteration phase using Equations 6 and the 7. The Euclidean length calculated to select the neighbors. The positions are updated till the best global solution is acquired.

4. Design of work

In this paper, the last changed routing protocol is the DDIEEPB-DFO, the following Figure 3 shows the phases of the working procedure in the flowchart.

A. Pegasis Protocol

The PEGASIS protocol is a cluster chain-based routing protocol in that a sensor point is closed to its nearest neighbor point and selects a master sensor point for information to be sent to the base station. The sensor point information is sended to the next nearest neighbor sensor point and this procedure continues through two sensor points up to hundred sensor points, with the leader sensor point among these selected sensor point transmitting the data to the base station node. In this paper, a PEGASIS routing protocol with an optimization algorithm is proposed



.

Fig3. Pegasis Architecture

B. Double chain Double sink node Routing

The design is dependent the base station and it broadcasts to the all the sensor points. Thus, all the sensor points will transmit information and receive data about neighboring nodes and choose the leader sensor point to transmit information to the base station. Sensor nodes form a chain for the inner link, and all member sensor points of the chain transmit data to the head of the chain using the concept of chain routing. This process involves two steps, namely the selection of the master sensor point of the chain and the transfer of data on the chain. All chain heads create cluster-based routing as links. Finally, the selected cluster head sends the aggregated data to the base station. If any node in the selected chain is dead and also if the cluster head is far from the base station, so more energy will be used to transmit data, so double chains are created to transmit data in case of any chain failure.

C. PEGASIS-DA

In this section details the proposed mechanism, which is depends on the power-saving dual-chain PEGASIS protocol and the named dragonfly algorithm (PEGASIS-DCDSDA). Nodes are spaced and provided the layout is in a circular region M*M area to cover the entire desired space. These nodes energetically same and the each node has own identifier. The base station and the all sensor points are fixed after deployment and their position is located by GPS.

When transmitting data in PEGA-DA, each sensor point selects a typical way to communicating with neighbor nodes. To avoid the consumption of a large amount of energy by several nodes with a certain position, protection techniques are used in relation to the average left over energy of their neighbors. The stationary receivers are used to receive information for different regions.

D. Ideal Distance for Communication

Based on the power consumption pattern, the power consumption gradually rises as the communication distance increases. To avoid long-distance communication and save energy, multi-hop transmission is implemented. In general, the calculation of the distance between sensor nodes based on the Euclidean distance formula.

$$d = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \quad (8)$$

E. Network Initial Stage

The initialization stage begins after the nodes are allocated. The major aim of the stating stage is to exchange information in preparation for information transfer. First, a table is called Network data is generated for the system. This table has the location or position, region, neighbor information, and total power utilization of the every node in the sensor network. Figure 3 shows a block diagram of the nodes in the structure.

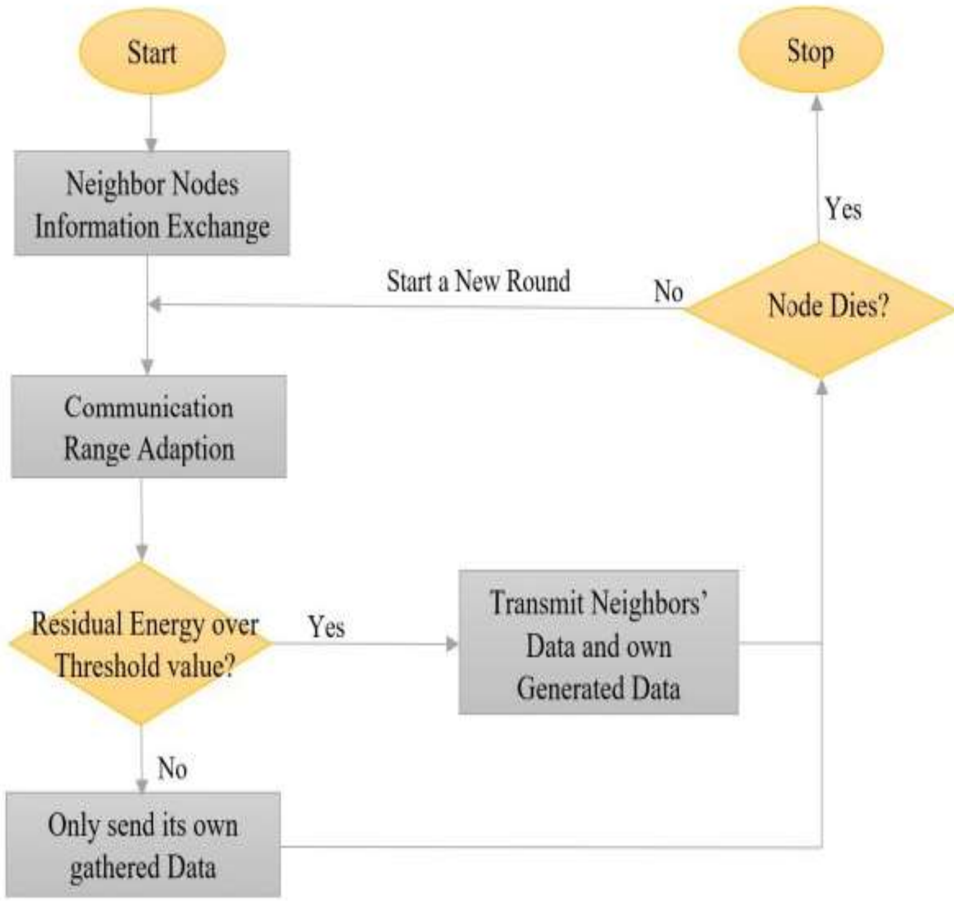


Fig 4. Work flow of nodes

F. Network Topology Generation

In general, for information transmission in the PEGASIS, each sensor point selects its closest neighbor as the transmission node, these results in high information transmission load and high network delay. With the proposed mechanism, finite energy consumption is achieved by reducing data transmission time. In the addition to information transfer, representative communication and hop counting are used the chain the design is dependent on base station and broadcast to every sensor points. Thus, all sensor points send information and are received in closest sensor points and choose the leader node to send information to the base station. Sensor nodes form circuits for internal connectivity, and all circuit member nodes send information to the circuit head using circuit-based routing concepts. This process involves two steps, namely the selection of the master node of the chain and the transfer of data on the chain. All chain heads create cluster-based routing as links. Finally, a selected cluster head sends the aggregated information to the base station. If a node in the selected circuit dies, then the cluster head is far from the base station, more energy is used to transmit data, so a duplicate circuit is created to transmit data in case of chain failure

5. Results and Discussion

The sensor network model is considered as the free space model. Have Tx and Rx separated by the distance d. The amplifying circuit is also there in Tx and Rx.

The following properties are considered for WSNs::

- All sensor points are scrambled and stationary.
- All sensor points are similar and have finite energy.
- The BS is immobile and may be within or outside the sensor's range.
- Each node collects data periodically and always has some data to pass.
- The node does not know the exact position or the position of the other nodes.
- Nodes are self-regulating and do not need to be monitored after use.
- Data collection is used to reduce the total amount of information passed.
- Each sensor point can act as a cluster head.

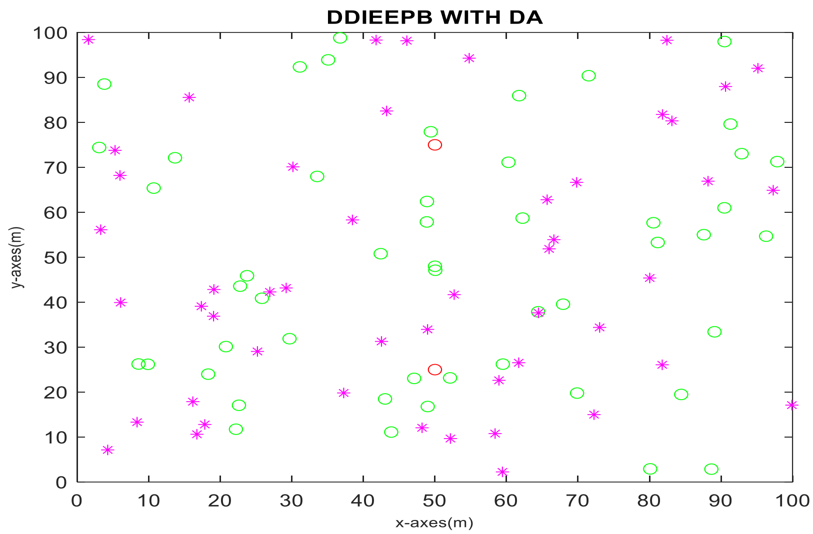


Fig5. Distribution of Nodes

The major objective of the work proposed in this paper is to upgrade IEEPB to DDIEEPB-DFO to maximize network life and minimize power utilization. Therefore, this document only focuses on improving network performance due to reduced energy consumption and shows the results of the sensor network lifetime percentage depends on the number of active sensor points per ring, residual power per ring and power utilization per node for each cycle Figure 6, Figure 7 and Figure 8 To evaluate the performance of the proposed PEGASIS-DA algorithm, the Matlab 2019 simulator was used to carry out the experiment. The proposed algorithm is compared with a typical PEGASIS.

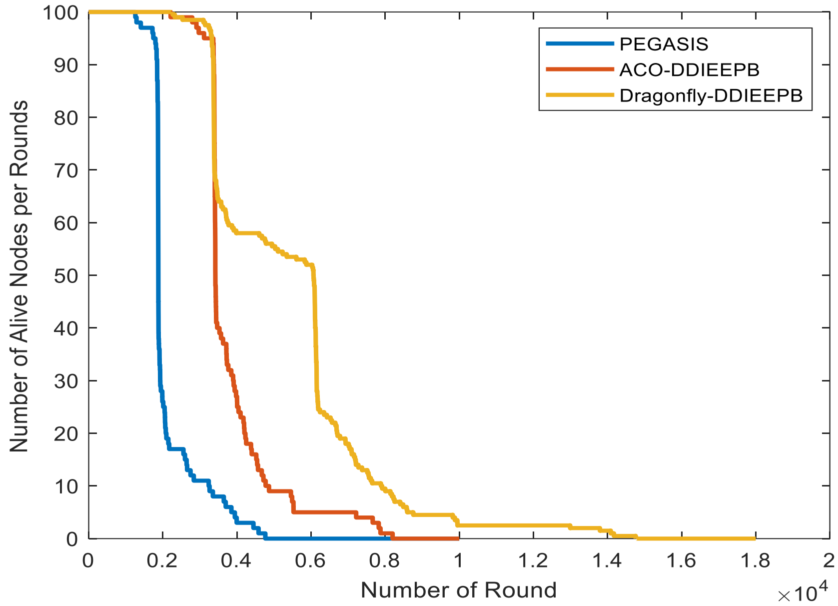


Fig6. Number of alive nodes

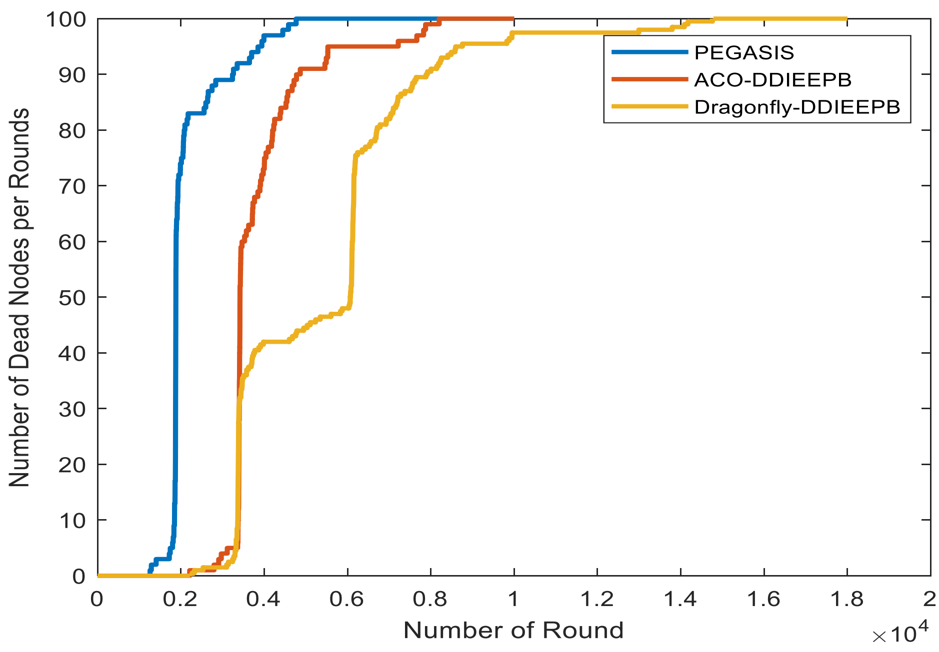


Fig7. Number of Dead Nodes

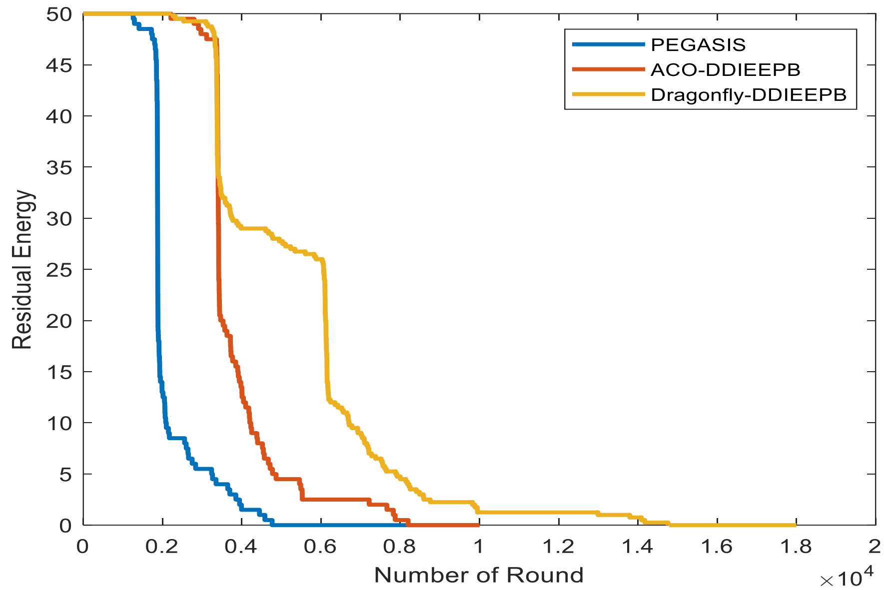


Fig8. Residual Energy

The result values obtained using various techniques are shown below.

------------------PEGASIS-----------------

First Dead PEGASIS : 1490.000000

Half Dead PEGASIS : 2279.000000

Last Dead PEGASIS : 4767.000000

Throughput PEGASIS : 82.967000

Packet Delivey Ratio PEGASIS : 84.791000

Delay PEGASIS : 18.879000

-----------------ACO-DDIEEPB-----------------

First Dead ACO-DDIEEPB : 2795.000000

Half Dead ACO-DDIEEPB : 3419.000000

Last Dead ACO-DDIEEPB : 8201.000000

Throughput ACO-DDIEEPB : 89.714000

Packet Delivey Ratio ACO-DDIEEPB : 91.936000

Delay PEGASIS : 13.851000

------------------DRAGON-DDIEEP-----------------

First Dead PEGASIS : 2324.000000

Half Dead PEGASIS : 6066.000000

Last Dead PEGASIS : 14765.000000

Throughput PEGASIS : 92.592100

Packet Delivey Ratio PEGASIS : 94.196300

Delay PEGASIS : 11.298400

6. Conclusion

In this paper, the PEGASIS protocol is used to increase the energy of a sensor network using optimization algorithms. Energy consumption reduction is designed and implemented by WSN. The PEGASIS-DA algorithm includes a two-chain dual sink to improve WSN performance. The PEGASIS is protocol is used along with the a Dragonflies algorithm is to reduce information redundancy, neighbouring node distances and long link transmission delays to finally achieve an optimal chain. The simulation results demonstrate the benefits of the proposed mechanism in terms of left over power state along with a significant increase in network life and effective delay reduction compared to the current PEGASIS protocol. In addition, hybrid optimization techniques can be used to obtain the best routing and energy-maximizing network.

References

[1] J. N. Al-Karaki and A. E. Kamal, “Routing techniques in wireless sensor networks: a survey,” *IEEE Wirel. Commun*, vol. 11, no. 6, pp. 6–28, 2004.

[2] T. K. Jain, D. S. Saini, and S. V. Bhooshan, “Cluster head selection in a homogeneous wireless sensor network ensuring full connectivity with minimum isolated nodes,” *J. Sensors*, vol. 2014, 2014.

[3] D. R. Dandekar and P. R. Deshmukh, “Energy balancing multiple sink optimal deployment in multi-hop wireless sensor networks,” in *2013 3rd IEEE International Advance Computing Conference (IACC)*, 2013, pp. 408–412.

[4] S. Lindsey and C. S. Raghavendra, “PEGASIS: Power-efficient gathering in sensor information systems,” *Proceedings, IEEE aerospace conference*, vol. 3, 2002.

[5] N. Ramluckun and V. Bassoo, “Energy-efficient chain-cluster based intelligent routing technique for Wireless Sensor Networks,” *Appl. Comput. Informatics*, 2018.

[6] B. Singh, E. S. Kaur, and B. Singh, “An Improved Energy-Efficient BBO-Based PEGASIS Protocol in Wireless Sensors Network,” *Int. J. Eng. Res. Appl*, vol. 4, no. 3, pp. 470–474, 2014.

[7] R. Vijayashree and M. Subramaniakumar, “Energy Efficient Packet Transmission-Chain Based Routing Algorithm Using Artificial Bee Colony Approach with Multiple Mobile Sinks in,” vol. 126, pp. 52448–52451, 2019.

[8] W. Linping, B. Wu, C. Zhen, and W. Zufeng, “Improved algorithm of PEGASIS protocol

introducing double cluster heads in wireless sensor network," *2010 International Conference on Computer*, vol. 1, pp. 148–151, 2010.

[9] A. Bachir, M. Dohler, T. Watteyne, and K. K. Leung, "MAC essentials for wireless sensor networks," *IEEE Commun. Surv. Tutorials*, vol. 12, no. 2, pp. 222–248, 2010.

[10] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wirel. Commun*, vol. 1, no. 4, pp. 660–670, 2002.

[11] S. Mirjalili, "Dragonfly algorithm: a new metaheuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput. Appl*, vol. 27, no. 4, pp. 1053–1073, 2016.

[12] R. W. Russell, M. L. May, K. L. Soltesz, and J. W. Fitzpatrick, "Massive swarm migrations of dragonflies (Odonata) in eastern North America," *Am. Midl. Nat*, vol. 140, no. 2, pp. 325–342, 1998.

[13] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proceedings of ICNN'95-International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.

[14] X. Yang, *Firefly Algorithm," Nature-Inspired Metaheuristic Algorithms Second Ed*. London: Springer, 2010.

[15] E. Yavuz, K. İbrahim, and Ü. İlker, *Optimization in Renewable Energy Systems*. Elsevier Ltd, 2017.

[16] A. A. Hussein and R. A. Khalid, "A Comparative Study of Swarm Intelligence-Based Optimization Algorithms in WSN"," *Asian Journal of Engineering and Applied Technology*, vol. 8, no. 3, pp. 1–7, 2019.

[17] A. Mohajerani1 and D. Gharavian, "An ant colony optimization based routing algorithm for extending network lifetime in wireless sensor networks"," *Wireless Networks*, vol. 8.

[18] R. Kumar and D. Kumar, "Hybrid swarm intelligence energy-efficient clustered routing algorithm for wireless sensor networks"," *Journal of Sensors*, vol. 2016, pp. 1–19, 2016.

