

Improving of Error Nodes in WSN with ENI Algorithm

Madhuri.K

Dept of Computer Science and Engineering
T.John Institute Of Technology
Bangalore, India

Ms. P. Bindhu Madhavi. Asst.Professor

Dept of Computer Science and Engineering
T.John Institute Of Technology
Bangalore, India

Abstract—An Error Node Improvement (ENI) algorithm is to expand the life span of a wireless sensor network while a small amount of the sensor nodes shut down. In genuine wireless sensor networks, the sensor nodes employ battery power supplies and thus have some degree of energy resources.

In adding together to the routing, it is vital to explore the optimization of sensor node replacement, reducing the replacement cost for some extent, and reuse a large amount routing paths when some sensor nodes are non functional. The algorithm is based on the grade diffusion algorithm pooled with the genetic algorithm.

The algorithm can upshot in smaller amount replacements of sensor nodes and additional reused routing paths. In the simulation, the algorithm increases the quantity of energetic nodes up to 8.7 times, reduces the rate of data crash by roughly 98.8%, and reduces the rate of power utilization by roughly 31.9 %.

Keywords — Grade diffusion algorithm; Genetic algorithm; Wireless sensor network; Non functional nodes

I. INTRODUCTION

Modern advances in micro processing, wireless and battery technology, and smart sensors have improved data processing wireless communication, and detection capability. In sensor networks, each sensor node has inadequate wireless computational power to process and shift the live data to the base station or data anthology center. Consequently, to increase the sensor area and the transmission area the wireless sensor network typically contains many sensor nodes.

In general, each sensor node has a low level of battery power that cannot be replenished. When the power of a sensor node is tired, wireless sensor network leaks will appear, and the unsuccessful nodes will not pass on data to the other nodes during transmission processing. As a result, the other sensor nodes will be loaded with increased transmission processing.

This paper proposes an error node improvement (ENI) algorithm to improve the lifetime of a wireless sensor network (WSN) when some of the sensor nodes blackout, either because they no longer have battery energy or they have reached their operational threshold.

Using the ENI algorithm can cause in smaller amount of replacements of sensor nodes and more reused routing paths. Accordingly, the algorithm not only enhances the

WSN lifetime but as well reduces the rate of replacing the sensor nodes.

II. RELATED WORK

The conventional approaches to sensor network routing contain the directed diffusion (DD) algorithm and the grade diffusion (GD) algorithm. The algorithm planned in this paper is based on the GD algorithm, with the objective of replacing smaller amount of sensor nodes that are out of order or have exhausted batteries, and of reusing the utmost amount of routing paths. These optimizations will ultimately enhance the WSN lifetime, as well reduces the rate of replacing the sensor nodes.

A. Directed Diffusion Algorithm

A sequence of routing algorithms for wireless sensor networks has been proposed in recent years. The aim of the DD algorithm is to reduce the data relay transmission counts for power management. The DD algorithm is a query-driven transmission procedure. The collected data is transmitted only if it matches the query from the sink node. In the DD algorithm, the sink node provides the queries in the form of attribute-value pairs to the other sensor nodes by broadcasting the query packets to the whole network. Subsequently, the sensor nodes send the data back to the sink node only when it fits the queries.

B. Grade Diffusion Algorithm

Grade Diffusion (GD) algorithm is to improve the ladder diffusion algorithm using ant colony optimization (LD-ACO) for wireless sensor networks. The GD algorithm not only creates the routing for each sensor node but also identifies a set of neighbor nodes to reduce the transmission loading. Each sensor node can select a sensor node from the set of neighbor nodes when its grade table lacks a node able to perform the relay. The GD algorithm can also record some information regarding the data relay. Then, a sensor node can select a node with a lighter loading or more available energy than the other nodes to perform the extra relay operation. That is, the GD algorithm updates the routing path in real time, and the event data is thus sent to the sink node quickly and correctly. Whether the DD or the GD algorithm is applied, the grade creating packages or interested query packets must first be broadcast. Then, the sensor nodes transfer the event data to the sink node,

according to the algorithm, when suitable events occur. The sensor routing paths are shown in Fig. 1.

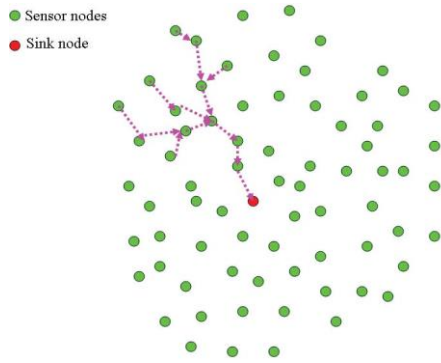


Fig.1. Wireless sensor node routing.

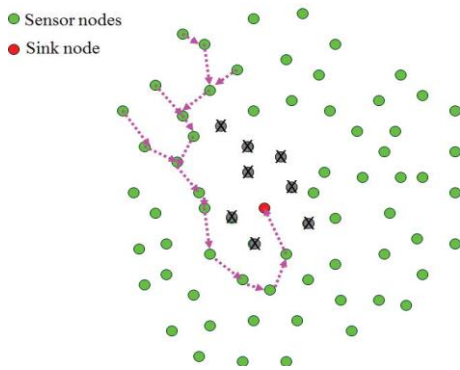


Fig.2. Wireless sensor node routing path when some nodes are not functioning.

The WSN may not succeed due to a variety of causes, including the following: the routing path might experience a break; the WSN sensing area might experience a leak; the batteries of some sensor nodes might be depleted, requiring more relay nodes; or the nodes wear out after the WSN has been in use a long period of time. In Fig. 2, the situation in which the outside nodes transfer event data to the sink node via the inside nodes (the sensor nodes near the sink node) in a WSN illustrate the accommodation measures for non-working nodes.

The inside nodes thus have the largest data transmission loading, consuming energy at a faster rate. If all the inside nodes deplete their energy or otherwise cease to function, the event data can no longer be sent to the sink node, and the WSN will no longer function.

The power consumption of the sensor nodes in WSNs is unavoidable. Conventional search techniques are often incapable of optimizing nonlinear functions with multiple variables.

One scheme, the genetic algorithm (GA), is a directed random search technique, based on the concept of natural genetics. An error node improvement algorithm based on the GD algorithm combined with the GA.

The ENI algorithm creates a routing table using the GD algorithm and replaces sensor nodes using the GA

when the number of sensor nodes that are not functioning exceeds the threshold.

III. ERROR NODES IMPROVEMENT ALGORITHM

In a Genetic algorithm, a population of candidate solutions to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties which can be mutated and altered.

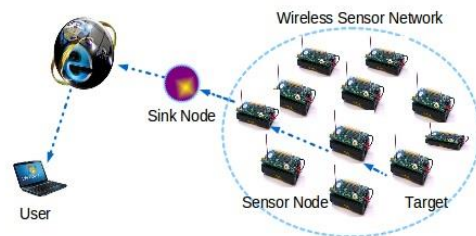


Fig.3. Connections between user and WSN

Based on Natural Selection:

After an initial population is randomly generated, the algorithm evolves through three operators:

- Selection which equates to survival of the fittest;
- Crossover which represents mating between individuals;
- Mutation which introduces random modifications.

System Architecture

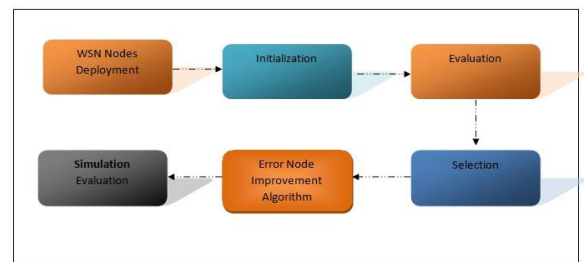


Fig.4. System architecture

Initialization: Initialization involves setting the parameters for the algorithm, creating the scores for the simulation, and creating the first generation of chromosomes. Here we generate two random values for Energy and Trust. The fitness value is thus created for every node by the sum of energy and trust values.

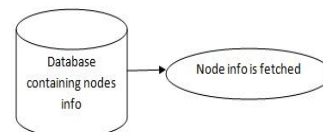


Fig.5. Initialization

Transfer: Includes all the details of nodes. Here the source and destination nodes are selected among the group of other nodes and this process will highlight the source and destination nodes in a wireless sensor network on selecting the submit option. The distance between each nodes are

calculated using distance formula and thus, finding out the neighboring nodes for every node. This process also includes finding top three paths which has a shortest path from source to destination node using Dijkstra's algorithm.

Selection: Evaluation: Each of the chromosomes in a generation must be evaluated for the selection process. In this process source and destination nodes are not considered, others nodes of the path are selected for fitness checking. Fitness checking is done by considering a threshold fitness value and comparing with the fitness values of the nodes.

- Key idea: give preference to better individuals, allowing them to pass on their genes to the next generation.
- The goodness of each individual depends on its fitness.
- Fitness may be determined by an objective function or by a subjective judgment.

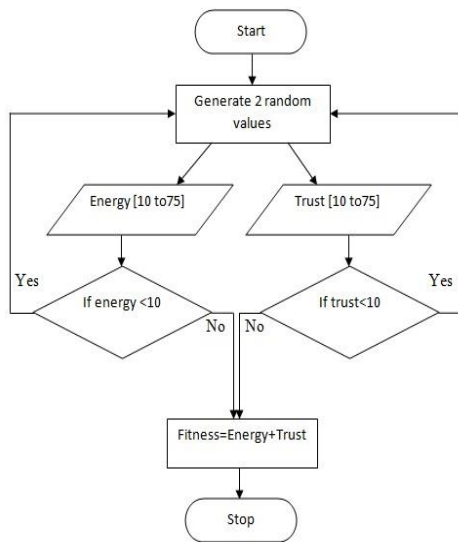


Fig.6.Flow chart for Initialization Process

Crossover: In crossover process, all the chromosomes are paired up, and with a probability crossover, they are crossed over. The crossover is accomplished by randomly choosing a site along the length of the chromosomes and exchanging the genes of the two chromosomes for each gene past this crossover site. The good nodes are crossed over in this process. Prime distinguished factor of GA from other optimization techniques

- Two individuals are chosen from the population using the selection operator
- A crossover site along the bit strings is randomly chosen
- The values of the two strings are exchanged up to this point
- If S1=000000 and S2=111111 and the crossover point is 2 then S1'=110000 and S2'=001111
- The two new offspring created from this mating are put into the next generation of the population
- By recombining portions of good individuals, this process is likely to create even better individuals

Mutation: After the crossover, for each of the genes of the chromosomes, the gene will be mutated to any one of the codes with a probability of Mutation, with the crossover and mutation completed; the chromosomes are once again evaluated for another round of selection.

- With some low probability, a portion of the new individuals will have some of their bits flipped.
- Its purpose is to maintain diversity within the population and inhibit premature convergence.
- Mutation alone induces a random walk through the search space
- Mutation and selection (without crossover) create a parallel, noise-tolerant, hill-climbing algorithms

IV. SIMULATION

A simulation of the error node improvement algorithm was performed to verify the method. The experiment was designed based on 3-D space, using 100× 100 × 100 units, and the scale of the coordinate axis for each dimension was set at 0 to 100. The radio ranges (transmission range) of the nodes were set to 15 units. In each of these simulations, the sensor nodes were distributed uniformly over the space. There are three sensor nodes randomly distributed in 10 × 10 × 10 space, and the Euclidean distance is at least 2 units between any two sensor nodes.

Therefore, there are 3000 sensor nodes in the 3-D wireless sensor network simulator, and the center node is the sink node. The data packages were exchanged between random source/destination pairs with 90 000 event data packages. In our simulations, the energy of each sensor node was set to 3600 Ws that is the actual available energy. Each sensor consumed 1.6 Ws when it conducts a completed data transformation (Rx +T x). In the GA algorithm, the population size was 20; the crossover rate was 50%; and the mutation rate was 2%.

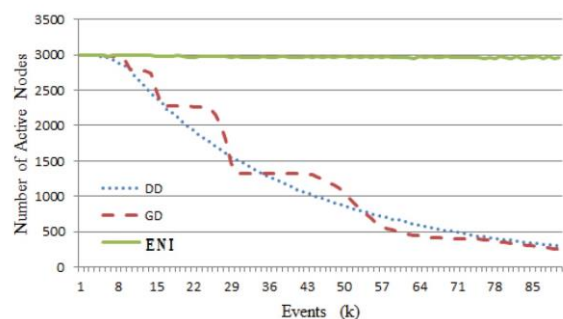


Fig.7. Number of active nodes.

The ENI, DD, and GD algorithms were implemented. The active sensor nodes and total data loss after 90 000 events are shown in Figs.7 and 8. The active nodes mean that the sensor node has enough energy to transfer data to other nodes, but some sensor nodes can be deleted from the active nodes list if their routing tables do not have a sensor node that can be used as a relay node, or if they are not in the routing table of any other sensor nodes.

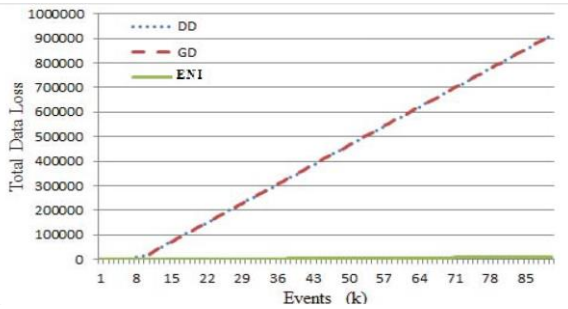


Fig.8.Total data loss.

The ENI algorithm has 2931 sensor nodes available, but the DD and GD algorithms only have 305 and 256 sensor nodes available after 90 000 events, as shown in Fig. 7. This new algorithm enhances the number of active nodes by 8.7 and 10.8 times, respectively. The ENI algorithm has the most active sensor nodes compared with the DD and GD algorithms because the algorithm can replace the sensor nodes after the number of nonfunctioning nodes exceeds the threshold, by using the GA algorithm.

Fig. 8 compares the total data loss using the ENI algorithm to the total data loss using the DD and GD algorithms. In this simulation, event data was destroyed and recorded into the loss count if the data had already been relayed over 20 times. Moreover, sensor nodes might detect the same event when an event appeared and transfer it to the sink node in this simulation setting. Hence, the total data loss might exceed 90 000 events. Therefore, sensor nodes can detect more events and transfer them to the sink node if the WSN lifetime is increased.

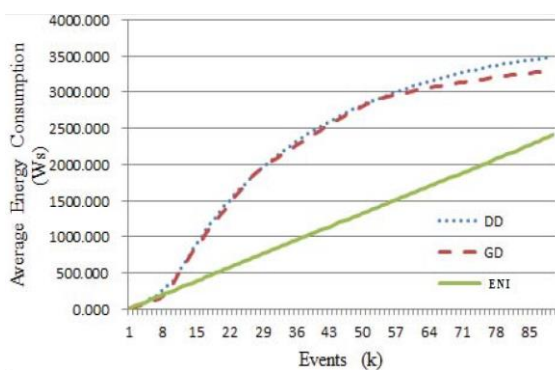


Fig.9.Average energy consumption.

TABLE I

AVERAGE ENERGY CONSUMPTION IN DIFFERENT NODE DENSITITE

Algorithm	10^{-3}	2×10^{-3}	3×10^{-3}	4×10^{-3}
	Nodes/Unit ³	Nodes/Unit ³	Nodes/Unit ³	Nodes/Unit ³
DD	3540.51 Ws	3517.18 Ws	3495.17 Ws	3505.48 Ws
GD	3132.4 Ws	3300.77 Ws	3298.29 Ws	3316.07 Ws
ENI	2969.56 Ws	2790.82 Ws	2407.68 Ws	2393.06 Ws

In Fig.8, the ENI algorithm exhibits smaller data losses because the algorithm can replace fewer sensor nodes and reuse more routing paths if the number of sensor nodes that are nonfunctioning exceeds the threshold. After the simulation, the ENI algorithm had only suffered 11025 data losses, but the DD and GD algorithm had suffered 912462 and 913449 data losses. This new algorithm can reduce data loss by 98.8% compared to the traditional algorithms.

Fig. 9 compares the average energy consumption of a WSN managed using the ENI algorithm to the average energy consumption using the DD and GD algorithms. The DD and GD algorithms allow the WSN to consume more energy after 8000 events because the inside nodes are energy-depleted, but the outside nodes continue to attempt to transfer event data to the sink node through the inside nodes until they are also energy depleted. After 90000 events, the DD and GD algorithm-managed WSNs had consumed 3495.17 Ws and 3298.29 Ws, respectively.

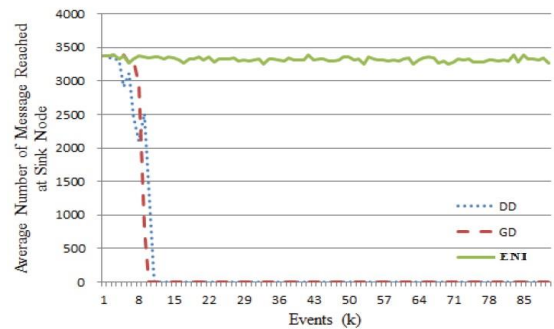


Fig.10.Average number of messages reaching the sink node.

The proposed algorithm increases the WSN lifetime by replacing some of the sensor nodes that are not functioning. In addition to enhancing the active nodes and reducing the data losses, the ENI algorithm reduces the relayed energy consumption by reducing the number of data relayed, as the replaced sensor nodes are usually used the most. After 90000 events, using the proposed algorithm, the WSN had consumed only 2407.68 Ws, and, compared to using the DD and GD algorithms, exhibited a reduction in energy consumption of 31.9% and 27%, respectively.

After that, we experiment different node densities in our simulation environment to compare the average energy consumption. We can find that the ENI algorithm has the least average energy consumption in all case, and it can save 31.73% energy at most in Table I. Hence, the ENI algorithm has the best energy saving performance no matter under any node densities.

The average number of messages that reach the sink node when each algorithm manages the network is compared. Using the traditional DD and GD algorithms, the sink node can receive no messages after 8000 events because all of the inside nodes are energy-depleted, and the WSN lifetime is ended.

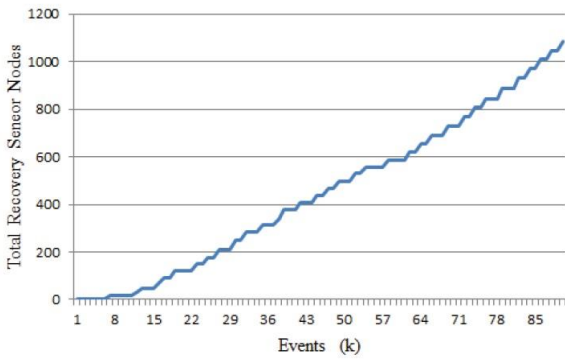


Fig. 11. Total number of sensor nodes recovered.

This proposed algorithm replaces energy-depleted sensor nodes to increase the WSN lifetime. Therefore, the average number of messages received using this algorithm is higher than when using the other algorithms. By using this algorithm, the sensor nodes are not only replaced, but the replacement cost is reduced, and more routing paths are reused.

The total number of sensor nodes recovered, 1085 sensor nodes were recovered, and the ENI algorithm continues to run for 34 iterations after 90 000 events. In the simulation, the algorithm replaced, on average, approximately 32 sensor nodes for each calculation, extending the lifetime of the WSN.

The average residual energy of the WSN using the ENI and the GD algorithms after 8000 and 90 000 events is shown. Because the ENI algorithm is based on the GD algorithm, the comparison demonstrates how the ENI has changed the algorithm. Using the GD algorithm, after 8000 events the grade 1 sensor nodes only have 145.57 Ws energy remaining, and the other grade sensor nodes still have enough energy to function. Using the ENI algorithm, the grade 1 sensor nodes still have 1568.34 Ws. The grade 1 sensor nodes are near the sink node, and they are relay nodes for the other grade sensor nodes, so they consume their energy rapidly. The ENI algorithm can replace some of the energy-depleted sensor nodes. Hence, the available sensor nodes are more numerous than when using the traditional algorithms. The average energy consumption for each grade is calculated after 90 000 events.

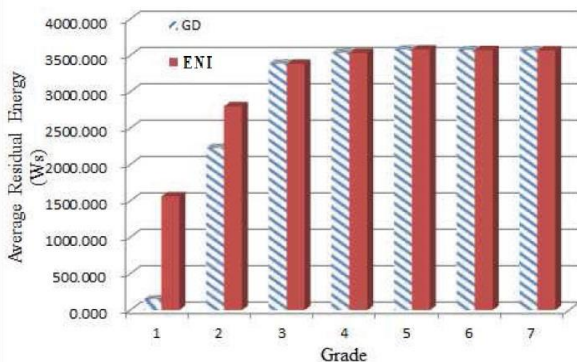


Fig.12.Average residual energy after 8000 events.

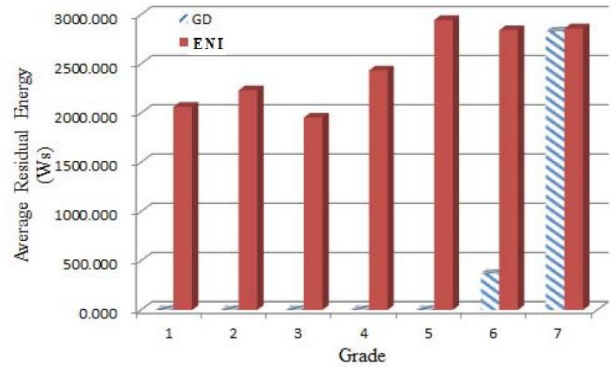


Fig. 13. Average residual energy after 90 000 events.

Using the GD algorithm, the sensor nodes consume their energy rapidly because they try to transfer event data to the sink node using neighbor nodes if the grade 1 sensor nodes are energy-depleted or their routing table is empty. The ENI algorithm has sample energy for each grade sensor node because the algorithm can replace the sensor nodes, but it reuses more routing paths compared to using the traditional algorithm. The number of replaced sensor nodes and the total number of messages reached the sink node for each replaced node are analyzed. For the first time of node recovery, the ENI algorithm just replaced 16 sensor nodes because there are not many sensor nodes that cannot work.

After the WSN has been in use for a considerable period of time, in average, 32 sensor nodes are replaced in each run. As a result, the WSN lifetime can be significantly extended. Each node is capable of detecting and sending approximately 27 327 event messages. The ratio of total messages to recovery nodes after each replacement is reported. The ENI algorithm tends to replaces Grade 1 sensor nodes in the first place, since the loading of the Grade 1 sensor nodes is larger than the loading of others.

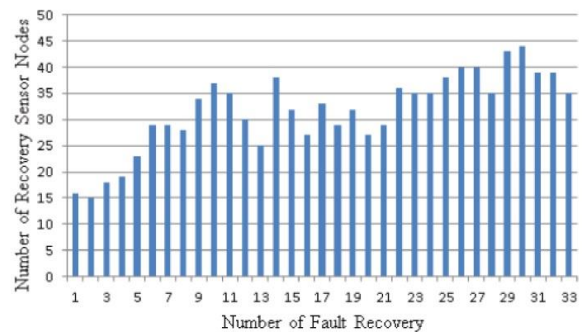


Fig.14. Number of recovery sensor nodes in each replacement.

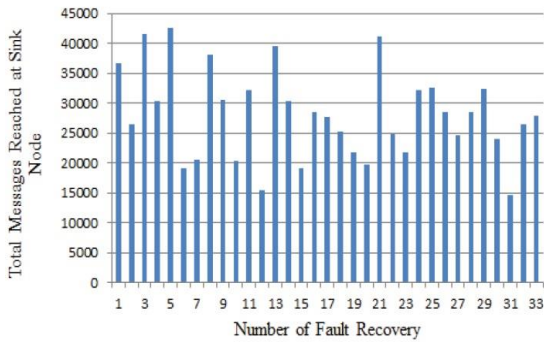


Fig.15.Total messages reaching the sink node for each replaced node.

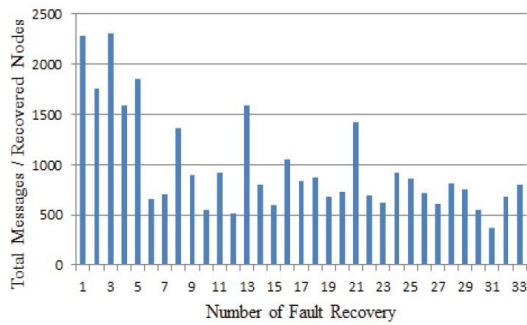


Fig.16. Rate of total messages to recovery nodes.

V. CONCLUSION

In real wireless sensor networks, the sensor nodes use battery power supplies and thus have limited energy resources. In addition to the routing, it is important to research the optimization of sensor node replacement, reducing the replacement cost, and reusing the most routing paths when some sensor nodes are nonfunctional. This paper proposes an error node improvement algorithm for WSN based on the grade diffusion algorithm combined with a genetic algorithm. The ENI algorithm requires replacing fewer sensor nodes and reuses the most routing paths, increasing the WSN lifetime and reducing the replacement cost.

In the simulation, the proposed algorithm increases the number of active nodes up to 8.7 times which exceeds other algorithms. The number of active nodes is enhanced 3.16 times on average after replacing an average of 32 sensor nodes for each calculation. The algorithm reduces the rate of

data loss by approximately 98.8% and reduces the rate of energy consumption by approximately 31.9%. Therefore, the ENI algorithm not only replaces sensor nodes, but also reduces the replacement cost and reuses the most routing paths to increase the WSN lifetime.

REFERENCES

- [1] J. A. Carballido, I. Ponzoni, and N. B. Brignole, "CGD-GA: A graphbased genetic algorithm for sensor network design," *Inf. Sci.*, vol. 177, no. 22, pp. 5091–5102, 2007.
- [2] F. C. Chang and H. C. Huang, "A refactoring method for cache-efficient swarm intelligence algorithms," *Inf. Sci.*, vol. 192, no. 1, pp. 39–49, Jun. 2012.
- [3] S. Corson and J. Macker, *Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations*. New York, NY, USA: ACM, 1999.
- [4] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Design*. New York, NY, USA: Wiley, 1997.
- [5] Z. He, B. S. Lee, and X. S. Wang, "Aggregation in sensor networks with a user-provided quality of service goal," *Inf. Sci.*, vol. 178, no. 9, pp. 2128–2149, 2008.
- [6] J. H. Ho, H. C. Shih, B. Y. Liao, and S. C. Chu, "A ladder diffusion algorithm using ant colony optimization for wireless sensor networks," *Inf. Sci.*, vol. 192, pp. 204–212, Jun. 2012.
- [7] J. H. Ho, H. C. Shih, B. Y. Liao, and J. S. Pan, "Grade diffusion algorithm," in *Proc. 2nd Int. Conf. Eng. Technol. Innov.*, 2012, pp. 2064–2068.
- [8] T. P. Hong and C. H. Wu, "An improved weighted clustering algorithm for determination of application nodes in heterogeneous sensor networks," *J. Inf. Hiding Multimedia Signal Process.*, vol. 2, no. 2, pp. 173–184, 2011.
- [9] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed diffusion for wireless sensor networking," *IEEE/ACM Trans. Netw.*, vol. 11, no. 1, pp. 2–16, Feb. 2003.
- [10] W. H. Liao, Y. Kao, and C. M. Fan, "Data aggregation in wireless sensor networks using ant colony algorithm," *J. Netw. Comput. Appl.*, vol. 31, no. 4, pp. 387–401, 2008.
- [11] T. H. Liu, S. C. Yi, and X. W. Wang, "A fault management protocol for low-energy and efficient wireless sensor networks," *J. Inf. Hiding Multimedia Signal Process.*, vol. 4, no. 1, pp. 34–45, 2013.
- [12] J. Pan, Y. Hou, L. Cai, Y. Shi, and X. Shen, "Topology control for wireless sensor networks," in *Proc. 9th ACM Int. Conf. Mobile Comput. Netw.*, 2003, pp. 286–299.
- [13] E. M. Royer and C. K. Toh, "A review of current routing protocols for ad-hoc mobile networks," *IEEE Personal Commun.*, vol. 6, no. 2, pp. 46–55, Apr. 1999.
- [14] H. C. Shih, S. C. Chu, J. Roddick, J. H. Ho, B. Y. Liao, and J. S. Pan, "A reduce identical event transmission algorithm for wireless sensor networks," in *Proc. 3rd Int. Conf. Intell. Human Comput. Interact.*, 2011, pp. 147–15