Fault Node Recovery Algorithm for a Wireless Sensor Network

Hong-Chi Shih, *Student Member, IEEE*, Jiun-Huei Ho, Bin-Yih Liao, *Member, IEEE*, and Jeng-Shyang Pan, *Senior Member, IEEE*

***Abstract*— This paper proposes a fault node recovery algorithm to enhance the lifetime of a wireless sensor network when some of the sensor nodes shut down. The algorithm is based on the grade diffusion algorithm combined with the genetic algorithm. The algorithm can result in fewer replacements of sensor nodes and more reused routing paths. In our simulation, the proposed algorithm increases the number of active nodes up to 8.7 times, reduces the rate of data loss by approximately 98.8%, and reduces the rate of energy consumption by approximately 31.1%.**

***Index Terms*— Genetic algorithm, grade diffusion (GD) algorithm, gradient diffusion algorithm, wireless sensor networks (WSN).**

I. Introduction

RECENT advances in micro processing, wireless and battery technology, and smart sensors have enhanced data processing [3], [11], [13], wireless communication, and detection capability. In sensor networks, each sensor node has limited wireless computational power to process and transfer the live data to the base station or data collection center [2], [5], [8]. Therefore, to increase the sensor area and the transmission area [1], [12], the wireless sensor network usually contains many sensor nodes. Generally, each sensor node has a low level of battery power that cannot be replenished. When the energy of a sensor node is exhausted, wireless sensor network leaks will appear, and the failed nodes will not relay data to the other nodes during transmission processing. Thus, the other sensor nodes will be burdened with increased transmission processing.

This paper proposes a fault node recovery (FNR) algorithm to enhance the lifetime of a wireless sensor network (WSN) when some of the sensor nodes shut down, either because they no longer have battery energy or they have reached their operational threshold. Using the FNR algorithm can result in fewer replacements of sensor nodes and more reused routing paths. Thus, the algorithm not only enhances the WSN lifetime but also reduces the cost of replacing the sensor nodes.

Manuscript received November 5, 2012; revised January 11, 2013, February 20, 2013, and March 13, 2013; accepted March 21, 2013. Date of publication April 1, 2013; date of current version May 31, 2013. The associate editor coordinating the review of this paper and approving it for publication was Prof. Octavian Postolache.

H.-C. Shih and B.-Y. Liao are with the Department of Electronic Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung 807, Taiwan (e-mails: hqshi@bit.kuas.edu.tw; byliao@cc.kuas.edu.tw).

J.-H. Ho is with the Department of Computer Science and Information Engineering, Cheng Shiu University, Kaohsiung 803, Taiwan (e-mail: jhho@csu.edu.tw).

J.-S. Pan is with the Innovative Information Industry Research Center, Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen 518006, China (e-mail: jengshyangpan@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSEN.2013.2255591

II. Related Work

The traditional approaches to sensor network routing include the directed diffusion (DD) [9] algorithm and the grade diffusion (GD) [13] algorithm. The algorithm proposed in this paper is based on the GD algorithm, with the goal of replacing fewer sensor nodes that are inoperative or have depleted batteries, and of reusing the maximum number of routing paths. These optimizations will ultimately enhance the WSN lifetime and reduce sensor node replacement cost.

A. Directed Diffusion Algorithm

A series of routing algorithms [10], [14] for wireless sensor networks have been proposed in recent years. C. Intanagonwiwat *et al.* presented the Directed Diffusion (DD) algorithm [9] in 2003. The goal of the DD algorithm is to reduce the data relay transmission counts for power management. The DD algorithm is a query-driven transmission protocol. 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

H. C. Shih *et al.* presented the Grade Diffusion (GD) algorithm [7] in 2012 to improve the ladder diffusion algorithm using ant colony optimization (LD-ACO) for wireless sensor networks [6]. 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.

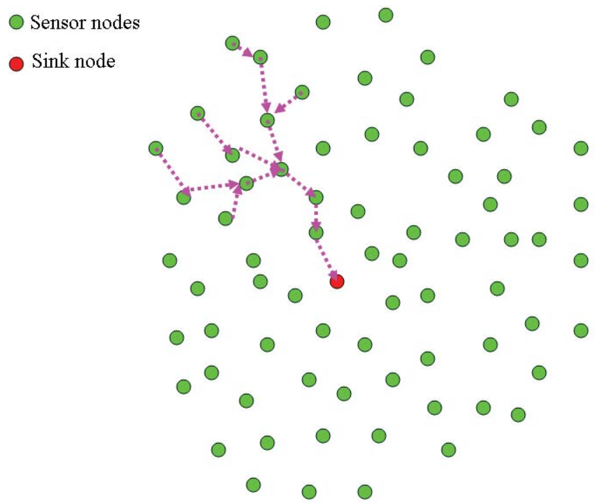

Fig. 1. Wireless sensor node routing.

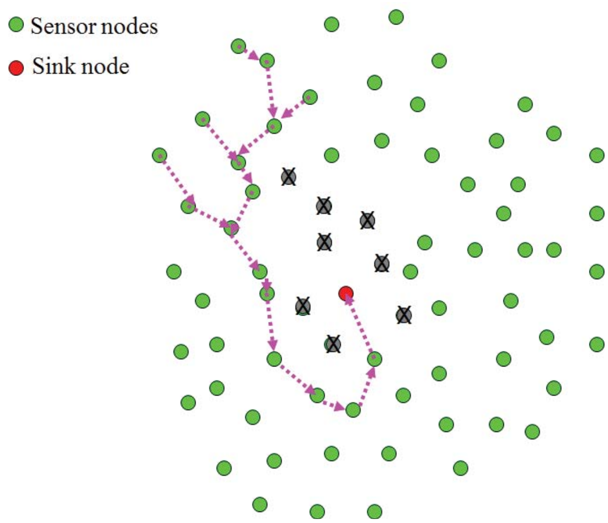

Fig. 2. Wireless sensor node routing path when some nodes are not functioning.

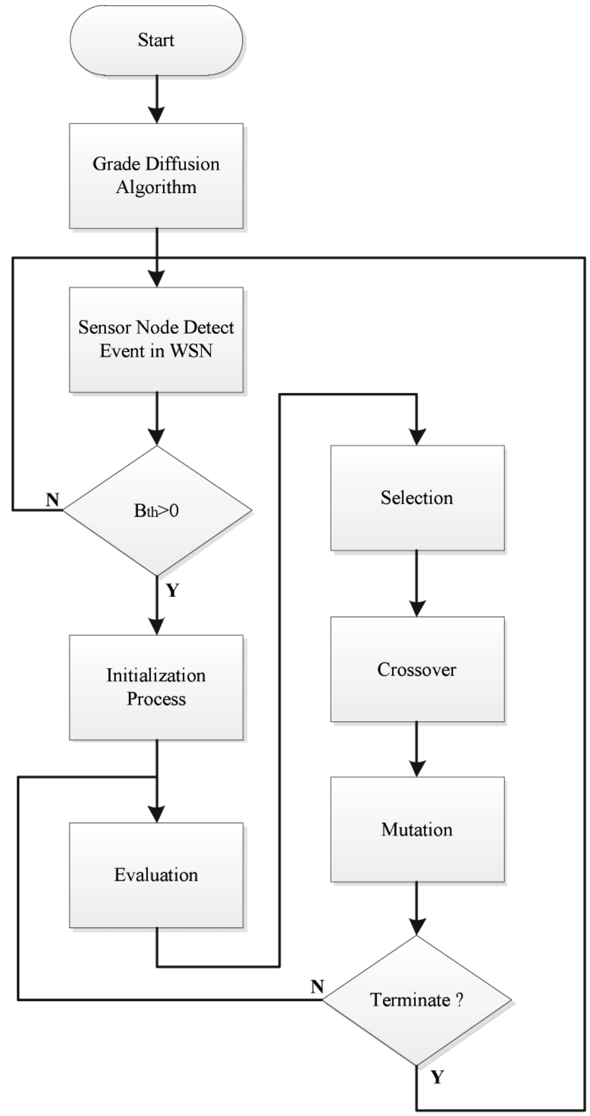

Fig. 3. Fault node recovery algorithm flow chart.

The WSN may fail 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. This paper, however, proposes an algorithm to search for and replace fewer sensor nodes and to reuse the most routing paths. Conventional search techniques are often incapable of optimizing nonlinear functions with multiple variables. One scheme, the genetic algorithm (GA) [4], is a directed random search technique developed in 1975, based on the concept of natural genetics. The current paper proposes a fault node recovery (FNR) algorithm based on the GD algorithm combined with the GA. The FNR 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. This algorithm not only reuses the most routing paths to enhance the WSN lifetime but also reduces the replacement cost.

III. Fault Node Recovery Algorithm

This paper proposes a fault node recovery (FNR) algorithm for WSNs based on the grade diffusion algorithm combined with the genetic algorithm. The flow chart is shown in Fig. 3. The FNR algorithm creates the grade value, routing table, neighbor nodes, and payload value for each sensor node using the grade diffusion algorithm. In the FNR algorithm, the number of nonfunctioning sensor nodes is calculated during the wireless sensor network operation, and the parameter B_{th} is calculated according to (1).

9	7	10	81	23	57	34	46	66	70
0	0	1	0	1	1	0	1	1	0

Fig. 4. Chromosome and its gene.

In Fig. 3, the FNR algorithm creates the grade value, routing table, a set of neighbor nodes, and payload value for each sensor node, using the grade diffusion algorithm. The sensor nodes transfer the event data to the sink node according to the GD algorithm when events appear. Then, B_{th} is calculated according to (1) in the FNR algorithm. If B_{th} is larger than zero, the algorithm will be invoked and replace nonfunctioning sensor nodes by functional nodes selected by the genetic algorithm. Then the wireless sensor network can continue to work as long as the operators are willing to replace sensors

$$B_{\text{th}} = \sum_{i=1}^{\max\{\text{Grade}\}} T_i$$

$$T_i = \begin{cases} 1, & \frac{N_i^{\text{now}}}{N_i^{\text{original}}} < \beta \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

In (1), Grade is the sensor node's grade value. The variable N_i^{original} is the number of sensor nodes with the grade value i. The variable N_i^{now} is the number of sensor nodes still functioning at the current time with grade value i. The parameter β is set by the user and must have a value between 0 and 1. If the number of sensor nodes that function for each grade is less than β, T_i will become 1, and B_{th} will be larger than zero. Then, the algorithm will calculate the sensor nodes to replace using the genetic algorithm.

The parameters are encoded in binary string and serve as the chromosomes for the GA. The elements (or bits), i.e., the genes, in the binary strings are adjusted to minimize or maximize the fitness value. The fitness function generates its fitness value, which is composed of multiple variables to be optimized by the GA. At each iteration of the GA, a predetermined number of individuals will produce fitness values associated with the chromosomes.

There are 5 steps in the genetic algorithm: Initialization, Evaluation, Selection, Crossover, and Mutation. Descriptions of the steps follow.

A. Initialization

In the initialization step, the genetic algorithm generates chromosomes, and each chromosome is an expected solution. The number of chromosomes is determined according to the population size, which is defined by the user. Each chromosome is a combination solution, and the chromosome length is the number of sensor nodes that are depleted or nonfunctioning. The elements in the genes are either 0 or 1. A 1 means the node should be replaced, and a 0 means that the node will not be replaced.

Fig. 4 represents a chromosome. The chromosome length is 10 and the gene is 0 or 1, chosen randomly in the initialization step. In this case, there are 10 sensor nodes not functioning, and their node numbers are 9, 7, 10, 81, 23, 57, 34, 46, 66, and 70.

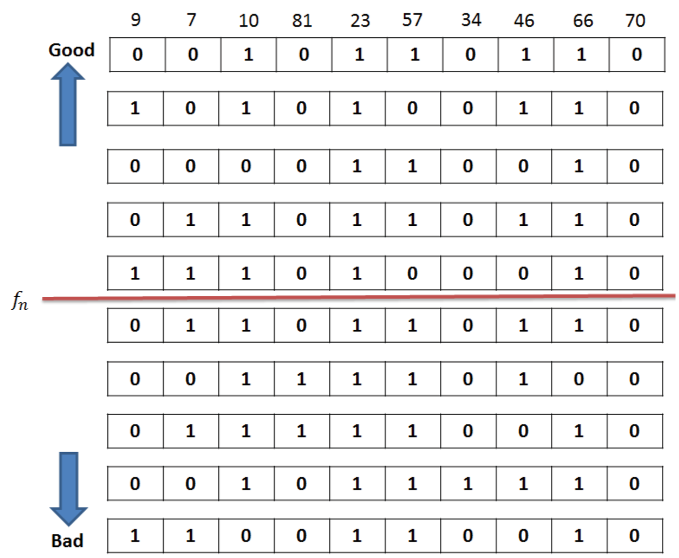

Fig. 5. Selection step.

B. Evaluation

In general, the fitness value is calculated according to a fitness function, and the parameters of the fitness function are the chromosome's genes. However, we cannot put genes directly into the fitness function in the FNR algorithm, because the genes of the chromosome are simply whether the node should be replaced or not. In the FNR algorithm, the goal is also to reuse the most routing paths and to replace the fewest sensor nodes. Hence, the number of routing paths available if some nonfunctioning sensor nodes are replaced is calculated, and the fitness function is shown as (2)

$$f_n = \sum_{i=1}^{\max(\text{Grade})} \frac{P_i \times \text{TP}^{-1}}{N_i \times \text{TN}^{-1}} \times i^{-1}. \quad (2)$$

In (2):

$N_i =$ the number of replaced sensor nodes and their grade value at i.

$P_i =$ the number of re-usable routing paths from sensor nodes with their grade value at i.

$TN =$ total number of sensor nodes in the original WSN.

$TP =$ total number of routing paths in the original WSN.

In (2), a high fitness value is sought because the WSN is looking for the most available routing paths and the least number of replaced sensor nodes.

C. Selection

The selection step will eliminate the chromosomes with the lowest fitness values and retain the rest. We use the elitism strategy and keep the half of the chromosomes with better fitness values and put them in the mating pool. The worse chromosomes will be deleted, and new chromosomes will be made to replace them after the crossover step. The process is shown in Fig. 5.

D. Crossover

The crossover step is used in the genetic algorithm to change the individual chromosome. In this algorithm, we use the

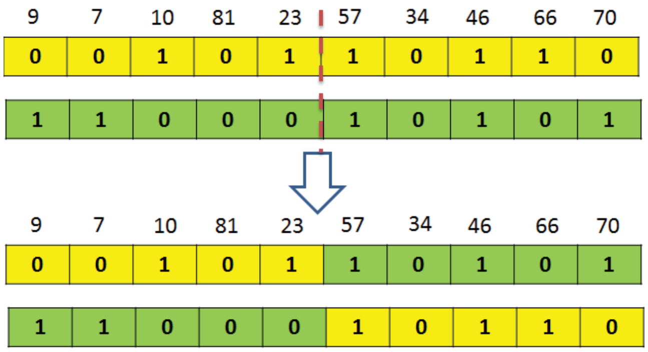

Fig. 6. Crossover step.

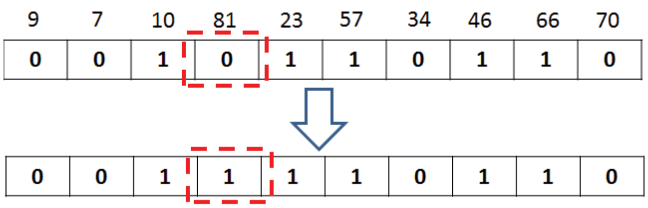

Fig. 7. Mutation step.

one-point crossover strategy to create new chromosomes, as shown in Fig. 6. Two individual chromosomes are chosen from the mating pool to produce two new offspring. A crossover point is selected between the first and last genes of the parent individuals. Then, the fraction of each individual on either side of the crossover point is exchanged and concatenated. The rate of choice is made according to roulette-wheel selection and the fitness values.

E. Mutation

The mutation step can introduce traits not found in the original individuals and prevents the GA from converging too fast. In this algorithm, we simply flip a gene randomly in the chromosome, as shown in Fig. 7.

The chromosome with the best fitness value is the solution after the iteration. The FNR algorithm will replace the sensor nodes in the chromosome with genes of 1 to extend the WSN lifetime.

IV. Simulation

A simulation of the fault node recovery algorithm as described in Section 3 was performed to verify the method. The experiment was designed based on 3-D space, using $100 \times 100 \times 100$ 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 $10 \times 10 \times 10$ 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 $(Rx + Tx)$. In the GA algorithm, the population size was 20; the crossover rate was 50%; and the mutation rate was 2%.

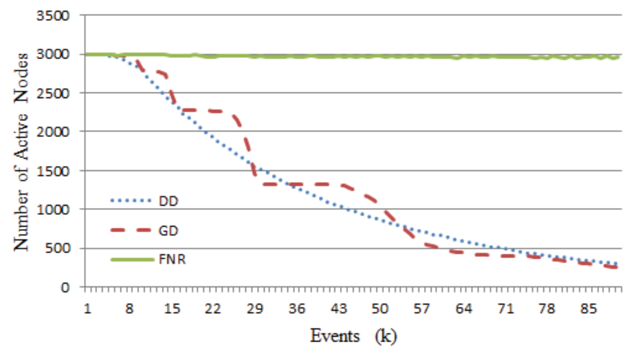

Fig. 8. Number of active nodes.

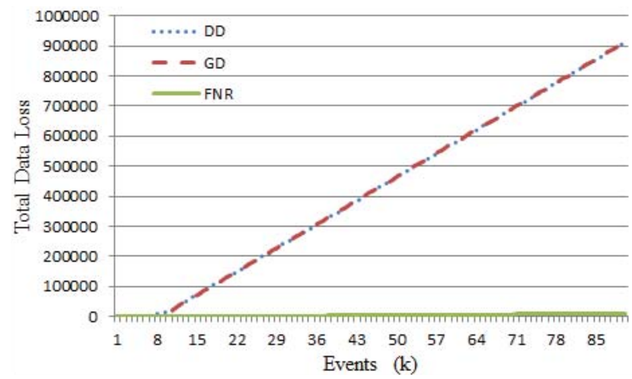

Fig. 9. Total data loss.

The FNR, DD, and GD algorithms were implemented. The active sensor nodes and total data loss after 90 000 events are shown in Figs. 8 and 9. 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.

The FNR 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. 8. This new algorithm enhances the number of active nodes by 8.7 and 10.8 times, respectively. The FNR 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. 9 compares the total data loss using the FNR 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

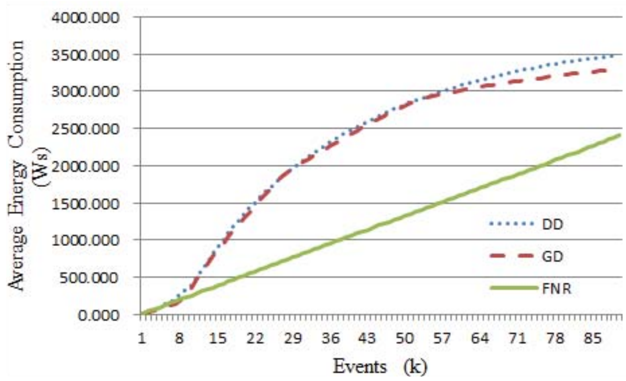

Fig. 10. Average energy consumption.

TABLE I
AVERAGE ENERGY CONSUMPTION IN DIFFERENT NODE DENSITITE

Algorithm	10^{-3} Nodes/Unit^3	2×10^{-3} Nodes/Unit^3	3×10^{-3} Nodes/Unit^3	4×10^{-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
FNR	2969.56 Ws	2790.82 Ws	2407.68 Ws	2393.06 Ws

and transfer them to the sink node if the WSN lifetime is increased.

In Fig. 9, the FNR 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 FNR algorithm had only suffered 11 025 data losses, but the DD and GD algorithm had suffered 912 462 and 913 449 data losses. This new algorithm can reduce data loss by 98.8% compared to the traditional algorithms.

Fig. 10 compares the average energy consumption of a WSN managed using the FNR 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 8 000 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 90 000 events, the DD and GD algorithm-managed WSNs had consumed 3495.17 Ws and 3298.29 Ws, respectively.

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 FNR algorithm reduces the relayed energy consumption by reducing the number of data relayed, as the replaced sensor nodes are usually used the most. After 90 000 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 31.1% and 27%, respectively.

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

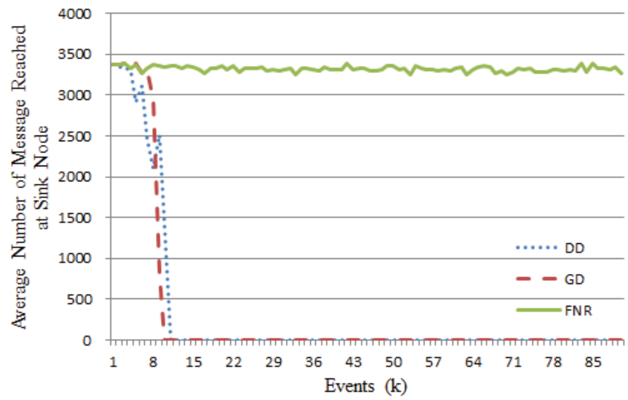

Fig. 11. Average number of messages reaching the sink node.

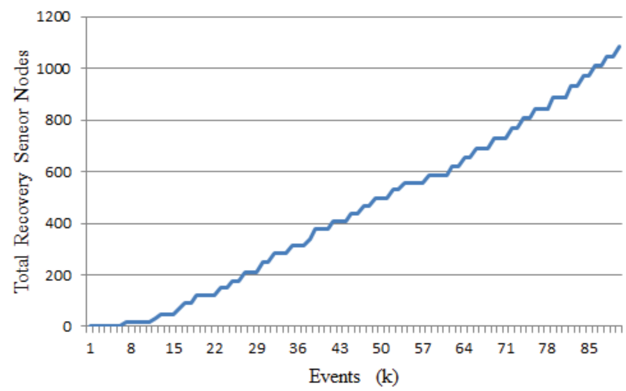

Fig. 12. Total number of sensor nodes recovered.

The average number of messages that reach the sink node when each algorithm manages the network is compared in Fig. 11. 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. 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 is shown in Fig. 12.

From Fig. 12, 1085 sensor nodes were recovered, and the FNR 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. In Figs. 13 and 14, the average residual energy of the WSN using the FNR and the GD algorithms after 8000 and 90 000 events is shown. Because the FNR algorithm is based on the GD algorithm, the comparison demonstrates how the FNR has changed the algorithm.

In Fig. 13, 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 FNR algorithm, the grade 1 sensor

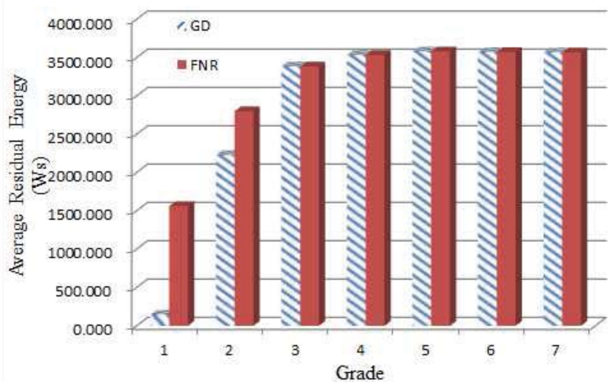

Fig. 13. Average residual energy after 8000 events.

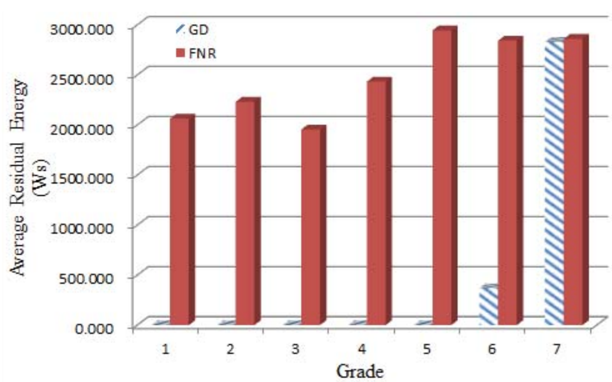

Fig. 14. Average residual energy after 90 000 events.

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 FNR algorithm can replace some of the energy-depleted sensor nodes. Hence, the available sensor nodes are more numerous than when using the traditional algorithms.

In Fig. 14, the average energy consumption for each grade is calculated 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 FNR algorithm has ample 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, as shown in Figs. 15 and 16.

For the first time of node recovery, the FNR algorithm just replaced 16 sensor nodes because there are not many sensor nodes that cannot work, as shown in Fig. 15. 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, as shown in Fig. 16.

The ratio of total messages to recovery nodes after each replacement is reported in Fig. 17. The FNR algorithm tend

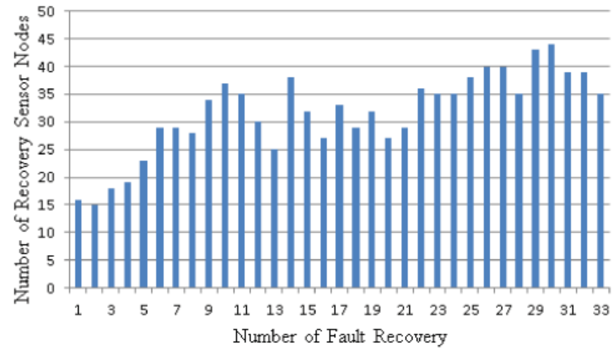

Fig. 15. Number of recovery sensor nodes in each replacement.

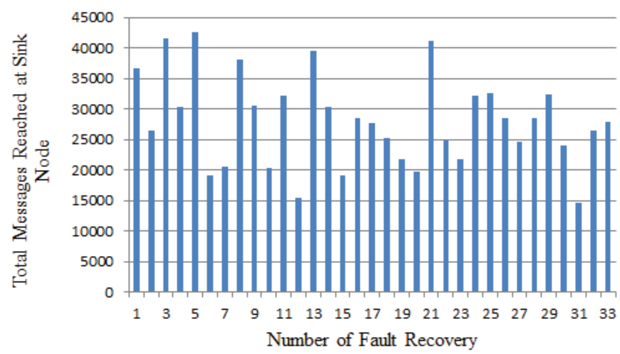

Fig. 16. Total messages reaching the sink node for each replaced node.

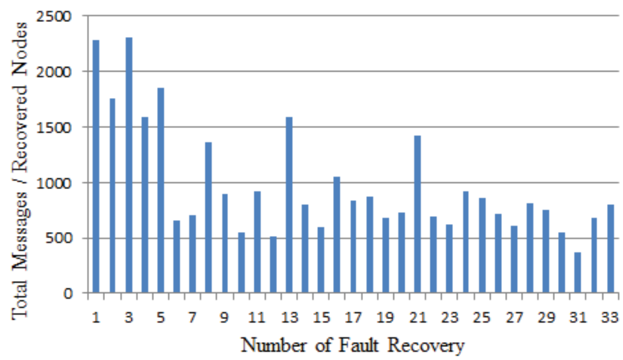

Fig. 17. Rate of total messages to recovery nodes.

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.

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 a fault node recovery algorithm for WSN based on the grade diffusion algorithm combined with a genetic algorithm. The FNR 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. 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 31.1%. Therefore, the FNR 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 graph-based 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. 147–154.

Hong-Chi Shih (S'08) received the M.S. degree from the National Kaohsiung University of Applied Science, Kaohsiung, Taiwan, where he is currently pursuing the Ph.D. degree at the Department of Electrical Engineering.

Jiun-Huei Ho received the B.S. degree in electronic engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 1990, and the M.S. degree from the Institute of Computer and Information Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan, in 1994, and the Ph.D. degree from the Department of Computer Science and Engineering, National Sun Yat-Sen University, in 2005.

He is currently an Associate Professor with the Department of Computer Science and Information Engineering, Cheng Shiu University, Kaohsiung, Taiwan. His current research interests include wireless sensor network, mobile computing systems, internet technology, and distributed system.

Bin-Yih Liao (M'08) received the Ph.D. degree from the Department of Electronic Engineering, National Cheng Kung University, Tainan, Taiwan.

He is currently a Professor with the Department of Electrical Engineering, National Kaohsiung University of Applied Science, Kaohsiung, Taiwan. His current research interests include the areas of image processing, information security, and embedded system.

Jeng-Shyang Pan (SM'09) received the B.S. degree in electronic engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 1986, the M.S. degree in communication engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1988, and the Ph.D. degree in electrical engineering from the University of Edinburgh, Edinburgh, U.K., in 1996.

He is currently a Professor with the Department of Electronic Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan. He was invited to be a Doctoral Advisor with the University of South Australia, Adelaide, Australia, and the Shenzhen Graduate School, Harbin Institute of Technology, Harbin, China. His current research interests include soft computing, information security, and signal processing.