International Journal of Emerging Engineering Research and Technology
Volume 2, Issue 9, December 2014, PP 70-76
ISSN 2349-4395 (Print) & ISSN 2349-4409 (Online)

Fault Node Recovery Algorithm for a Wireless Sensor Network

Chaitrali Brahme, Srushti Gadadare, Rutushree Kulkarni, Pranjal Surana, Prof. M. V. Marathe

Department of Computer Engineering, Pune Vidyarthi Griha's College of Engineering and Technology, Pune, India

Abstract: *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. Therefore, it is necessary that network failures are detected in advance and appropriate measures are taken to sustain network operation. 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*

Keywords: *Genetic algorithm, grade diffusion (GD) algorithm, directed diffusion algorithm (DD), ladder diffusion algorithm(LD), wireless sensor networks(WSN)*

1. Introduction

Wireless Sensor networks have various applications in Health-care monitoring, Air-pollution monitoring, Landslide detection, water quality monitoring and industrial monitoring. These sensor networks have enhanced data processing, wireless communication and detection capability. Wireless sensor networks consist of large number of heterogeneous sensor node devices spread over a large field. The main task of a sensor node in a sensor field is to detect events, perform quick local data processing, and then transmit the data.

WSN Sensor nodes are severely constrained in terms of storage resources, computational capabilities, communication bandwidth and power supply. The wireless sensor node, being a micro-electronic device, can only be equipped with a limited power source (<0.5 Ah, 1.2 V). In some application scenarios, replenishment of power resources might be impossible. Sensor node lifetime, therefore, shows a strong dependence on battery life time. In a multihop ad hoc sensor network, each node plays the dual role of data originator and data router. The dis-functioning of few nodes can cause significant topological changes and might require re-routing of packets and re-organization of the network. Hence, power conservation and power management take on additional importance. It is for these reasons that researchers are currently focusing on the design of power-aware protocols and algorithms for sensor networks [1],[3],[5],[8].

In order to increase the lifetime of wireless sensor network due to various problems with the sensor nodes like limited power source or operational threshold, we have proposed a Fault Node Recovery (FNR) algorithm. With the help of this algorithm such faulty sensor nodes can be detected and there are fewer replacements of these nodes in order to increase the lifetime of WSN [9],[11].

2. Related Work

2.1. Directed Diffusion Algorithm

In recent years in order to make WSN efficient, several algorithms have been proposed. C. Intanagonwiwatet. al. presented the Directed Diffusion(DD) algorithm. Directed Diffusion is designed for robustness, scaling and energy efficiency. It is data centric. Directed diffusion consists of several elements: interests, data messages, gradients, and reinforcements [6].

The DD algorithm is a query-driven transmission protocol. The collected data are transmitted only if they fit the query from the sink node, thereby reducing the power consumption from data transmission. First, the sink node provides interested queries in the form of attribute-value pairs to the other sensor nodes by broadcasting the interested query packets to the entire network. Subsequently, the sensor nodes only send the collected data back to the sink node if they fit the interested queries. In

DD, all of the sensor nodes are bound to a route when broadcasting the interested queries, even if the route is such that it will never be used. In addition, several circle routes, which are built simultaneously when broadcasting the queries, result in wasted power consumption and storage. In the real world, the number of the sensor nodes in a system is in the hundreds or even thousands. Such a waste of power consumption and storage becomes worse and the circle route problem becomes more serious with larger-sized systems.

2.2. Ladder Diffusion Algorithm

Ladder diffusion (LD) algorithm is used to map out the data relay routes in wireless sensor nodes. The algorithm focuses on balancing the data transmission load, increasing the lifetime of sensor nodes and their transmission efficiency. This study evaluates the performance of this algorithm for random wireless sensor networks with regard to the number of sensor nodes and relay hops required for data collection. The ladder diffusion algorithm is used to identify routes from sensor nodes to the sink node and avoid the generation of circle routes using the ladder diffusion process. The LD algorithm is fast and completely creates the ladder table in each sensor node based on the entire wireless sensor network by issuing the ladder create packet that is created from the sink node[4],[7].

3. ALGORITHMS USED

3.1. Grade Diffusion

In 2011, H. C. Shih et al.[12]proposed a Grade Diffusion algorithm that improves upon the LD-ACO algorithm[4] to enhance node lifetime, raise transmission efficiency, and solve the problem of node routing and energy consumption. The GD algorithm broadcast grade completely and quickly creates packages from the sink node to every node in the wireless sensor network by the LD-ACO algorithm. The GD algorithm increases the sensor node's lift time and the sensor node's transmission effect. Moreover, the sensor node can save some backup nodes to reduce the energy for the re-looking routing by GD algorithm in case the sensor node's routing is broken. Finally, the grade diffusion algorithm has the less data package transmission loss and the hop count than the tradition algorithms in our simulate setting. Hence, in addition to balance the sensor node's loading and reduce the energy consumption, our algorithm can send the data package to destination node quickly and correctly.

The Grade Diffusion algorithm overcomes the disadvantages of Direct Diffusion algorithm by broadcasting the neighbors to only first neighbor set. After that nodes are picked up based on hop count or rules and the amount of RREQ exchange is reduced hence amount of power required is less as compared to Direct Diffusion. However problem still persist as the number of routes discovered increases the battery power decreases and node becomes obsolete sooner. 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.

3.1.1. Explanation

Grade Diffusion Algorithm is proposed to solve the power consumption and transmission routing problems in wireless sensor networks[10]. In the grade diffusion algorithm the first step is to assign the grade for the sensor nodes and to update the routing table for each node. The data transmission takes place from higher grade nodes to lower grade nodes and hence it is named as grade diffusion algorithm. The GD algorithm is fast and completely creates the grade table in each sensor node based on the entire wireless sensor network by issuing the grade create packet that is created from sink node. The process of grade diffusion is as follows.

First, the sink node broadcasts the grade-creating package with the grade value of zero. A grade value of one means that the sensor node receiving this grade-creating package transmits data to the sink node requires only one hop. The sensor nodes "b" and "c" receive a grade-creating package with a grade value of one from sink node "a". The sensor nodes "b" and "c" increase the grade value of the grade-creating package to two and broadcast the modified grade-creating package. A grade value of two means that the sensor node receiving this grade-creating package sends data to the sink node requires two hop count. The sensor nodes "d", "e" and "f" receive grade-creating packages with a grade vale of two from nodes "b" and "c". Sensor nodes "d" ,"e" and "f" increase the grade value of the grade-creating package to three and broadcast the modified package again. Sensor nodes "b" ,"g" and "h" increase receive this grade-creating package, but node "b" discards the package because the grade value of this package is three, which is higher than the recording value in the grade table for node "b". Moreover, if many sensor nodes simultaneously broadcast grade-creating packages with the same grade value,the sensor nodes receive and record the packages in their respective grade tables as back-up nodes. Thus, node "h" recording nodes "d" and "f" in the grade table.

Finally, sensor nodes "g" and "h" increase the grade value of the grade-creating package to four and broadcast the grade-creating package. But the sensor nodes discard the package because the grade value of the sensor nodes surrounding nodes "g" and "h" are less than four. The process of gradediffusion is thus complete, as shown in Fig .1

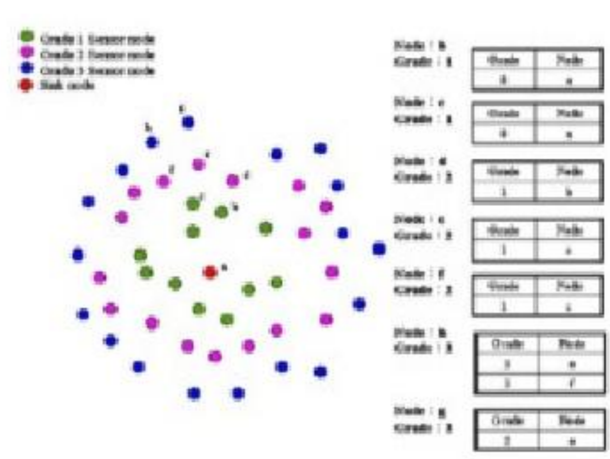

Fig1.*Graded Diffusion Process*

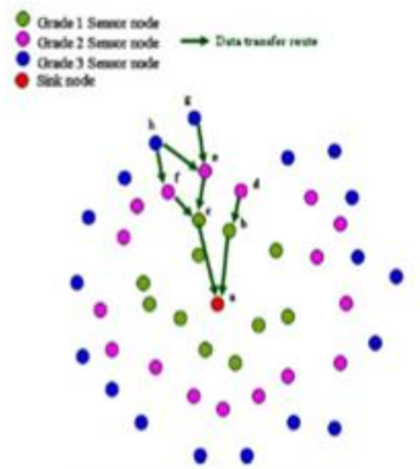

Fig2.*Data Transfer Route of Sensor Nodes*

The data transmission takes place from higher grade node to lower grade node as shown in Fig.2. Each sensor node records the grade value in the grade table when the grade diffusion algorithm is applied. The sensor node can record more than one node as relay nodes in the grade table when receiving the grade-creating package with a grade value less than itself.

3.2. B^{th} Calculation

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.

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

$$T_i \;\; = 1 \;, \;\; \frac{N_i^{now}}{N_i^{original}} < \beta$$

$$= 0 \,, \;\; \text{otherwise} \tag{1}$$

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 parameterβ 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.

3.3. Genetic Algorithm

This paper implements an algorithm for WSNs based on the grade diffusion algorithm combined with the genetic algorithm. Genetic algorithms are inspired by Darwin's theory about evolution. Solution to a problem solved by genetic algorithms is evolved[2].

Algorithm is started with a set of solutions (represented by chromosomes) called population. Solutions from one population are taken and used to form a new population. This is motivated by a hope, that the new population will be better than the old one. Solutions which are selected to form new solutions (offspring) are selected according to their fitness - the more suitable they are the more chances they have to reproduce. This is repeated until some condition (for example number of populations or improvement of the best solution) is satisfied.

The 5 steps in Genetic Algorithm are:

3.3.1. 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.

3.3.2. Evaluation

The genes of the chromosome give us the information about the replacement of the node. The goal is also to reuse the most routing paths and to replace the fewest sensor nodes. In general all these factors need to be considered while calculating the fittness function. The fitness value is calculated according to a fitness function, and the parameters of the fitness function are the chromosomes genes. The fitness function is calculated by the following formula:

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

Ni = the number of replaced sensor nodes and their grade value at i

Pi = the number of reusable 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

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.

3.3.3. Selection

The selection step will eliminate the chromosomes with the lowest fitness values and retain the rest. We use the elitism strategy. In elitism strategy, when creating new population by crossover and mutation, we have a big chance, that we will loose the best chromosome.

Elitism is name of method, which first copies the best chromosome (or a few best chromosomes) to new population. The rest is done in classical way. Elitism can very rapidly increase performance of GA, because it prevents losing the best found solution.

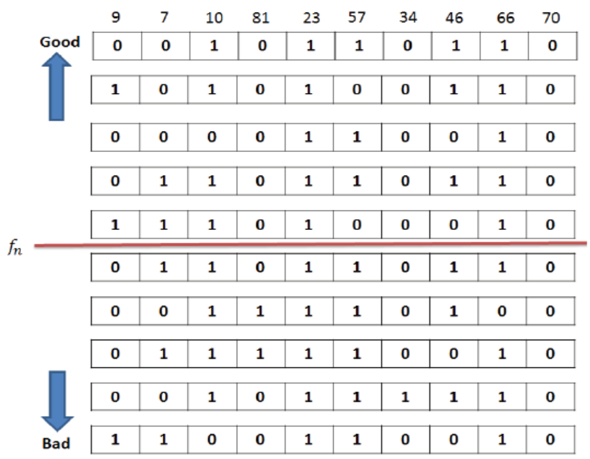

Fig3. *Selection Step*

Using this strategy we keep the half of the chromosomes with better fitness values and put them in the mating pool. The worst chromosomes will be deleted, and new chromosomes will be made to replace them after the crossover step.

3.3.4. Crossover

The crossover step is used in the genetic algorithm to change the individual chromosome. In this algorithm, we use the one-point crossover strategy to create new chromosomes.

One-point crossover - one crossover point is selected, binary string from beginning of chromosome to the crossover point is copied from one parent, the rest is copied from the second parent.

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.

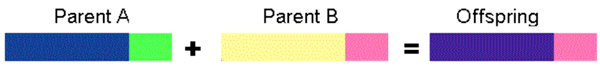

Fig4. *One-point crossover*

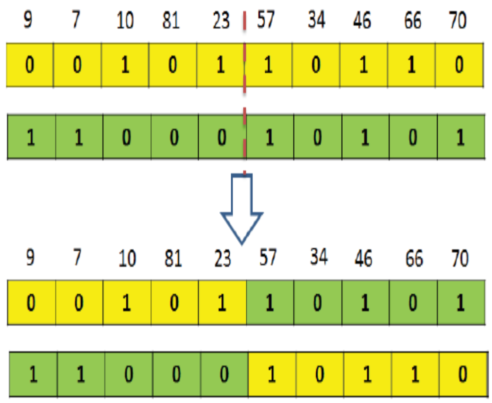

Fig5. C*rossover step*

3.3.5. Mutation

In this algorithm, we simply flip a gene randomly in the chromosome i.e. we do bit inversion.

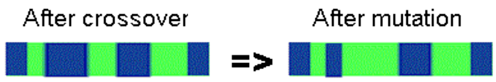

Fig6. *Mutation*

The chromosome with the best fitness value is the solution after the iteration. The mutation step can introduce traits not found in the original individuals and prevents the GA from converging too fast. The proposed algorithm will replace the sensor nodes in the chromosome with genes of 1 to extend the WSN lifetime.

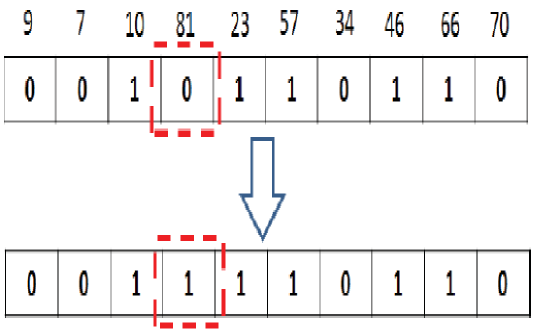

Fig7. *Mutation step*

4. Tool Used

Simulation of the proposed algorithm will be performed with the help of Network Simulator-2(NS-2). It is an open-source tool and can work on any platform. It is event driven packet level network simulator. It consists of two languages: C++ and OTcl. OTcl acts as the frontend (i.e., user interface), C++ acts as the backend running the actual simulation.

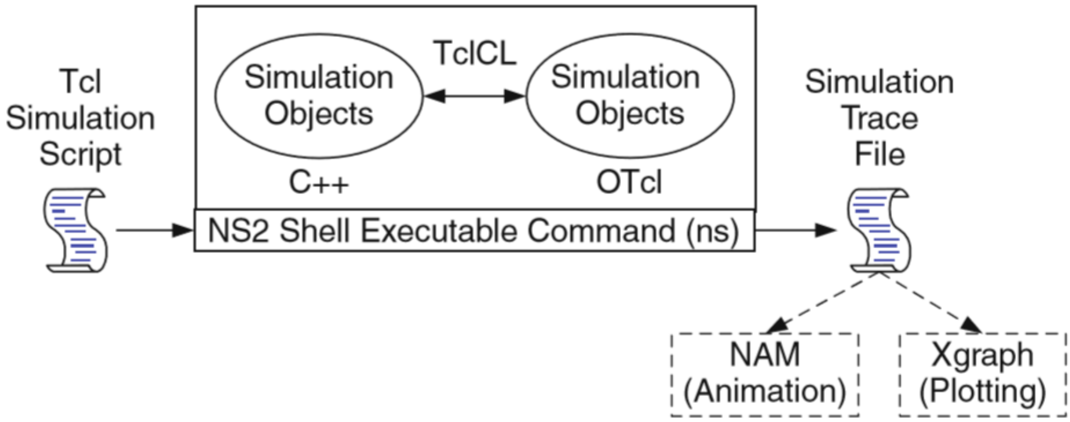

Fig8. *Basic architecture of NS*

The output will be shown by animation(NAM file) and graph(Xgraph). The simulation results will show how the faulty sensor nodes are recovered by using most reused paths and these results are compared with existing models.

5. Conclusion

In real wireless sensor networks, the sensor nodes 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 cost.

Our system proposes a faulty node recovery and replacement algorithm for WSN based on the grade diffusion algorithm combined with 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 reduces the rate of energy consumption.

References

[1] S. Corson and J. Macker, Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. New York, NY, USA: ACM, 1999.

[2] M. Gen and R. Cheng, Genetic Algorithms and Engineering Design. New York, NY, USA: Wiley, 1997.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] H. C. Shih, J. H. Ho, B. Y. Liao, and J. S. Pan, Grade Diffusion Algorithm, Proceedings of the Information Technology and Application in Outlying Islands (2011).

AUTHORS’ BIOGRAPHY

Chaitrali Brahme is a student pursuing her B.E Degree under Department of Computer Engineering from University of Pune. She is presently working on mathematical module of the framework and mathematical implementation of sub algorithms.

Srushti Gadadare is a student pursuing her B.E Degree under Department of Computer Engineering from University of Pune. She is currently involved in system developing activities for developing algorithm for framework and designing GUI.

Rutushree Kulkarni is a student pursuing her B.E Degree under Department of Computer Engineering from University of Pune. She is currently involved in system developing activities concerned with the algorithms and designing GUI of the framework.

Pranjal Surana is a student pursuing her B.E Degree under Department of Computer Engineering from University of Pune. She is working on designing GUI & the transitions involved in model of framework.

Prof. Ms. M. V. Marathe is an assistant professor under department of Computer Engineering. She is having 12+ years’ experience in the field of teaching as well as research. Her research interests include domains like Web mining and NLP.