

Novel Approach for Fault Detection in Wireless Sensor Network

Kavita Tandon¹, Neelima Mallela², Nishi Yadav³

^{1,2}B.Tech VIII Semester, Computer Science & Engineering,
Central University, ITGGV, Bilaspur, Chhattisgarh, INDIA

³Assistant Professor, Computer Science & Engineering,
Central University, ITGGV, Bilaspur, Chhattisgarh INDIA

Abstract -The Wireless Sensor Network is the wireless autonomous network with a large number of distributed sensor nodes. The nodes in WSN are small in size hence very much portable. These sensor nodes are mainly used to monitor environmental conditions. They collect information from its surroundings and also from each neighbour. Then whole data are sent to main location for its analysis. This cooperative approach of message transmission has made sensor network much reliable. But if they receive any wrong information from their neighbours, it makes them faulty. For the whole system's performance, it is very important to maintain the accuracy of data. Hence detecting faulty nodes in sensor network is a measure issue in network management. If no central system is there to monitor the fault issues, sensor nodes have to implement their own ways to detect fault. This paper focuses on the implementation of fault detection feature. We have proposed fault detection algorithm for diagnosing the hardware fault and implemented the algorithm which is having better performance over the existing algorithm.

Keywords-Wireless Sensor Network, Fault Detection, Fault Diagnosis.

1. INTRODUCTION

Wireless sensor networks rely on the information collected by the nodes individually. If any node is faulty, it may propagate in the network and cause severe damage to the meaning of data. So, it's very essential to find out the faulty node immediately. Fault can be detected based on the comparisons between neighbouring nodes and dissemination of the decision made at each node [1].

Unfortunately, nodes may have to find out the faulty node by their own, if there is no centralized system to find out, while maintaining their performance as well. Also, even small disruptions or problems in collected data must be addressed quickly, as the overall quantity of data gathered is small relative to long term deployments [2]. Security is also the measure concern in the sensor networks due to wireless channel access. The topology is mostly unknown before deployment. Also, there is no protection after deployment. Hence the network is very much vulnerable to security attacks and it's very difficult to provide them security because these are deployed in hazardous environments. Besides security, limited battery life is the measure concern in these networks [3].

Wireless Sensor Network has been used in many critical areas like habitat monitoring [5], scientific exploration [6], and infrastructure protection [7]. The readings of individual node are crucial in these applications, e.g., in a surveillance network [8], so these reading must be accurate to avoid false alarms and missed detections. Some applications may be fault tolerant to some extent but if their faulty nodes are replaced with good ones, this will significantly improve the performance of the network. Before applying any maintenance technique, we must find out the faulty nodes. And for this it's essential to investigate methods for detecting faulty nodes [4].

2. LITERATURE REVIEW

Wireless sensor networks are the means of crucial information gathering. Each node participating in the network has to collect information as well as has to detect faulty nodes, if any. Wireless networks are always very interesting and challenging area of work for the researchers. Several works have been done in wireless sensor networks and many are going on. Each research must pass through simulation so that the actual effect of it on the performance of network may be analyzed prior to its implementation on the real network. In 2009, Shuo Guo, Zigu Zhong and Tian He et al [4] proposed a thesis FIND, which detects faulty nodes based on their relative sensing results, i.e. node sequences. They stated a theory that average ranking differences of nodes in detected sequence and original sequence can be used as an effective indicator for faulty nodes. Based on the theoretical study, they developed a detection algorithm to obtain a blacklist when an accurate defective rate is unavailable.

In [9], Bertrand-Krajewski et al developed a set of seven criteria derived from physical processes underlying the data and measurement system to determine validity of the data. These criteria are: status of sensor, physical range, and locally realistic range, duration since last maintenance, signal gradient, material redundancy, and analytical redundancy. If a data point fails a single criterion, the point is considered unreliable. The authors advocate a significant initial effort to gather data and select parameters for the criteria. The node which behaves differently from other nodes may be a node with faults. Outlier detection is a

conventional method for identifying readings that depart from the norm. For example, Ding [10] proposes detecting faulty nodes by determining if the difference between a nodes reading and its neighbor's is above a threshold. However, its correctness is based on the assumption that neighboring nodes have similar readings. For many phenomena of interest (e.g., thermal radiation and acoustic signals) in sensor networks, such an assumption does not hold, because these signals attenuate over space.

In [13], they proposed a general approach to fault diagnosis that is widely applicable and only needs a limited number of connections among units. The algorithm uses a majority vote among the neighbors of a unit to determine the status of the unit. In [14], they say that sensor nodes with permanent faults and without fault which was judged as faulty are identified with high accuracy for a wide range of fault rate and keep false alarm rate for different levels of sensor fault model and also correct nodes are identified by accuracy.

3. FAULT MANAGEMENT IN SENSOR NETWORKS

Basically the whole fault management procedure can be categorized in two phases: Fault detection, and Fault recovery. In Fault detection phase, the faulty nodes are found. If the battery life of the node is over, it can inform the cell manager, if any, by itself. Otherwise cell manager can also initiate the procedure to find out the active nodes in the network by sending "get messages" in regular intervals. If the node replies with regular update, it means it's not in sleep mode. But if the manager doesn't receive the update message from any node, then it sends an instant message to that node to inquire its status. If manager does not receive any acknowledgement message, it declares the node as faulty. There is a group manager above cell manager level which handles several cell managers. It detects faults in cell managers in same manner. If the node itself identifies about its 'low battery' status and informs other nodes in network, the fault detection technique is known as self detection. If the status of node is identified by other nodes, e.g. cell manager or group manager, it is known as active detection.

If the fault has been detected, it should be immediately replaced with fault free nodes. For example sleeping nodes can be awaked or mobile nodes can replace the faulty nodes. A cell manager should always have a secondary cell manager to act as a backup. If the battery of cell manager itself is down, it notifies of its low battery status to all the nodes in its network. If the cell manager has exhausted its battery life, then the secondary cell manager is treated as the new cell manager. The existing node becomes a common node and the new cell manager will choose a secondary cell manager for itself. When a group manager detects the sudden death of a cell manager, it then informs the cell members of that faulty cell manager (including the secondary cell manager). This is an indication for the secondary cell manager to start acting as a new cell manager. The same case is also applied to the group manager. It also maintains a backup of group manager node within the group to replace it when required. If the group

manager residual energy drops below the threshold value, it may downgrade itself to a common node or enter into a sleep mode, and notify its backup node to replace it. This information is propagated to its cell managers and also to neighboring group managers. If the backup node does not have enough energy to replace the group manager, cell managers within a group co-ordinate to appoint a new group manager for themselves based on residual energy.

Each cell in the network maintains their health in terms of energy. They are low energy, medium energy and high energy. By analyzing the health status of nodes in network, group manager can predict the future faults and may take actions to prevent it.

4. PROPOSED WORK

In this paper, we have proposed the novel algorithm for implementation of fault detection technique for detecting hardware faults in wireless sensor network. For implementing this, we have used Visual Basic .Net as a simulation tool. We assume that the sensor nodes are randomly deployed and the transmission range of all the nodes is same. Also each node has at least three neighbouring nodes. This can be easily implemented by using a large number of sensor nodes in a given area. Faults can occur at different levels of the sensor network [11], such as system software, hardware, physical layer, and middleware. In our paper, the hardware layer fault is diagnosed and others are assumed as fault tolerant. Even in hardware, we have many categories like storage subsystem, computation engine power supply infrastructure, sensors and actuators. Sensors are actuators and they are placed in the open environmental conditions, so they are more prone to failure. However in this paper, we are going to diagnose only about sensor nodes.

If the sensor nodes are in the same transmission range, they are considered as neighbours. We are using [12] as base of our research and modifying their work. So, first we have diagnosed their work and then propose ours. Hence we can define our parameters as below:

- n: number of sensors;
- k : number of neighbours;
- p: probability of fault in a sensor;
- s : set of the sensors;
- $N(s_i)$: set of the neighbors of i th set;
- X_i : s_i 's measurement ;
- s_j : neighbour node of s_i
- $d^{t_{ij}}$: at a given time, difference between X_i and X_j ;
- $d^{t_{ij}} = X_{t_i} - X_{t_j}$
- $\Delta t_i = (t_i + 1) - (t_i)$;
- $\Delta d^{t_{ij}}$: difference between s_i and s_j from (t_i) to $(t_i + 1)$
- $\Delta d^{\Delta t_{ij}} = d^{(t_i + 1)_{ij}} - d^{t_{ij}} = (x_{t_i + 1}^{(t_i + 1)_{ij}} - x_{t_i}^{(t_i + 1)_{ij}}) - (x_{t_i}^{t_{ij}} - x_{t_i}^{t_{ij}})$
- C_{ij} : test between s_i and s_j , $c_{ij} = \{0, 1\}$
- \emptyset_1 and \emptyset_2 represent threshold values
- T_i : tendency value of a sensor, $T_i = \{LG, LF, GD, FT\}$

At first, we are analyzing the existing algorithm [12]. Based on that algorithm, we have proposed our work. So, the existing algorithm can be given as:

**TABLE.1
EXISTING ALGORITHM**

Steps	Contents
STEP 1	Each sensor s_i , set $c_{ij}=0$ and compute d^{ij} If $ d^{ij} > \emptyset_1$ then Calculate $\Delta d^{\Delta ij}$ If $ \Delta d^{\Delta ij} > \emptyset_2$ then $c_{ij}=1$
STEP 2	If $\sum s_j \in N(s_i) \ c_{ij} < N(s_i)/2 $ Where $N(s_i)$ is the number of the s_i 's neighbouring nodes then $T_i=LG$ Else $T_i=LF$ Communicate T_i to neighbours
STEP 3	If $\sum s_j \in N(s_i)$ and $T_j=LG$, $(1-2*c_{ij}) > N(s_i)/2 $ Then $T_i=GD$ Communicate T_i to neighbours
STEP 4	For $i=1$ to n If $T_i=LG$ or $T_i=LF$ then If $T_j=GD$ (FOR ALL) $s_j \in N(s_i)$ then If $c_{ij}=0$ then $T_i=GD$ Else $T_i=FT$ Else repeat Communicate T_i to neighbors
STEP 5	For each s_i , if $T_j=T_h=GD$ As $s_j, s_h \in N(s_i)$, where $j \neq h$ And if $c_{ji} \neq c_{hi}$ then If $T_i=LG$ (or LF) then $T_i=GD$ (or FT)

A test result c_{ij} is generated by sensor s_i based on its neighbour's s_j measurements using two variables d^{ij} and $d^{\Delta ij}$, and two predefined threshold value \emptyset_1 and \emptyset_2 . If a sensor is faulty, it can generate arbitrary measurements. If c_{ij} is 0, most likely either both are good or both are faulty. Otherwise, if c_{ij} is 1, and are most likely in different status. Sensors can be either LG or LF, determined by using test value from its neighboring sensors. Each sensor sends its tendency value to all its neighbors. The number of the LG sensors with coincident test results determines whether the sensors are GD or FT. If a GD sensor is found in the network, its test result can be used to diagnose other sensor's status. The information can be propagated through the whole network to diagnose all other sensors as good or faulty. If the diagnosis is consistent with the test results, the diagnosis is valid. If there is no sensor being diagnosed, all its neighbors are either not diagnosed or are diagnosed as faulty.

In this paper, we have implemented the proposed algorithm which minimizes the execution time for detecting number of hardware faults in the same scenario as assumed in the

existing algorithm [12] and hence improves the performance in comparison with the existing algorithm [12]. In place of step 4 in the existing algorithm, we used the new concept that if there are no neighbors of s_i whose initial detection status is LG, and if the initial detection status T_i of s_i is LG, then set the status of s_i as normal (GD), otherwise as fault (FT). Also, we checked in step 5 whether detection of status of all nodes in network is completed or not .if it has been completed, and then exit otherwise, repeat the steps (1), (2) ,(3) and (4).

5. RESULT

From the Fig.1 and TABLE.1 given below, it can be clearly shown that proposed algorithm detects number of hardware faults in lesser time than existing algorithm which improves the performance of proposed algorithm as compared to existing algorithm [12].

As per the simulation result, TABLE-1 and Fig.1 represents the comparison between existing and proposed algorithm. Fig.1 shows a graph in which green line represents the performance of existing algorithm, whereas blue line represents the performance of proposed algorithm. The number of nodes is varying from 50 to 210 and the time taken for each case has been estimated in microseconds. Finally the graph has been plotted to show the difference in time taken by the existing and proposed algorithm.

TABLE.2

Number of nodes Vs Time elapsed in existing and proposed algorithm

S.No	No.of nodes	TIME TAKEN(microsec)	
		Existing Algorithm	Proposed Algorithm
1	50	80	50
2	70	85	55
3	90	105	75
4	110	100	78
5	130	115	80
6	170	120	85
7	190	125	95
8	210	145	110

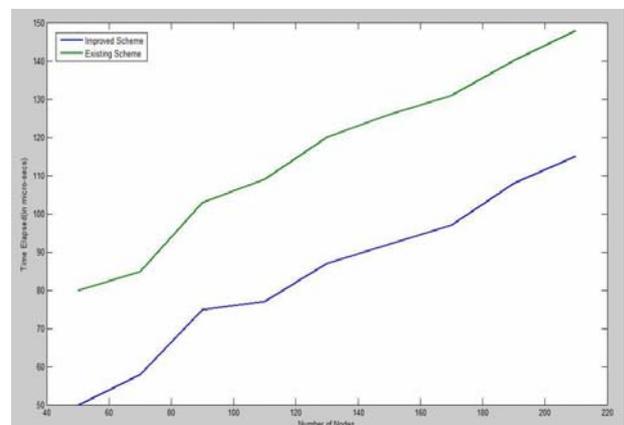


Fig. 1 Comparison of Number of nodes Vs Time Elapsed

6. CONCLUSION AND FUTURE WORK

In above graph Fig.1, we can see that our proposed algorithm is giving better performance than existing algorithm [12]. In this paper, we have implemented a distributed localized faulty sensor detection algorithm with improved performance. Here each sensor identifies its own status to be either "good" or "faulty" and the claim is then supported or reverted by its neighbors as they also evaluate the node behavior. In future, with respect to other parameters we will compare the proposed algorithm with the existing algorithm and detection accuracy would be determined. Detection accuracy means the ratio of the number of faulty sensors detected to the total number of faulty sensors in the network. We can also implement the method to find out the time consumed by this proposed approach to find out the faulty node i.e. performance of our approach. It will be very helpful in optimizing the overall fault management process and improving the performance.

ACKNOWLEDGMENT

We sincerely acknowledge the commendable initiatives taken by our research advisor, Mrs. Nishi Yadav for her guidance to complete this paper. We would also like to thank all the staffs of our department especially to head of the department, Dr. Manish Shrivastav, to provide us the necessary resources for this research.

REFERENCES

- [1] Myeong-Hyeon Lee, Yoon-Hwa Choi, "Fault detection of wireless sensor networks", *Computer Communications*, Volume 31, Issue 14, 5 September 2008, Pages 3469-3475, ISSN 0140-3.
- [2] N. Ramanathan, L. Balzano, M. Burt, D. Estrin, T. Harmon, C. Harvey, J. Jay, E. Kohler, S. Rothenberg, M. Srivastava, "Rapid Deployment with Confidence: Calibration and Fault Detection in Environmental Sensor Networks." In Technical Report CENS-TR-62, Center for Embedded Networked Sensing, 2006.
- [3] M. Yu, H. Mokhtar, M. Merabti, "A survey on Fault Management in wireless sensor network," in *Proceedings of the 8th Annual PostGraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting* Liverpool, UK, 2007.
- [4] Shuo Guo, Ziguo Zhong, Tian He, "FIND: Faulty Node Detection for Wireless Sensor Networks", NSF grants CNS- 0626609, CNS-0626614 and CNS-0917097.
- [5] R. Szcwcyk, A. Mainwaring, J. Anderson, and D. Culler, "An Analysis of a Large Scale Habit Monitoring Application" In *SenSys'04*, 2004.
- [6] G. Tolle, J. Polastre, R. Szcwcyk, N. Turner, K. Tu, S. Burgess, D. Gay, P. Buonadonna, W. Hong, T. Dawson, D. Culler, "A Macroscopic in the Redwoods", In *SenSys'05*, November 2005.
- [7] N. Xu, S. Rangwala, K. K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, D. Estrin, "A Wireless Sensor Network for Structural Monitoring", In *SenSys'04*, 2004.
- [8] T. He and et. al. *VigilNet*, "An Integrated Sensor Network System for Energy-Efficient Surveillance", *ACM Transactions on Sensor Networks*, Vol. 2(No. 1):Page 1 – 38, February 2006.
- [9] J. Bertrand-Krajewski, J. Bardin, M. Mourad, Y. Beranger, "Accounting for sensor calibration, data validation, measurement and sampling uncertainties in monitoring urban drainage systems", *Water Science and Technology*, pp. 95–102, 2003.
- [10] M. Ding, D. Chen, K. Xing, X. Cheng, "Localized Fault-Tolerant Event Boundary Detection in Sensor Networks", In *INFOCOM*, 2005.
- [11] B. Krishnamachari, S. Iyengar, "Distributed bayesian algorithms for fault tolerant event region detection in wireless sensor networks", *IEEE Transactions on Computers*, 53(3):241–250, March 2004.
- [12] Jinran Chen, Shubha Kher, Arun Somani "Distributed Fault Detection of Wireless Sensor Networks", *Dependable Computing and Networking Lab Iowa State University, Ames*.
- [13] D. Blough, S. Sullivan, G. Masson, "Fault diagnosis for sparsely interconnected multiprocessor systems", In *Proc. of FTCS-19*, 1989, pp.62-69.
- [14] Ayasha Siddiqua, Shikha Swaroop, Prashant Krishan, Sandip Mandal, "Distance Based Fault detection in wireless sensor network", *International Journal on Computer Science and Engineering (IJCSSE)*, Vol. 5 No. 05 May 2013, pp.368-375.