

MDFD and DFD Methods to detect Failed Sensor Nodes in Wireless Sensor Network

Mustafa Khalid Mezaal
Researcher
Electrical Engineering Department
University of Baghdad, Baghdad, Iraq

Dheyaa Jasim Kadhim
Assistant Professor
Electrical Engineering Department
University of Baghdad, Baghdad, Iraq

ABSTRACT

Wireless Sensor Network (WSN) contains many low cost and low power sensor nodes (SNs), these nodes may fail to communicate with each other according to some reasons such as battery lifetime or uncontrolled events which will lead to partition the network and reduce the Quality of Service (QoS) as well as the reliability and efficiency of the whole network. The motivation of this paper is detecting these malfunctions using Distributed Fault Detection (DFD) method considered with random proposed network model. Then a modification on DFD method (MDFD) proposed to enhance the efficiency and the reliability of the whole network and handling the error occurred with DFD method. The two methods analyzed and tested using MATLAB[®] and they must applied with homogeneous WSNs only that contain only one type of sensors, percentage error of DFD method was about 25% (for three SNs) due to its algorithm limitations in using only half of the neighbor SNs, this percentage error reduced in MDFD method in which all neighbor SNs considered to detect the failed SN reaching full detection accuracy but with latency tradeoff.

General Terms

Distributed Fault Detection, Proposed Method, Failed Sensor Nodes.

Keywords

Wireless Sensor Network, Malfunction, Node Failure, Quality of Service.

1. INTRODUCTION

WSNs are consisting of many small and low-cost SNs that form in a self-organized, multi-hop and monitoring network; sensors cooperative with each other, collecting data from the physical medium covered by the sensors and analyzing these data then transferring data through the network reaching to the main SN or to the sink node [1].

The great development in sensors manufacturing, microcontrollers, and communication technologies increased the ability of constructing a real WSN consists of many SNs, for that the QoS of WSN increased and measurement accuracy of various parameters in the field has been increased [4]. This increasing in using many SNs in one WSN, increased the SN failure probability, or increased the malfunction events at SNs, this decreased the QoS, so network portion and data transfer failure probabilities increased, so these failure SNs decreased the reliability and efficiency of the entire WSN, then it is important to find methods to detect such failures [7].

Many SNs are often deployed in uncontrollable and hostile environments. Therefore, failure in SNs can occur more easily

than in other systems; and the applications of WSNs are being widened. WSNs are also deployed in some occasions such as monitoring of nuclear reactor where high security and accuracy is required, fault detection for SNs in this specified application is of great importance [4]. SNs are usually battery-powered and the energy is limited, so it is common for faults to occur due to battery depletion. So it is troublesome and impractical to manually examine whether the SNs are functioning normally; or correct information cannot be obtained by the control center because failed nodes would produce erroneous data. Moreover, it may result in collapse of the whole network in serious cases [4].

SN status in any WSN can be divided into two types: normal and faulty. Normal, when SN already worked as its specified application; and Faulty in turn can be “permanent fault” or “static fault”. The so-called “permanent fault” means failed nodes will remain faulty until they are replaced, and the so-called “static fault” means new faults will not generate during fault detection [5].

Other proposed classification of fault SNs in WSNs can be divided into two categories: hard and soft. The so-called “hard fault” is when a SN cannot communicate with other nodes because of the failure of a certain module (e.g., communication failure due to the failure of the communication module, being out of the communication range of entire mobile network because of the nodes’ mobility and energy depletion of node); The so-called “soft fault” means the failed nodes can continue to work and communicate with other nodes (hardware and software of communication module are normal), but the data sensed or transmitted is not correct [5].

Whenever the use of large numbers of SNs in WSN increased the fault events and malfunction occurrence of SNs for different reason, these decreased the reliability and efficiency, as well as the QoS of the whole WSN will be decreased, for that it is important to detect this failure and handle it. Failure may occur in WSN due to uncontrolled environment, battery related problem, or failure in communication module. Failure detection is essential because failed or malfunctioning SN may produce incorrect analysis or detection of parameter. Manually checking of such failed SN in WSN is troublesome. To achieve the good quality of WSN through accuracy, reliability and efficiency, detection of SN failure or malfunctioning is essential [5].

Different methods used to handle faulty SNs depends on data of neighbor SNs to decide whether the current SN is Normal or Faulty SN, localization methods used this concept in its calculations to localize the faulty SN such as ToA [10], RSSI [2], and AoA [3] methods that depends on time delay, received signal strength and direction of arrival respectively

all that estimation to calculate a confidence factor to detect faulty SN. Node failure detection in localization methods depends on two kinds of SNs: normal SNs that distributed through the WSN with unknown positions, and ANs that usually distributed in the center and border parts of any WSN with pre-known positions in order to use their locations in localization of normal SNs [10].

Other used algorithms to detect the faulty SNs such as Round Trip Delay (RTD) method which used to detect the faulty SNs or malfunctioning with the help of confidence factors. Confidence factor of round trip path in network is estimated by using the round trip time. This method detected the failure in SN for symmetrical network conditions. In this way it helps to detect failed or malfunctioning sensor, which can be used to get correct data in WSN or the exact SN can be repaired or working status (health) of the WSN can be checked [9].

Another proposed algorithm to identify the faulty SNs which was DFD method depended on the number of success neighbor SNs to decide whether the current SN was success or faulty SN. But such methods has some shortcomings which the fault detection accuracy will decrease rapidly in the case of the number of neighbor nodes to be diagnosed are all small and the node's failure ratio is high [6].

1.1 DFD Method

DFD node fault detection determines the status of SN by testing it among neighbor SN mutually, For two neighbor SNs S_i and S_j , a test result C_i and C_j is produced by the data (such as temperature) sensed by each of them. The data at the moment t should be very close to each other because they are near, and the difference d_{ij}^t between this data should not exceed a certain threshold θ_1 [8].

Besides, the next moment $t+1$, the difference of the data of the two neighbor nodes is d_{ij}^{t+1} , and the difference of d_{ij}^t and d_{ij}^{t+1} is Δd_{ij}^t which should not exceed a certain threshold θ_2 . If one of these two conditions is not met, at least one of S_i and S_j is considered as a failure, and the test result $C_{ij}=1$, otherwise $C_{ij}=0$. For any node S_i , its test result with each node in $Neighbor(S_i)$ can be obtained. If there are more than $Num(Neighbor(S_i))/2$ nodes whose test results are 1 in $Neighbor(S_i)$, then the initial detection status T_i of node S_i is possibly failed SN (LT), otherwise, it may be possibly normal SN (LG) as in Equation 1:

$$\sum C_{ij} < Num(Neighbor(S_i))/2 \quad (1)$$

Constraints: $|d_{ij}^t| < \theta_1$ or $|\Delta d_{ij}^t| < \theta_2$

When the initial detection status of all nodes in the network is obtained, the following detection criterion is used for any node S_i : for the nodes in $Neighbor(S_i)$ whose initial detection status is LG, subtract the number of nodes whose test result with S_i is 0 from the number of nodes whose test result is 1. If the result is not less than $Num(Neighbor(S_i))/2$ then the status of S_i is normal, otherwise, the status of S_i is faulty, Figure 1a shows the flowchart of DFD method [8].

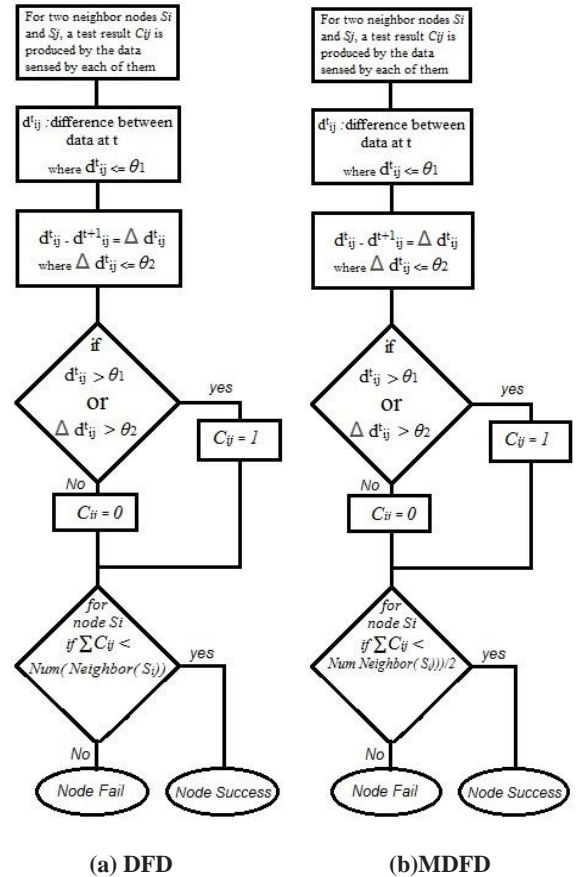
1.2 MDFD Method

From the realization of DFD node fault detection method, use all the DFD principles for the thresholds but for the neighbors instead of taking half of them and check for the fault, take all the neighbors nodes as in Equation 2:

$$\sum C_{ij} < Num(Neighbor(S_i)) \quad (2)$$

Constraints: $|d_{ij}^t| < \theta_1$ or $|\Delta d_{ij}^t| < \theta_2$

For any node S_i , its test result with each node in $Neighbor(S_i)$ can be obtained. If all neighbors of S_i $Num(Neighbor(S_i))$ nodes whose test results are 1 in $Neighbor(S_i)$, then the initial detection status T_i of node S_i is possibly faulty (LT), otherwise, it may be possibly normal (LG), Figure 1b shows also the mechanism of improved DFD method.



Figures 1. Flowcharts of DFD and MDFD methods

2. DFD AND MDFD SIMULATION RESULTS

SN failure detection in DFD method and its modification MDFD depended on the state of neighbor SNs whether they are approximately equal in sensing data or not, and the sensing data in the current time and the next time duration. The two methods analyzed and tested using MATLAB®.

2.1 DFD Method Results

In DFD method, the network tested with two models: three and four SNs random distributed, the results are acceptable for few SNs such as results in Table 1, but the percentage error for SN detection is increased whenever number of SNs is increased too such as results in Table 2, and this because of the DFD method did not considered the whole neighbor SNs in its calculations.

Table 1. DFD Method for Three SNs

C(SN ₀)	C(SN ₁)	C(SN ₂)	$\sum C_{01}$	$\sum C_{02}$	$\sum C$	State of SN ₀
0	0	0	0	0	0	Success
0	0	1	0	1	1	Success
0	1	0	1	0	1	Success
0	1	1	1	1	2	Success
1	0	0	1	1	2	Success
1	0	1	1	2	3	Fail
1	1	0	2	1	3	Fail
1	1	1	2	2	4	Fail
						%Error=25%

Table 2. DFD Method with Four SNs Network

C(SN ₀)	C(SN ₁)	C(SN ₂)	C(SN ₃)	$\sum C_{01}$	$\sum C_{02}$	$\sum C_{03}$	$\sum C$	State of SN ₀
0	0	0	0	0	0	0	0	Success
0	0	0	1	0	0	1	1	Success
0	0	1	0	0	1	0	1	Success
0	0	1	1	0	1	1	2	Success
0	1	0	0	1	0	0	1	Success
0	1	0	1	1	0	1	2	Success
0	1	1	0	1	1	0	2	Success
0	1	1	1	1	1	1	3	Fail
1	0	0	0	1	1	1	3	Fail
1	0	0	1	1	1	2	4	Fail
1	0	1	0	1	2	1	4	Fail
1	0	1	1	1	2	2	5	Fail
1	1	0	0	2	1	1	4	Fail
1	1	0	1	2	1	2	5	Fail
1	1	1	0	2	2	1	5	Fail
1	1	1	1	2	2	2	6	Fail
								%Error=50%

Table 3. MDFD Method with Four SNs Network

C(SN ₀)	C(SN ₁)	C(SN ₂)	C(SN ₃)	ΣC ₀₁	ΣC ₀₂	ΣC ₀₃	ΣC	State of SN ₀
0	0	0	0	0	0	0	0	Success
0	0	0	1	0	0	1	1	Success
0	0	1	0	0	1	0	1	Success
0	0	1	1	0	1	1	2	Success
0	1	0	0	1	0	0	1	Success
0	1	0	1	1	0	1	2	Success
0	1	1	0	1	1	0	2	Success
0	1	1	1	1	1	1	3	Success
1	0	0	0	1	1	1	3	Success
1	0	0	1	1	1	2	4	Success
1	0	1	0	1	2	1	4	Success
1	0	1	1	1	2	2	5	Success
1	1	0	0	2	1	1	4	Success
1	1	0	1	2	1	2	5	Success
1	1	1	0	2	2	1	5	Success
1	1	1	1	2	2	2	6	Fail

In Table 1, the number of SNs is three which are: SN₀, SN₁ and SN₂. So SN₀ had two neighbor SNs, and the number of neighbor SNs by two is (3/2=1.5 ceiled to 2). DFD method had a right decision to set SN₀ as a success SN for the first four cases, because whatever there is a neighbor SN had a data sensing much different than SN₀, but there is always a neighbor SN in some cases and still working had a zero C between it and SN₀. For the last case, DFD did a right decision to set SN₀ as Failure SN. There are two cases as mentioned with pink color (case 6 and 7) in Table 1 that still have neighbor SNs with zero C but DFD method considered SN₀ as failure SN according to DFD algorithm, whether SN₀ had to be a Success SN.

2.2 MDFD Method Results

In Modified DFD (MDFD), all neighbor SNs of the current nodes responsible for SN₀ state not only half of them, this simple modification did a wide different in SN detection but with little latency as a tradeoff instead of losing the whole data of the current SN.

Results of MDFD modification in Table 3 shows the improvement clearly for the same data of Table 2 which present a WSN consisted of four SNs: SN₀, SN₁, SN₂ and SN₃, once the cases in pink color (cases 8-15) improved from Failure SNs in Table 2 with DFD method to Success SNs with MDFD method as well as for more SNs. So, the MDFD

modification used the benefits of all neighbor SN checking instead of half neighbor SNs in spite of the minimum latency difference between this method and DFD method as in Figure 2.

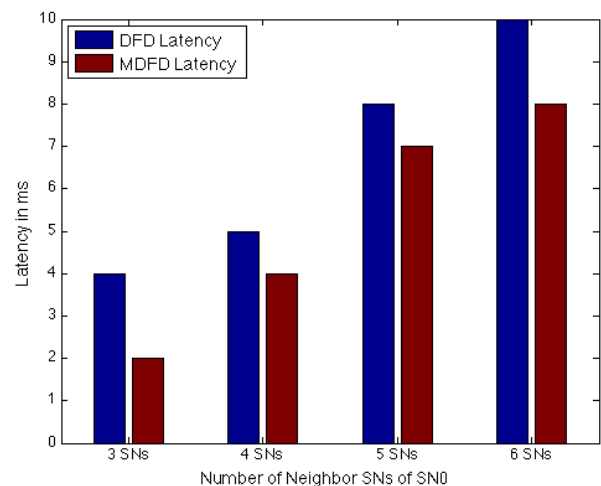


Figure 2. DFD and MDFD latency comparison.

3. CONCLUSION

DFD method success to detect the failed SN, but it had its own weak point that does not applicable with many SNs in a network, for that a new proposed named (MDFD) solved this problem and success to detect the failed SN that has many neighbor SNs with minimum latency as a tradeoff.

Only one reason made DFD and MDFD not applicable that if the WSN was not homogeneous i.e. the network consist of different sensor purposes and applications such as humidity and temperature sensors in the same network, that produces a large different between any two different sensors which make Cij unstable and unknown, and get undesired results.

4. REFERENCES

- [1] Dashwan, et al., 2006, Maximum Network Lifetime in Wireless Sensor Networks with Adjustable Sensing Ranges, 7th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD'06), IEEE, pp. 285-289.
- [2] E. Elnahrawy, X. Li, and R. P.Martin, 2014, The limits of localization using signal strength: a comparative study, Proceeding of Sensor and Ad Hoc Communications and Networks, IEEE, pp. 406–414, 2014.
- [3] G. D. Stefano, and A. Petricola, 2008, A Distributed AoA Based Localization Algorithm for Wireless Sensor Networks, Journal Of Computers, ACADEMY, Vol. 3, No. 4, pp. 1-8.
- [4] G. Singh, and A. Kaur, 2013, Evaluating Wireless Sensor Network on Quality of Services Using Mobile Sink Nodes, International Journal of Science and Research (IJSR), Vol. 3, No. 7, pp. 1209-1214.
- [5] G. Singh, and V.K. Sandhu, 2014, Enhanced Optimal Routing Leach Protocol Using Genetic Algorithm for Wireless Sensor Networks, International Journal of Science and Research (IJSR), Vol. 3, No. 8, pp.782-785.
- [6] J. Chen, S. Kher, and A. Somani, 2006, Distributed Fault Detection of Wireless Sensor Networks, workshop on Dependability Issues in Wireless Ad hoc Networks and Sensor networks (DIWANS), pp. 65-72.
- [7] L. Paradis, and Q. Han, 2007, A Survey of Fault Management in Wireless Sensor Networks, Journal of Network and Systems Management, Vol. 15, No.2, pp. 171-190.
- [8] P. Jiang, 2009, A New Method for Node Fault Detection in Wireless Sensor Networks, Sensors, Vol. 9, pp. 1282-1294.
- [9] R. N. Duche, and N. P. Sarwade, 2012, Round Trip Delay Time as a Linear Function of Distance between the Sensor Nodes in Wireless Sensor Network, International Journal of Engineering Sciences and Emerging Technology (IJESET), Vol. 1, No. 2, pp: 20-26.
- [10] S. Ravindra, and S. N. Jagadeesha, 2013, Time of Arrival Based Localization in Wireless Sensor Networks: A Linear Approach, Signal and Image Processing: An International Journal (SIPIJ), Vol. 4, No. 4, pp. 13-30.