jet INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2021, VOL. 67, NO. 1, PP. 87-93
Manuscript received April 18, 2020; revised January, 2021. DOI: 10.24425/ijet.2021.135948

Fault Tolerant Radiation Monitoring System Using Wireless Sensor and Actor Network in a Nuclear Facility

Mohamed Yehia Habash, Nabil Mohamed Abd Elfatah Ayad, and Abd Elhady Abd Elazim Ammar

Abstract—**In nuclear facilities, the reading of the sensors is very important in the assessments of the system state. The existence of an abnormal state could be caused by a failure in the sensor itself instead of a failure in the system. So, being unable to identify the main cause of the "abnormal state" and take proper actions may end in unnecessary shutdown for the nuclear facility that may have expensive economic consequences. That is why, it is extremely important for a supervision and control system to identify the case where the failure in the sensor is the main cause for the existence of an abnormal state. In this paper, a system based on a wireless sensor network is proposed to monitor the radiation levels around and inside a nuclear facility. A new approach for validating the sensor readings is proposed and investigated using the Castalia simulator.**

Keywords—**WSAN, Fault Tolerant Monitoring System, Sensor Readings Validation, WSN in Nuclear Facilities**

I. INTRODUCTION

IN recent , Wireless Sensor and Actor Networks (WSAN) have become one of the most important areas of research, due to its wide range of applications. Also, due to the ongoing research that led to the technological advances in multiple areas including memory, processor, radio, wireless communications, Micro Electro Mechanical Systems (MEMS) [1], integrated circuits, and low power consumption devices; it is easy to get small size, low cost, low power consumption sensor nodes. WSAN are wireless networks consisting of distributed sensor and actor devices at different locations. Sensors are used to monitor environmental conditions or physical phenomena, while actors are used to control the environment. Each sensor node is a battery-operated device that has a small size memory to store code and sensor data, processor to perform data analysis, and computations and an RF transceiver to communicate with other neighboring nodes. These sensor and actor nodes are coupled with the physical phenomena to be monitored. While each sensor node has limited processing power, sensing area, and energy, grouping a large number of sensor nodes lead to an accurate, robust, and reliable sensor network covering a larger area. These sensors cooperate with each other to collect the sensed data and send it to a sink node or a Base Station (BS) which in turn sends these data to a central processing and storage system. The location of these sensor nodes can be acquired through local positioning algorithm [2], or a Global Positioning System (GPS). Now, WSN is implemented inside the nuclear plants [3] as in Comanche Peak nuclear power plant (USA), they used wireless sensors to observe the vibration and temperature of motors and pumps in the secondary part of the nuclear plant. Also, in Farley nuclear power plant (USA), they used wireless radiation sensors to monitor and track the levels of radiation in and out the plant. The organization of the paper is as follows. Part two provides a survey of the related work. Part three includes the implementation of the proposed schemes for validating the sensor readings, whereas part four includes the simulation, and evaluation results for the proposed algorithm. Finally, part five concludes the whole work and presents the advantages of the proposed algorithm.

II. RELATED WORK

The sensor readings are important for both the supervision and control systems. The faulty sensor readings will make an adverse impact on the decisions taken for the services, maintenance and the operation of the nuclear facility. So, a Fault Detection and Identification (FDI) should be present to validate and detect the sensor's fault status as it affects the performance of the monitoring and control systems. Sensor fault detection and identification approaches [4], can be classified according to their architectures into, *centralized* and *distributed*. Some other researchers classified the problem into two different approaches: *physical* and *analytical* redundancy methods. The analytical redundancy requires advanced information processing techniques, such as state estimation, and various logical operations [5] which may be limited by the RAM size and the processing power of the wireless sensor network. The selection of the suitable technique for sensor data validation depends on the purpose of the application [6]. In [7], the authors proposed an algorithm to differentiate between the intermittent and the permanent sensor faults. It is based on that each sensor node broadcasts its reading value to its m neighbors. The decision is taken based on the result of the comparison between the node's reading xi and the neighbor's readings xj (j = 1, 2…m). The sensor node is identified to be fault free or normal if its reading matches with reading values of more than half of its neighbors. So, xi is similar to xj when $|xi - xj| < \text{threshold } \Delta$, where the Δ value is application dependent. They use a mechanism to identify hard faulty sensor nodes that is based on the timeout timer. The sensor node vi identifies sensor node $vj \in m$ as hard faulty, if the value of the sensor reading vj is not received within Tout. A

This work was supported by the Nuclear Research Center, Egyptian atomic energy authority, and Faculty of Engineering, Al azhar University.

Mohamed Yehia, Nabil M. A. Ayad are with Nuclear Research Center, Egyptian Atomic Energy Authority, Egypt (e-mail: mohammed.yehia_sds@yahoo.com, n_ayad51@yahoo.com).

Abd Elhady A. Ammar is with Electrical Engineering Dept., Faculty of Engineering, Al azhar University, Egypt (e-mail: hady42amar@gmail.com).

sensor node is identified as soft faulty (permanent) incase the sensor node fails multiple consecutive times. In [8], the authors used the weighted average value scheme to identify whether the node is faulty or not. A sensor node can identify itself as faulty or fault free through comparing the value of its reading with the neighbors' weighted average value$\overline{X}$. If the difference is less than the thresholdΔ , the measurement is regarded as right. Otherwise, the sensor node is considered as faulty. In [9], the authors proposed a two stage algorithm to identify the faulty sensor nodes. First stage starts when each sensor node finds the difference between its own reading and the readings of its m neighbors. Then, identifies itself as local good or local faulty depending on the number of matches or mismatches between its reading value and neighbors value respectively. Then, each sensor node sends its local status again to all its m neighbors. Second stage starts when receiving the local status of the neighbors. Only local good sensor nodes that have more than half of its neighbors with coincident test results are declared to be globally good. Finally, by using the globally good sensors, the status of their neighbors can be determined to be good or faulty if it matches or mismatches with the globally good sensor. In [10], the authors divided the sensor field into clusters with cluster heads, and cluster members. They assumed that, any sensor node knows the range of expected readings during the day, and has the ability to take a decision and identifies its own readings as faulty or good, where a "1" indicates a faulty reading. Then, the sensor nodes send their decisions to their cluster head. The cluster heads decide that there is an event using two thresholds. Once the decision of an event is made, each cluster head analyzes the readings of its members and updates their confidence levels. Once the confidence level reaches a predetermined value, the cluster head isolate the corresponding sensor nodes from the sensor network. In [11], the authors presented a centralized algorithm for the identification of faulty sensor nodes. An alert is generated and sent to the sink node by each sensor node when there is a mismatch between the node's reading value and its neighbors reading values. Then, the sink node analyzes these alerts and makes a graph in which an edge xy only is found if sensor node *x* and sensor node *y* are suspicious of each other. Then, a flag is assigned to each faulty or non-faulty sensor node such that, the assignment of these flags has no effect on the graph's consistency. Finally, only the sensor nodes that are flagged as faulty in these assignments are identified and labeled as faulty sensor nodes. In [12], the authors proposed a distributed algorithm in two stages. In the first stage each sensor node detects any suspicious behavior using the time correlation between its own readings, that means, each sensor node compares its own readings at different time steps (x(t), x(t-1), x(t-2)…) against a certain threshold. Then, in the second stage the suspected sensor nodes that have mismatched readings communicate with the neighboring nodes to confirm its status whether good or faulty.

III. THE PROPOSED FAULT DETECTION ALGORITHM

In Nuclear facilities, and some other complex systems, the supervision and control system takes the value of the sensor readings as an input to check, and analyze the system state, detect abnormality, and determine the main causes of the abnormal state in case of existence. That is in order to warn and guide the operator about the recommended actions to avoid any critical damage to the systems of the nuclear facility. The safety, reliability, and performance of a supervision and control system that uses the values of the sensor readings to take decisions depend on the accuracy, and reliability of the sensors. The existence of an abnormal state could be caused by a failure in the sensor itself instead of a failure in the system. So, it is important for a supervision and control system that gives the operator a full overview about the system states to be able to identify the main cause of the abnormal state. That is why, it is necessary to validate the sensor readings, detect sensor failures, and isolate faulty sensors from decision making process. These faults need to be identified and isolated before they have any harmful impacts on the nuclear facility. In this work, we developed a system that is based on wireless sensor and actor network (WSAN). It is basically used to monitor the radiation levels inside and around a nuclear facility. So, we assumed that sensor devices that are able to measure the radiation levels are located in defined positions throughout the nuclear facility. Our work is done in two main stages, the first one includes the sensor nodes deployment and organization into clusters with cluster-head, and cluster-members with one sink node in the whole network as shown in Fig. 1. This is done through the use of one of the clustering protocols, which is Low Energy Adaptive Clustering Hierarchy (LEACH) [13]. It includes distributed formation of clusters. LEACH algorithm chooses some sensors randomly as Cluster- Heads (CH). Then, this role is rotated in a certain way to distribute the energy consumption between the sensor nodes in the entire network. Then, the second stage includes the fault detection process which is implemented at the cluster heads. This step includes a new approach to validate the sensor readings and detect the faulty reading of the sensor nodes.

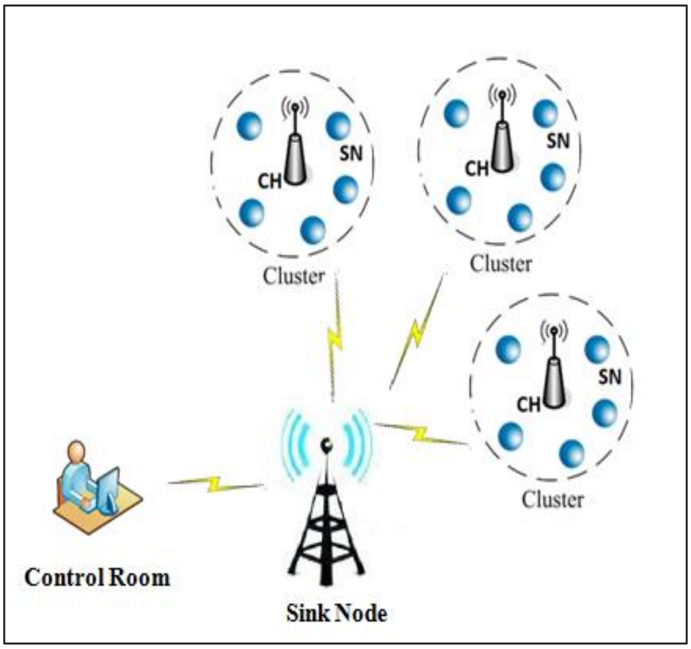


Fig. 1. Wireless Sensor Network Architecture of the Proposed System

A. DATA MODEL

It is assumed that *n* sensor nodes are regularly distributed in an area of $x * y$ squared meter. The reading of the sensor node $x_i(t)$ is considered as abnormal or faulty incase a mismatch between the value of the sensor reading and the measured value of the reference sensors X_{Ref} (t) is found. Sensor nodes with faulty readings are called faulty sensor nodes.

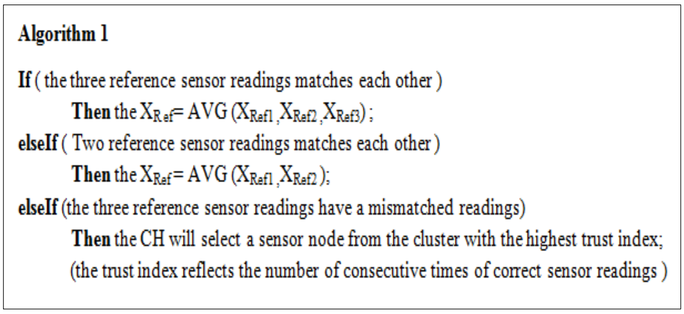
B. FAULT MODELS

It is assumed that, the proposed wireless sensor network does not have any abnormal sensor nodes (malicious nodes) that intentionally inject incorrect readings into the sensor network. In [14], they categorized data faults as constant, short, and noise. The above mentioned three faults have been acquired in the four data sets from different wireless sensor network deployments [14 - 16].

C. SENSOR READINGS VALIDATION PROCESS

Sensor validation can be considered part of the largest effort of improving the system reliability and safety. A system based on a WSN is proposed to monitor the radiation level inside and around a nuclear facility where the sensor nodes are positioned in selected fixed locations, and supplied with devices to measure the radiation. Sensors have a unique identifier in the network. After the sensor nodes are deployed, the network is organized into clusters with Cluster Head nodes (CH) and Cluster Member nodes (CM). Therefore, our sensor network composed of a sink node, group of clusters, and there are a number of sensor nodes per each cluster. The CM (Sensor node) sense the environment, then send the sensed values to their CH periodically every T seconds, which is appropriate for the phenomena being monitored. The cluster-heads receive the sensor readings from their sensor members, and put them in corresponding entry in the neighbor table. Then, they are responsible for detecting sensor faults by applying the fault detection algorithm. After that, they send the readings from all sensor nodes in their cluster combined with their status to the sink node. The sink node collects all these measurements inside the network, and relay it to the monitoring and control system to let the operator have a global overview about the network status for further analysis and control decisions. The Fault Detection algorithm is based on the idea that the sensors reading tends to be spatially correlated. This means that the sensor nodes that are adjacent to each other are expected to report correlated readings. By introducing reference nodes in each cluster, mainly three redundant nodes positioned at the same location for more accurate results. The same idea of Triple Modular Redundancy technique (TMR) [17], in which three systems perform a specific process, and by using a majority voting system to produce a trusted single output. If any one of the three systems fails, the remaining two systems can mask and correct the fault. Our fault detection is *two stages* and implemented only at the cluster-heads. In the first stage, the cluster-heads select one reference sensor from the three reference sensors to be used during fault detection process. The cluster-heads use the two out of three voting technique to select one good sensor as a reference; to check the status of other sensor nodes in the cluster.

The selection of one reference sensor has many cases and implemented as shown in Algorithm 1 in Fig. 2.



```
Algorithm 1
If ( the three reference sensor readings matches each other )
    Then the $X_{Ref} = AVG (X_{Ref1} , X_{Ref2} , X_{Ref3})$ ;
elseIf ( Two reference sensor readings matches each other )
    Then the $X_{Ref} = AVG (X_{Ref1} , X_{Ref2})$;
elseIf (the three reference sensor readings have a mismatched readings)
    Then the CH will select a sensor node from the cluster with the highest trust index;
    (the trust index reflects the number of consecutive times of correct sensor readings )
```

Fig. 2: Selection of good Reference Sensor Reading Cases

As seen in Fig 2, we have three cases. In the first case which is the normal case, the three reference sensors are in a good state or their readings match with each other (i.e. there are no sensor failures) so, the CH will use the average reading of the reference sensors, as a result of the majority voting technique.
In case 2 in which one reference sensor fails and produces a faulty reading, while the other two are working correctly and produce correct readings. The CH will use the average reading of the two matched reference sensors as a result of the majority voting technique. In case 3 in which the three reference sensors have mismatched readings. In this case, the CH will select a sensor node from its cluster with the highest trust index value to act as a reference sensor. The trust index reflects the number of consecutive times this sensor is diagnosed as a good. The CH will report to the supervision and monitoring system that the reference sensors need immediate maintenance and calibration. The CH will use the reading of the selected sensor to diagnose the rest of the sensor nodes in its cluster until the maintenance and calibration is done.

After the selection of the good reference sensor reading (**X_{Ref}**), the CH identifies that the sensor node is fault free if the reading's of the sensor node matches with the reading's of the reference sensor as shown in Fig. 3. That is, if $x_i(t) – x_{Ref}(t) <$ Threshold Δ , where Δ value depends on the *distance* between the sensor node and radiation source [18]. Otherwise, the reading is faulty. Then it updates both the node status and the trust index in the sensor node record which is maintained at the cluster head (CH).

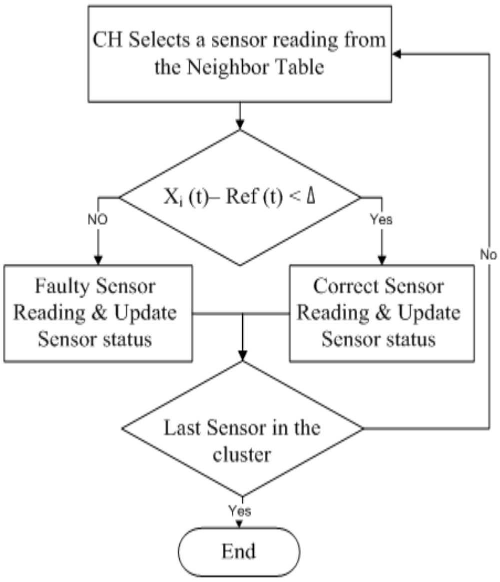


Fig. 3. The Flow Chart of the Proposed Algorithm

After the completion of the fault detection process, the CH sends the sensor readings combined with their status to the sink node, to relay it to the monitoring and control system. So, the operator can analyze the status of the systems, schedule maintenance and take corrective actions if required. For example, if the readings of the sensor node are identified as faulty multiple consecutive times, the CH will report that to the sink node which in turn will report that to the control system to take corrective actions if applicable. As the cluster-heads apply the fault detection step to every sensor reading, they can detect short and permanent faults. As seen, the proposed algorithm is a hybrid technique that is a distributed algorithm and uses the lowest possible physical redundancy of sensor nodes. Thus, having the advantages of using the physical redundancy, which gives robustness to the supervision and control system. And also, the advantages of the distributed algorithms which gives a low energy consumption. Another point, the detection accuracy of the proposed fault detection algorithm does not depend on the neighbor's accuracy as in the other fault detection schemes, but on the reference sensors accuracy which can be easily calibrated in periodic times as their numbers in the network are low. For other schemes to give accurate results, each sensor node should have more than half of the neighbors give accurate readings. So, more than half the sensor nodes should be calibrated which is difficult as the network size increases.

IV. SIMULATION AND PERFORMANCE EVALUATION

we created a simulation scenario to assess and test the overall performance of the suggested algorithm in terms of accuracy, communication overhead, and power consumption. we used the CASTALIA simulator, which is an event driven simulator for Wireless Sensor Networks (WSNs). It is used to test the distributed algorithms in a realistic wireless channel and radio models.

Table I, shows the different parameters used within the simulation scinareo including total number of nodes, communication parameters, total simulation time, and the area of interest where nodes were chosed to be distributed in fixed positions.

TABLE I
MAIN OPERATIONAL SETTINGS

Parameter	Value
Simulation Time (sec)	2000
Number of nodes	36
Area of Interest (m^2)	50 X 50
Carrier Frequency (GHz)	2.4
Node Initial Energy (J)	18720 (Two AA Batteries)

A. *Case Study*

As shown in Fig. 4, the sensor nodes are deployed in an area of 50 × 50 m2. They are fixed and are deployed in fixed position. Node 0, is configured to be the sink node to relay the sensor readings to the monitoring and control system, while nodes 9, 19, and 35 are selected to be the cluster-heads. The rest of nodes are organized to be cluster members. This arrangement is just for the simulation purpose and can be changed when requested. After receiving the advertisement from each cluster-head, each sensor node decides based on the advertisement message's strength to which cluster, it will join.

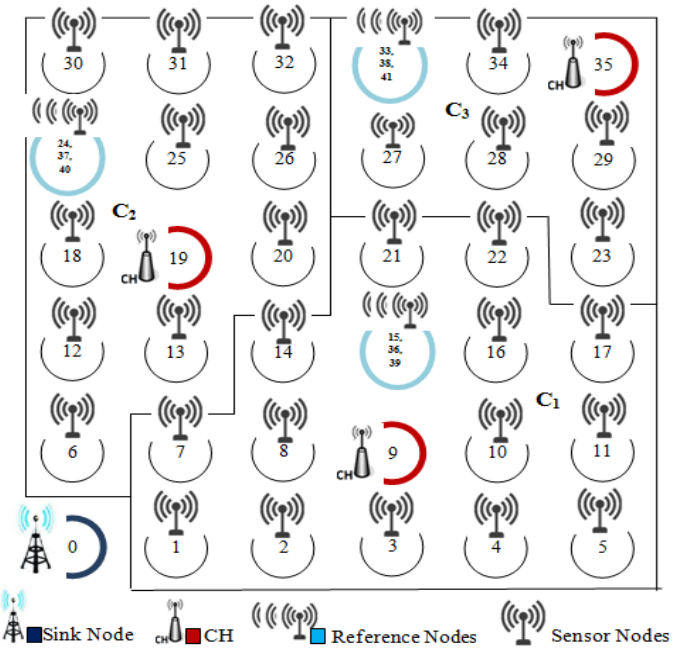


Fig. 4. Nodes Deployment in the Area of Interest

B. *Evaluation Parameters*

We are using the following parameters to evaluate the proposed algorithm detection accuracy/ False Alarm Rate, Number of messages received/ sent per node, Algorithm processing time, and finally the consumed energy.

1) *Detection Accuracy and False Alarm Rate:*

Detection accuracy and false alarm rate are used to test the performance of our proposed algorithm. The detection accuracy can be defined as the count of sensor nodes identified as faulty to the total count of faulty sensor nodes in the simulation scenario. The False Alarm Rate can be defined as the count of good sensor nodes detected as faulty to the total count of good sensor nodes in the simulation scenario. We use one Algorithm from the literature [9] as a reference to test the performance of our proposed algorithm. Different simulation scenarios with different fault probabilities are used to test, and fully investigate both algorithms. We injected the network with different fault probabilities 2.8%, 5.7 %, 8.5%, 11.4%, 14.2%, 17.1%, and 20% which corresponds to one, two, three, four, five, six, and seven faulty nodes. In the proposed algorithm the detection accuracy depends only on the accuracy of the reference sensors and we have three redundant reference sensors which increases the robustness of the system. As a result of having a good sensor reading, the CHs identify the sensors reading in their clusters correctly which gives higher detection accuracy. Unlike [9], the sensor nodes use their neighbors readings to judge the correctness of their readings. So, when having more than half of the neighbors having incorrect values, this leads to a faulty nodes can be identified as good and good sensors can be identified as faulty. This will affect the detection accuracy and false alarm rate. In simulation, when increasing the fault probability, the probability of having more faulty neighbors increases, which leads to the misjudgment as the condition of having more than half of the neighbors are good is not satisfied. In our algorithm, once you have a good reference sensor reading, the CH can easily and accurately detect the faulty sensor nodes as shown in Fig. 5.

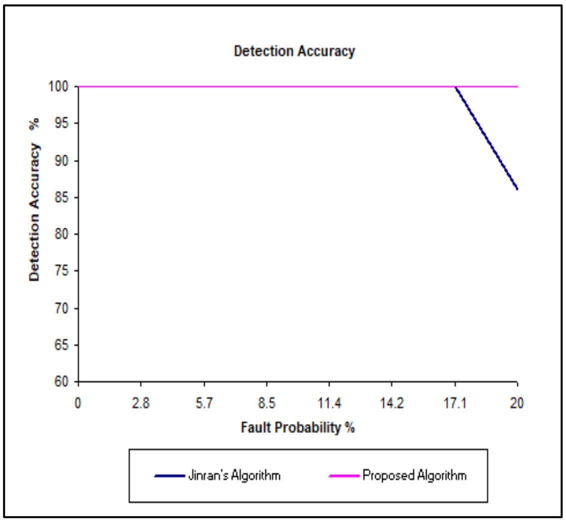


Figure 5. Detection Accuracy for Different Fault Probabilities

Also, No good sensor nodes are falsely identified to be faulty nodes as shown in Fig. 6. Even in the rare case of having the three reference sensors with mismatched readings (Need Immediate Calibration), the CH uses the highly trusted sensor (the trust index reflects the number of consecutive times of correct sensor readings) from the cluster to behave as a reference sensor and start the validation process using that sensor reading and report to the control and monitoring system that a maintenance or calibration is urgently required.

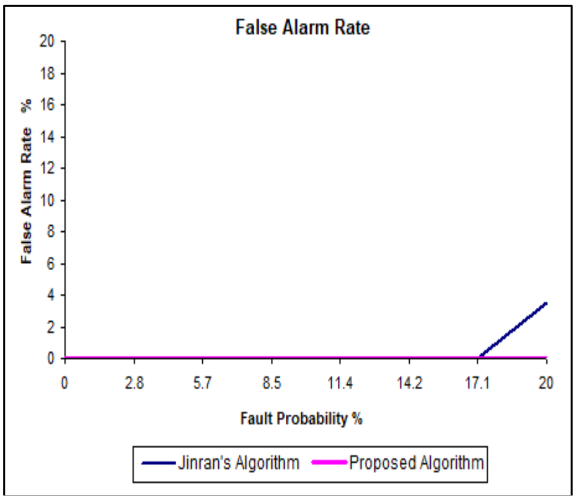


Figure 6. False Alarm Rate for Different Fault Probabilities

2) Number of Messages Received/ Sent Per Sensor Node.

In our proposed algorithm, the nodes only sense the environment, and send these readings to their cluster-heads; so, they receive zero messages from the neighboring nodes, as the readings are sent only to the CHs. In other algorithms as in [9], the nodes sense the environment, send these readings to their 1-hop neighbors (i.e. receive messages from their 1-hop neighbors), send their local status to their 1-hop neighbors, and also send their final status to their 1-hop neighbors and CHs. It is clear that, the number of received messages per node is directly proportional to the number of 1-hop neighbors. As the number of 1-hop neighbors increase, the number of received messages increase as well, as shown in Fig. 7. Regarding the number of sent messages per node and as shown in Fig. 8, in the proposed algorithm, each node sent about 32 messages to the cluster-head during the simulation time, which represents their measurements. In other algorithms as in [9], each node sent about 90 messages to either its 1-hop neighbors or cluster-head. That is because, for each sensor measurement, the senor node needs to send three massages; the measurement itself, the generated local tendency, and final node status.

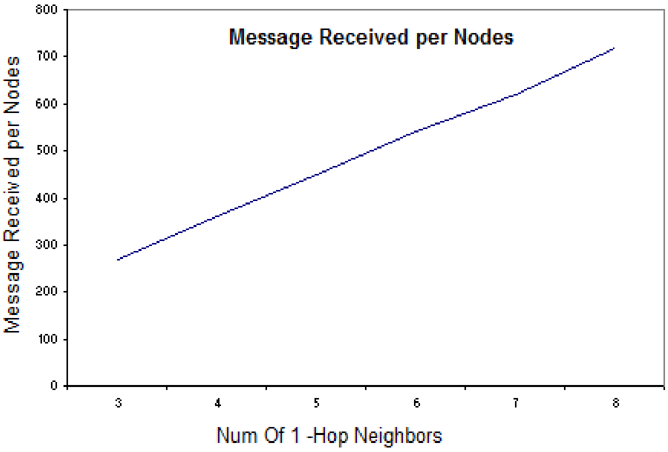


Figure 7. Messages Received Per Node in Jinran's Algorithm

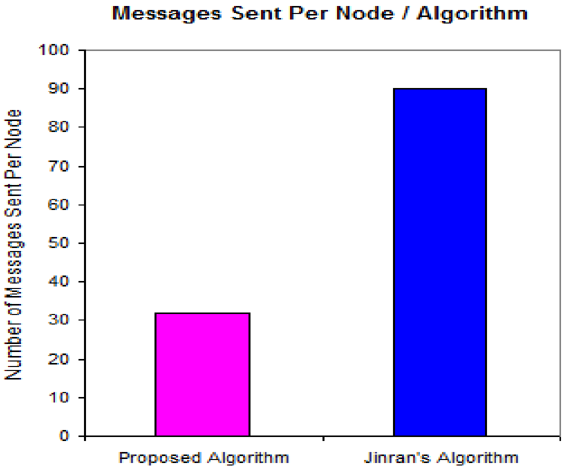


Figure 8. Number of Messages Sent Per Node / Algorithm

3) Algorithm Processing Time

As described before, the proposed algorithm is implemented at the CHs, which receive all the sensor readings in their clusters. So, it takes only one time slot to diagnose all sensor nodes. But, for other algorithms as in [9], each sensor node must receive its neighbor's readings, then generates its local tendency, then broadcasts its local tendency, then wait for the reception of the neighboring local tendencies, and finally diagnose itself either Good (GD) or Faulty (FT); or wait for the neighboring good sensors broadcasting their final status. So, as a result for the large process, it takes at least two time slots for diagnoses as shown in Table II.

TABLE II
FAULT DETECTION ALGORITHMS PROCESSING TIMES

Fault Detection Algorithm	Processing Time (Time Slots)
Proposed Algorithm	1
Jinran’s algorithm	≥ 2

4) *Consumed Energy*

The proposed algorithms showed better energy consumption than Jinran’s algorithm [9]. This is because the algorithm's functional steps. In the proposed algorithm, the sensor nodes are either in Tx, or in sleep states as a result of performing two functions only; sensing the environment, and sending the readings to the CH. But in Jinran’s algorithm, the sensor nodes are either in Tx, Rx, or in sleep states as a result of performing three functions; sensing the environment, sending the readings to the 1-hop neighbors, and receiving the readings from their 1-hop neighbors. Sending and receiving also includes the generated local tendencies, and the node final status. The cluster-heads in both algorithms almost consume the same amount of energy. That's because, they do the same functions.

V. CONCLUSIONS

In this work, we proposed a scheme to validate the sensors readings that is used in the assessment of the system states. We use the idea of Triple Modular Redundancy (TMR) in which three systems perform a specific process and by using a majority voting system to produce a trusted single output. We use three reference sensor nodes in each cluster.

Nomenclatures	
WSAN	Wireless Sensor and Actor Network
MEMS	Micro Electro Mechanical Systems
BS	Base Station
GPS	Global Positioning System
FDI	Fault Detection and Identification
CH	Cluster Head
CM	Cluster Member
TMR	Triple Modular Redundancy
GD	Good
FT	Faulty

The CH node uses the two out of three voting mechanism to select a good sensor from the three redundant reference sensors, to be used to check the correctness of other sensors in the same cluster based on the spatial correlation between sensor and reference nodes. The technique presented herein has a number of distinct advantages over other traditional sensor validation techniques: It has a lower number of messages sent / received per node than the other algorithms. It has lower power consumption than the other fault detection techniques. Its detection accuracy is very high as it detects all the faulty sensor readings under different fault probabilities. Also, no sensor nodes are falsely identified as faulty under different fault probabilities. It has a lower processing time, since others require more additional data transmissions.

REFERENCES

[1] B. Warneke, and K.S.I. Pister, "MEMS for Distributed Wireless Sensor Networks", in the Proceeding of 9th International Conference on Electronics, Circuits and Systems, Dubrovnik, Croatia, Sept. 2002, vol. 1, pp. 291-294.

[2] A.K. Paul, and T. Sato, “Localization in Wireless Sensor Networks: A Survey on Algorithms, Measurement Techniques, Applications and Challenges”, Journal of Sensors and Actuator Network, vol. 6, no. 4, pp. 1-24, 2017.

[3] IEC 62918 TR Ed.1, "Nuclear Power Plants-Instrumentation and Control Important to Safety- Technical Report on Use and Selection of Wireless Devices to be Integrated in Systems Important to Safety", 2014.

[4] G. KAKAMANSHADI, S. GUPTA, and S. SINGH, “A Survey on Fault Tolerance Techniques in Wireless Sensor Networks”, International Conference on Green Computing and Internet of Things, Delhi, India, 8-10 October 2015.

[5] V. Reppa, M. M. Polycarpou, and C. G. Panayiotou, " A Distributed Architecture for Sensor Fault Detection and Isolation using Adaptive Approximation ", in 2012 International Joint Conference on Neural Networks (IJCNN), Brisbane, QLD, Australia, 10-15 June 2012.

[6] A. Braun, R. Wichert, A. Kuijper, and D. W. Fellner, “A benchmarking model for sensors in smart environments,” in Ambient Intelligence: European Conference, Netherlands,11-13 November 2014.

[7] A. Mahapatroa, and P.M. Khilar, "Transient Fault Tolerance Wireless Sensor Networks", in the 2nd International Conference on Computer, Communication, Control and Information Technology, Hooghly, West Bengal, India. 25-26 February 2012.

[8] J. Sai, Y. Shen-fang, M. Ting-haui, and T. Chang, "Distributed Fault Detection for Wireless Sensor Based on Weighted Average", in the Proceedings of the 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing, Washington, DC, USA, , 2010, pp. 57-60.

[9] C. Jinran, K. Shubha, and S. Arun, "Distributed Fault Detection of Wireless Sensor Networks", in the Proceedings of DIWANS, New York, USA, Jan. 2006, pp. 65–72.

[10] M.C. Shin, and Y.H. Choi, "Malicious Node Detection Scheme Using Confidence-Level Evaluation in a Grid-Based Wireless Sensor Network", in wireless sensor network journal, vol. 05, pp. 52 – 60, Jan. 2013.

[11] Q. Zhang, T. Yu, and P. Ning, "A Framework for Identifying Compromised Nodes in Sensor Networks", in the Proceedings of 2nd IEEE Communication society, Baltimore, MD, August 2006, pp. 1-10 .

[12] K. P. Sharma and T. P. Sharma, “rDFD: reactive distributed fault detection in wireless sensor networks,” Wireless Networks, vol. 23, no. 4, pp. 1145–1160, May 2017.

[13] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan,”Energy-Efficient Communication Protocol for Wireless Microsensor Networks”, Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA, Jan. 2000, pp. 10 pp.

[14] A. Sharma, L. Golubchik, and R. Govindan, "On the Prevalence of Sensor Faults in Real-World Deployments", in the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, June 2007,pp. 213-222.

[15] N. Ramanathan, L. Balzano, M. Burt, D. Estrin, E. Kohler, T. Harmon, C. Harvey, 1. Jay, S. Rothenberg, and M. Srivastava, "Rapid Deployment with Confidence: Calibration and Fault Detection in Environmental Sensor Networks", Center of embedded network systems, Los Angeles, CA, 2006.

[16] G. Tolle, G. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, and W. Hong, "A Macroscope in the Redwoods", in the Proceedings of the 2nd International Conference on Embedded Networked Sensor System, New York, 2005, pp.51-63.

[17] R. E. Lyons, and W. Vanderkulk, "The Use of Triple-Modular Redundancy to Improve Computer Reliability", IBM Journal,1962.

[18] F.A. Frank, "Introduction to Radiological Physics and Radiation Dosimetry", Published online at http://onlinelibrary.wiley.com/book/10.1002/9783527617135, Dec. 2007.