HARDWARE AND SOFTWARE EXECUTION FAULT DETECTION USING ROUND TRIP DELAYS (RTD'S) AND PATHS (RTP')S IN WIRELESS SENSOR NETWORK

[1]Sandhya Rayavarapu, [2]G.Satyanarayana .

[1,2]*Sir C. R. Reddy College of Engineering, Eluru, West Godavari Dt, AP, India*

***Abstract*—**Recently large numbers of applications are developed in wireless sensor networks with more numbers of portable sensor nodes to increase the quality of service of such applications. The failure occurred in sensor nodes typically affects the quality of service of WSNs. This paper proposes a improve the Quality Of Service (QOS) of the network at low cost to gain better compassion and amount accuracy of various parameter in the field. The manual checking of failed sensor node is difficult so paper describes the Hardware and Software implementation of Sensor node failure detection using RTPs and RTDs in WSNs

Keywords*—Wireless Sensor Network; Round Trip Path; Round Trip Delay; faulty.*

I. INTRODUCTION

A Wireless Sensor Network is number of small, light weighted, low cost, low powered, multi- functioned and randomly deployed devices called sensors that monitors and detects physical as well as environmental conditions like moisture, stress, pollution, sound, temperature and daylight etc. They can operate devices like switches, actuators and motors that control these environmental conditions and hence ensure reliable and efficient means of communication through WSNs. Sensors have limitations in memory, battery life, storage, computational capabilities and bandwidth. They are usually deployed in ad-hoc manner into the environment of interest to gather the information from that area. Sensor nodes are made up of a sensor, a processor, a radio transceiver and a power supply or a battery. These nodes are placed in the field and the data sensed by the sensor nodes are collected, processed and transmitted to special node known as sink nodes where it is analyzed. As the sink nodes are located far away from the sensor nodes or the source nodes data is transmitted to sink by single path routing or though the intermediate nodes by multipath routing and further decisions are taken by the base stations. Sensor nodes communicate with intermediate nodes or with base station through radio signals, infrared and Bluetooth.

Sensor nodes are prone to failure due to following reasons: The communication device or module failure, battery failure, depletion of energy, problems related to deployment of sensor nodes in harsh and uncontrolled environment, Influence of various environmental factors due to the interaction of sensor networks and transceiver module with the environment into which they are deployed, other hardware failures , signal strength interference in the presence of obstacles, malicious attacks ,software failures , signal lost when away from the communication range of the network. A failed node produces either incorrect data that gives wrong analysis and parameter detection results or data that is deviated from the original value. So the data from them should be discarded.

Hence large numbers of portable sensor nodes are deployed in the network to increase Quality Of Service of WSNs .Use of large numbers of sensor nodes will increase the probability of sensor nodes failure [1,2]. The QOS of WSNs may be reduced due to the presence of faulty nodes in a network. Better QOS can be achieved by identifying and discarding the data from these failed nodes for analysis .So it is a need to find an accurate and a well organized method to detect failed and malfunctioned sensor nodes in the WSNs. Apart from the old methods a new method is suggested where the sensor node fault can be detected by finding out Round trip Delay of Discrete Round Trip Paths and comparing it with the threshold range value.

The algorithm, its hardware implementation and software implementation is explained in detail in the paper.

II. ALGORITHM TO DETECT AND REMOVE FAULTY NODE

There are mainly 3 steps involved in faulty node detection and they are:

A. Evaluation of Round Trip Path.: RTP is formed by minimum of 3 sensor nodes. If there are many nodes in a RTP, individual nodes will be present in many RTPs. So to detect the fault comparison of all those RTPs are required. This would be time consuming. Hence optimization of RTPs is required. Discrete selection of RTPs other than linear selection of RTPs is done where number of RTPs are reduced to two, ignoring two consecutive path after each selected linear path. This would save the analysis time. Discrete RTPs are selected by incrementing source node value by three .

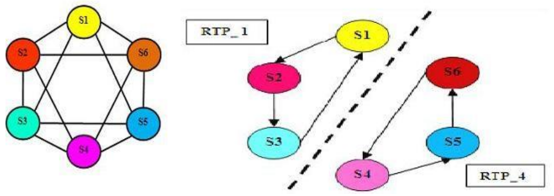


Fig.1: Circular topology and discrete RTP[1,2]

B. Estimation of RTD time Round Trip Delay Time of a specific Round Trip Path is estimated. Reduce the number of nodes that forms the RTP so RTD can be reduced and high accuracy can be obtained.

C. Detection of faulty sensor nodes: It has two phases. In the First phase highest RTD time is chosen from among the instantaneous RTD times of two discrete RTPs considering all nodes as properly working ones. This RTD time is set as threshold value. In the Second phase, instantaneous RTD time of each discrete RTP is compared with threshold value[1,2]. Discrete RTPs with RTD time greater than threshold RTD time is said to have the faulty or failed node in it and it is then analyzed.Consider a discrete path RTP (T) with RTD time greater than the threshold RTD and other RTPs have RTD time less than the threshold RTDSensor node sequence of RTP(T) is S_T-S_{T+1}-S_{T+2}.Let S_{T+1} sensor node forms next RTP. RTP (T+1) has sensor node sequence S_{T+1}-S_{T+2}-S_{T+3} and if the RTD time of RTP (T+1) is less than the threshold RTD, then the sensor node S_T is considered to be faulty node. RTD time of RTP(T) is again checked for verifying and analyzing the failed node[8]. If the RTD time of RTP(T) is infinity then S_T is considered to be failed node else it is malfunctioned node. Hence fault at S_T location is identified. If the faulty sensor node is not detected in the first stage, then RTD time of RTP (T+2) is calculated and compared with threshold RTD time conditioned the RTD time of RTP (T+1) is higher. If The RTD time of RTP (T+2) is found to be less than the threshold RTD time, then the sensor node S_{T+1} is determined as faulty node. If RTD time is found to be infinity, then S_{T+1} is failed node else it is malfunctioned node. If faulty node is not detected next the RTD time of RTP (T+2) is higher than the threshold RTD time, then sensor node S_{T+2} is determined as faulty node. The sensor node S_{T+2} will be failed if the RTD time of RTP(T+2) is infinity else it is a malfunctioned node. Hence the faults present at position S_T-S_{T+1}-S_{T+2} are monitored and located .Then S_T is compared with the N numbers of sensor nodes in Wireless Senor Network i.e. (S_N). If RTD is found to be less than threshold RTD, then next discrete RTP will be selected compared.

III. HARDWARE IMPLEMENTATION

Wireless Sensor nodes are designed using LPC11U24 micro-controller and XBEE series 2 (Digikey) wireless communication module. The software used for configuring the RTPs is XCTU and to simulate the entire network is Dock light V1.9.

A. Node Design

1) The microcontroller development board - LPC11U24:

It is a 40 pin, 54x26 mm mbed MCU. It is an ARM Cortex-M0 core processor, based on the NXP LPC11U24MCU used for prototyping low cost USB based designs, battery powered applications and 32- bit ARM Cortex-M0 based designs. It uses 5V USB, 4.5-9V supply or 2.4-3.3V battery and has simple instruction set and memory addressing. For easy programming it has a built in USB programming interface. The MCU operate at CPU frequencies of up to 50 MHz, 8KB of RAM and has a 32 KB of flash memory. Pins (p5-p30) can also be used as DigitalIn and DigitalOut interfaces. The mbed LPC11U24 microcontrollers are supported by the website mbed.org developer, with a lightweight online Compiler to provide quick access to the working environment which may be on Windows or Linux. Compiler has cookbook of published libraries, projects and code which are examples being published by the mbed community in C/C++ , SDK for high-level programming of peripherals. Board is connected using USB cable to the laptop or desktop.

2) Wireless solution

XBee Series 2, a wireless radio, named by Digi, is a Radio Frequency transceiver module that provides reliable wireless data communication over ZigBee. XBee refers to form factor and has ability to transmit and receive data at low-cost and low-power. A RF module is a small electronic circuit that can transmit and receive radio signals at different frequencies. XBee is the brand name of a family of RF modules that are produced by Digi and Xbee modules operate at 2.4GHz.Zigbee is a specification which is built on top of the IEEE 802.15.4 standard to provide communication in a wireless Personal Area Network, routing and multihop functions to packet based radio protocols. The baud rate of Xbee is set as 9600 bps. Xbee module has 20 pins and is mounted on CP2102-Serial to USB Converter which use two 10 pin receptacles each to receive XBee modules. Hardware USB Bus & Virtual Com Port drivers should be downloaded and installed for connecting CP2102 board to Laptop/Desktop via its USB port. Pin 1 of XBee is VCC it takes 2.8 to 3.4 V, pin 10 is GND ,pin 2 is the Transmit pin and pin 3 is the Receive pin .The operating frequency band of ZigBee is 2.4GHZ, 915MHz and 868MHz.Its battery is not rechargeable. The network join time is 30 milliseconds.

3) XCTU configuration:

XBee modules are configured using XCTU software provided by DigiKey. Once the Xbee modules are detected select the required Communication Ports and configure the XBee modules to Zigbee devices and assign it the Source and Destination address so the data reaches the destination node. XBees modules are mounted on CP2102 USB driver. There are mainly two modes in XCTU AT and API modes.In AT mode nodes are identified by the Destination address in memory and there is no need to change destination addresses very often. It is used for communication in simple network. API mode will switch the module from AT mode to a framed version of data where the data is framed with other information such as address and Checksum. It is used for larger networks having multiple targets. It allows the destination address to be changed fast.

4) Interfacing LPC11U24 AND XBEEE SERIES 2 module

Connect ARM mbed board with PC via USB cable. Using XCTU -configure the XBee modules as Zigbee Devices. Write required source code for mbed board through mbed.org developer web site. Compile it to load it to the controller. Now connect pin1 of XBee to pin 40 of mbed board and pin 10 of Xbee to pin 1 of mbed board to provide VCC and Ground. Connect TX(p2) and RX(p3) of Xbee to RX(p10) and TX (p9)of controller. On Controller reset the required operation will be performed and we can view the transmitted data on XCTU terminal.

B. Programming Details

1) Round Trip Path creation

The first step is creation of two discrete Round trip paths. Configure the nodes using XCTU. Assign the source and destination address for each node. The MAC address of Node 2 is given as the destination address for Node 1 ,the MAC address of Node 3 is given as the destination address for Node 2 and the
MAC address of Node 1 is given as the destination address for Node 3. Hence forming the RTP 1.Similarly configure for RTP 2 containing Node4, Node5, and Node 6.

2) Estimation of Round Trip time

Consider RTP1, start a timer and print a message to XBee 1. As the MAC address of one node is set as the destination address for the other the message will be transferred from Node 1-Node 2-Node 3-Node 1. As it reaches back to Node 1, the node checks whether the transmitted message is same as the received message. If so the timer is stopped. The difference between the start and stop time of the timer is calculated which gives the Round Trip Delay time of RTP1. Follow the same procedure for finding the RTD time of RTP2. Instantaneous value will be noted the highest of RTD time is set as threshold. After setting threshold if instantaneous RTD value of any of the two RTPs cross this threshold value then we can say nodes forming that RTP is faulty or failed c)Simulation of network

The entire network is simulated in real time using Docklight V1.9 software. It is a testing, analysis and simulation tool which uses serial communication protocols. Install Docklight V1.9 and the driver software, set baud rate as 9600 bps and choose the communication port of Node 1 to allow it to send and receive the data. Name the sequence and on clicking the send button we can see the data send and received by node 1 which is shown in the Figure 2.. d)Threshold RTD time detection

There is inherent time delays present in the hardware modules due to the delays imposed by two hardware devices that form the nodes. Delay due to microcontroller is 0.05239s and delay due to ZigBee device is 0.720s. Hence inherent delay is 0.6676s. RTP is formed by the combination of three nodes therefore total delay due to hardware will be three times of sensor node delay i.e.,2.002825s. So the threshold RTD time is 2.003s. On minimizing the delay associated with hardware devices ,inherent delay can be reduced.

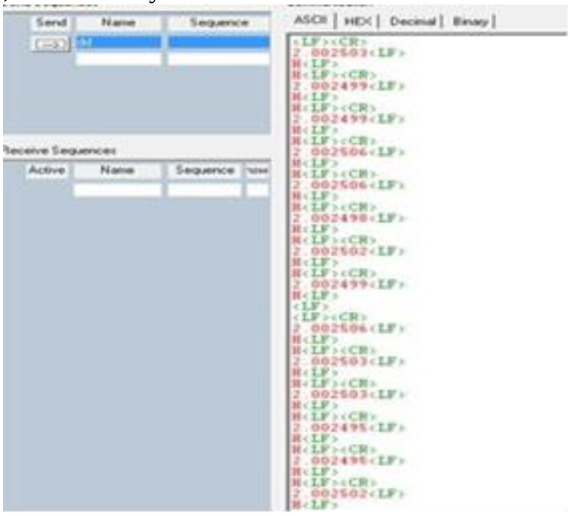


Fig.2: RTD values obtained on Hardware implementation

Fig.3: Node 1 made faulty

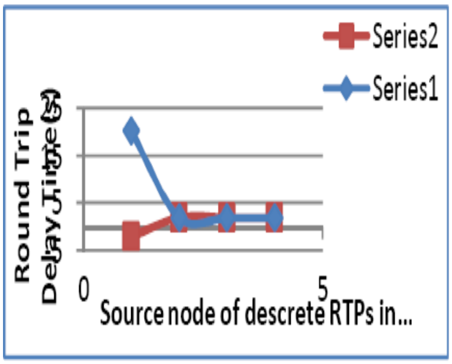
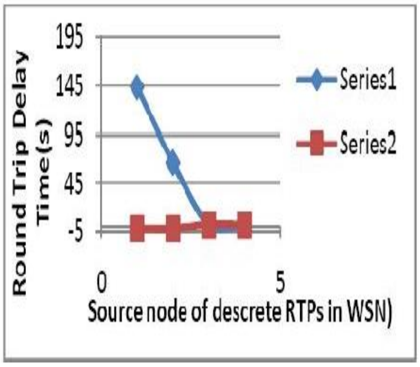


Fig.4: Node 2 made faulty



TEST CASES, RESULTS AND FAULT DETECTION

To test the method one of the sensor node in the Wireless sensor network is set faulty by switching off the power supply. The figures below shows the RTD time result of malfunctioned (Series 1) and faulty node(Series 2) which is indicated in red and blue color. Faulty nodes are located based on the algorithm explained above for fault detection. The infinity Value of RTD time indicates the node is faulty and is represented by -2.

1) CASE 1:Sensor node S1 is made faulty.S1 is the first node of RTP1. From the Figure 3 it is evident that RTP4 is working correctly as the RTD time of this RTP2 is less than the threshold value. Therefore S_4,S_5,S_6 are not faulty. As the RTD time of RTP1 is greater than threshold RTD time S_1 is said to be malfunctioned and if it infinity threshold value as indicated in series 2 shows S_1 is failed.

2) CASE2: Sensor node S_2 is made faulty.S_2 is the second node of RTP1 equivalent to S_{T+1} in the algorithm. Number of RTPs require other than source node is 4.Higher RTD time of RTP 1,2 and lesser RTD time of RTP 3 shows S_2 is malfunctioned and infinity value of RTP 1,2 as indicated in series 2 shows S_2 is failed. From Figure 4 it is evident that RTP4 is working correctly as its RTD time is less than the threshold value. Therefore S_4,S_5,S_6 are not faulty.

3) CAS33: Sensor node S_3 is made faulty.S_3 is third node of RTP1 equivalent to S_{T+2} in the algorithm. Number of RTPs require other than source node is 4.Higher RTD time of RTP 1,2,3 shows S_3 is malfunctioned and infinity value of RTP1,2,3 as indicated in series 2 shows S_3 is failed. From Figure 5 it is evident that RTP4 is working correctly as its RTD is less than the threshold value. Therefore S_4,S_5,S_6 are not faulty.

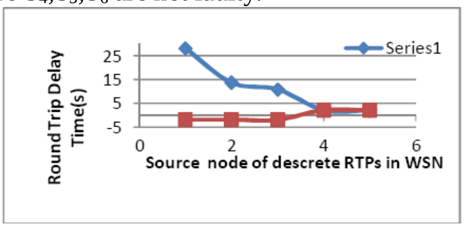


Fig.5: Node 3 made fault

IV. SOFTWARE IMPLEMENTATION HARDWARE & SOFTWARE COMPARISION

To prove the efficiency of the method software analysis Evaluation of net RTD in software and hardware The exact

RTD estimation in hardware implementation is shown in Figure 9. It ranges from 1ms to 36 ms in software and from

is done using the open source software NS2.Round Trip Path is formed by the combination of three nodes and RTD is estimate. Fault in node is detected using the algorithm explained above.

Node-Implementation Procedure First Create a simulator object and open the NAM trace file. Define a finish procedure, create nodes and name it. Set up UDP connection and CBR over UDP. Call the finish procedure and run the simulation. Number of sensor node may be 6, 10,20,30,50 or 100.Simulation area is 50X50 meters. Routing protocol used is RTD. Transmission range is 1 m. Simulation time is 5 s.

Algorithm Implementation a)RTP creation

Set up UDP connection between the Nodes1 -Node 2, Node 2 - Node 3 and Node 3- Node1, to form one RTP and between Node4 - Node5, Node 5 -Node 6 and Node 6 -Node 4 to form other RTP. Set up CBR over the UDP connection. Calculate throughput of the network .

RTD Estimation

Calculate the end to end delay between sources and sink nodes in RTP and estimate the RTD time. Delay assigned to link is 4ms. As there are 3 links, total delay of 12ms is applied to each link. CBR from one node to the other is delayed to make the node faulty or malfunctioned and it is cut off to make the node failed. This will produce delay in RTD time of the packet transfer. Hence the delayed RTD time is estimated.

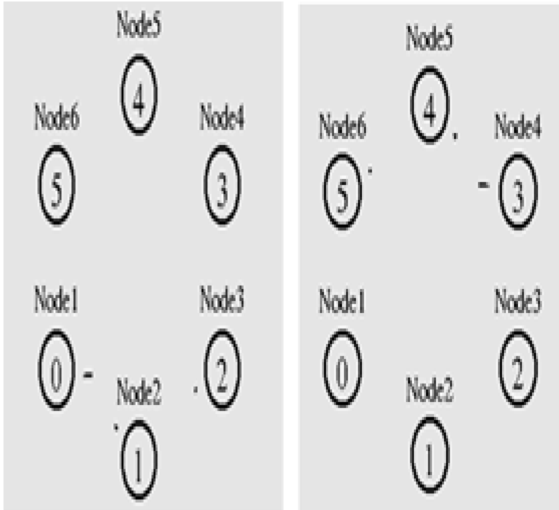


Fig.6: Two discrete RTPs created

4 s to 144 s in hardware. The hardware and software RTD time results are almost same evident from Figure 10 and Figure 11 below. This shows the efficiency of the method.

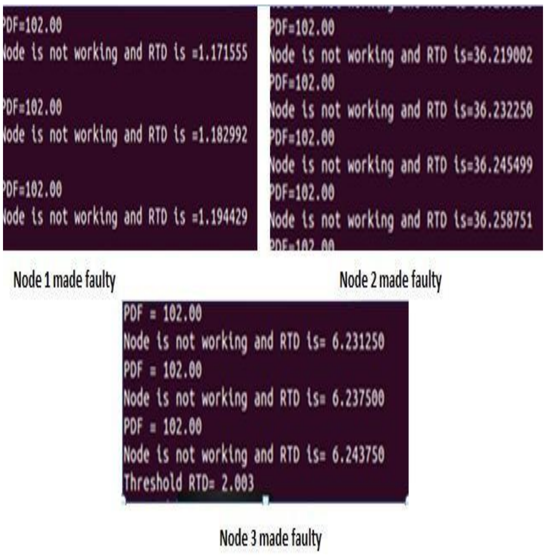


Fig.7: Estimated RTD

Fig.8: RTD of faulty nodes evaluated by software

RTP	*Sensor node sequence in RTP*	*Delay due to MCU and XBee (s)*	*Exact RTD time (s)*
RTP1	$S_1S_2S_3S_1$	*2.003*	*20.25*
RTP2	$S_2S_3S_4S_2$	*2.003*	*143.9*
RTP3	$S_3S_4S_5S_3$	*2.003*	*28.12*
RTP4	$S_4S_5S_6S_4$	*2.003*	*52.49*
RTP5	$S_5S_6S_1S_5$	*2.003*	*19.23*
RTP6	$S_6S_1S_2S_6$	*2.003*	*3.84*

Fig.9: Evaluation result of net RTD in hardware

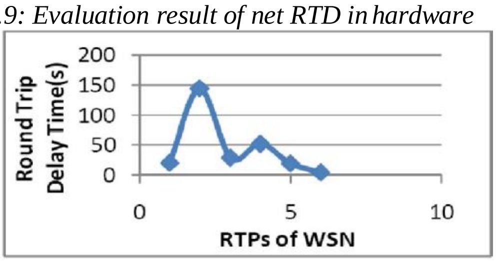


Fig.10: RTD time results for 6 RTPs in hardware

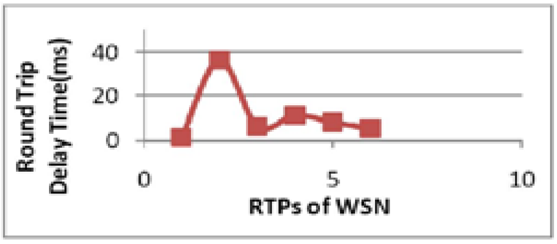


Fig.11 : RTD time results for 6 RTPs in Software

V. CONCLUSIONS

The algorithm is successfully implemented and tested on hardware and software methods hence the scalability and effectiveness of the methods is verified. This method provides an outstanding solution for detecting faults in a Wireless Sensor Network using the concepts of Round Trip Path and Round Trip Time.

REFERENCES

[1] Ravindra Navanath Duche and Nisha P. Sarwade, ”Sensor Node Failure Detection Based on Round Trip Delay and Paths in WSNs” IEEE sensors journal, vol. 14, no. 2,february 2014

[2] Swathi Ravindran, Bindu Sebastian, Jerrin Yomas, “An Improved Method for Sensor Node Failure Detection in Wireless Sensor Networks” IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 10, Issue 5, Ver. II (Sep - Oct .2015), PP 65-70

[3] Abolfazl Akbari, Arash Dana , Ahmad Khademzadeh and Neda Beikmahdavi, “Fault Detection and Recovery in Wireless Sensor Network Using Clustering”, International Journal of Wireless & Mobile Networks (IJWMN)Vol

[4] Krishna M., Chaitanya D. K., Soni L., Bandlamudi S.B.P.R., Karri., R.R.: (2019), "Independent and Distributed Access to Encrypted Cloud Databases". In: Omar S., Haji Suhaili W., Phon-Amnuaisuk S. (eds) Computational Intelligence in Information Systems. CIIS 2018. Advances in Intelligent Systems and Computing, vol 888. pp 107-116, Springer Nature. DOI: 10.1007/978-3-030-03302-6_10

[5] Dr. Marlapalli Krishna, V Devi Satya Sri, Bandlamudi S B P Rani and G. Satyanarayana. “Edge Based Reliable Digital Watermarking Scheme for Authorized Ownership” International Journal of Pure and Applied Mathematics pp: 1291-1299, Vol-119, Issue-7, 2018.

[6] Dr. Marlapalli Krishna, Bandlamudi S B P Rani, V Devi Satya Sri and Dr. Rama Rao Karri. “Filter Based Jpeg Compression for Noisy Images” Journal of Advanced Research in Dynamical and Control Systems, pp: 1233-1248, Vol-9, Issue-18, 2017.

[7] Dr. M. Krishna. “The VLIW Architecture for Real-Time Depth Detection in Image Processing”, International Journal of Computer Science & Mechatronics, pp: 1-9, Vol.2.Issue.VI, Dec-2016.

[8] Dr. M. Krishna. “An Efficient Multi Dimensional view for vehicles by Patch memory management in image processing”, International Journal of Computer Science & Mechatronics, PP:1-10, Vol.1.Issue.V, Dec-2016.

Gunupusala . Satyanarayana has completed his M.Tech in Computer Science and Engineering from JNTU Kakinada. His

research in wireless sensor networks . He is currently associated with Sir C R Reddy College of Engineering, ELURU, West Godavari District A.P. India. Affiliated to Andhra University.