International Journal of Engineering Science Invention (IJESI)
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org ||Volume 7 Issue 12 Ver. III || Dec 2018 || PP 65-71

Techniques for Fault Detection in Wireless Sensor Networks

Mohammed Bakhtawar Ahmed
Assistant Professor, Amity University, Chhattisgarh

Abstract: *At the moment, wireless sensor networks (WSN) emerge as a revolution in all aspects of our lives. WSNs have unique specifications of themselves that describe them differently than other networks. Fault tolerance is one of the most important challenges of these networks. During the development of WSN solutions it is necessary to take into account five key characteristics: scalability, safety, reliability, self-repair and robustness. In this paper, the main objective is to provide a comparative study of Fault tolerance techniques using different approaches. The sensor nodes have different power and computational constraints. To provide a quality service through coverage protocols, protocols must be developed to provide fault tolerance, event reporting and energy efficiency retention.*

Keywords: *wireless sensor network (WSN); fault tolerance; cluster head; fault tolerant systems; fault diagnosis;*

Date of Submission: 13-12-2018 Date of acceptance: 28-12-2018

I. Introduction

A wireless sensor network is a collection of nodes organized into a cooperative network **[1, 2].** A wireless sensor network (WSN) consists of tiny, low-powered sensors communicating with each other ossibly through multihop wireless links and collaborating to accomplish a common task. A wireless sensor network is a system of small, wirelessly communicating nodes where each node is equipped with multiple components **[5].** The nodes communicate wirelessly and often self-organize after being deployed in an ad hoc fashion. Such a network is envisioned to integrate the physical world with the Internet and computations. The power supply on each node is relatively limited, and replacement of the batteries is frequently often not practical due to the large number of the nodes in the network. Each node consists of may contain multiple types of memory (program, data and flash memories), processing capability (one or more microcontrollers, CPUs or DSP chips), have a RF transceiver (usually with a single omnidirectional antenna), have a power source (e.g., batteries and solar cells), and accommodate various sensors and actuators. Sensor nodes collaborate with each other to perform tasks of data sensing, data communication, and data processing **[2].** Systems of 1000s or even 10,000 nodes are anticipated. Such systems can revolutionize the way we live and work.

Advances in sensor technology and wireless communications have allowed design and development of large-scale and cost-effective sensor networks that are suitable for various applications, such as health monitoring, environmental monitoring and battlefield surveillance. A key aspect in the design of WSN is to keep them functional for as long as possible. Because of the low power (or energy) of the battery, the sensors can completely deplete the energy or have residual energy below the threshold required for the sensors to work properly. These sensors are called faulty because they can not perform any monitoring tasks properly. It is said that a WSN is functional if at any time there is at least one communication path between each pair of non-faulty sensors in the network.

However, the existence of communication paths between sensor pairs is related to another fundamental property of the WSN, called vertex connectivity (or simply connectivity). In general, detection applications must be fault tolerant, in which any pair of sensors is usually connected by multiple communication channels. Therefore, the network functionality and, therefore, the fault tolerance of the network depend to a large extent on connectivity. Figure 1 below represents the common architecture of wireless sensor networks and their nodes.

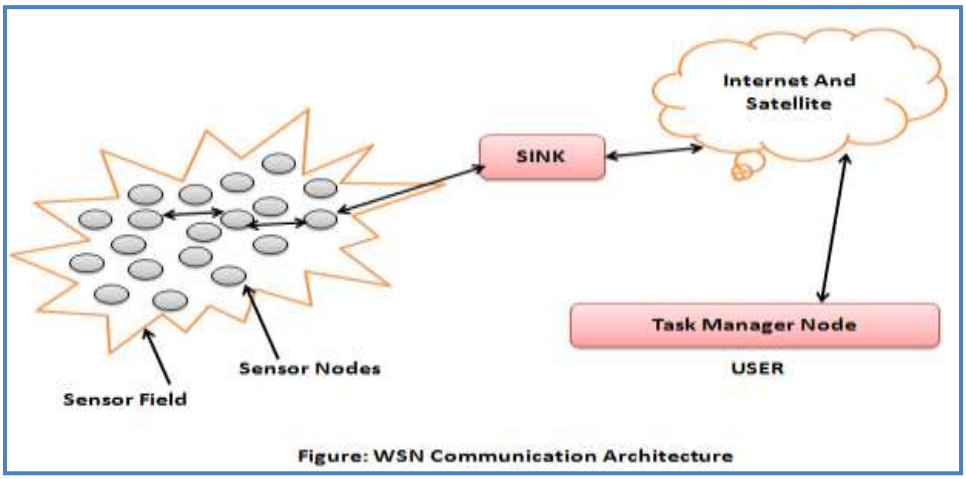

Figure 1: WSN Communication Architecture

Wireless sensor networks can detect and forward detected data and perform responses based on the received data in an appropriate manner. The WSN consists of sensor nodes and sink nodes. Sensor nodes usually have low costs, limited power and limited transmission range; They are responsible for detecting events or detecting environmental data. Sink nodes are more resource-rich nodes with abundant sources of energy, greater communication and computational capacity, and the ability to perform powerful reactions. When the receiving node performs an action, these nodes are called actor nodes. When a sensor node detects some data that will be delivered to its monitoring area, it will transmit the event to neighboring nodes, which in turn will send the event back to another jump. The hardware components of a sensor node are shown in Figure 2. In this way, the event reaches the sink. Once the receiving node receives the data, it will perform the corresponding reactions appropriately. WSNs allow some realistic applications, such as military control, phenomena control and attack detection **[1].**

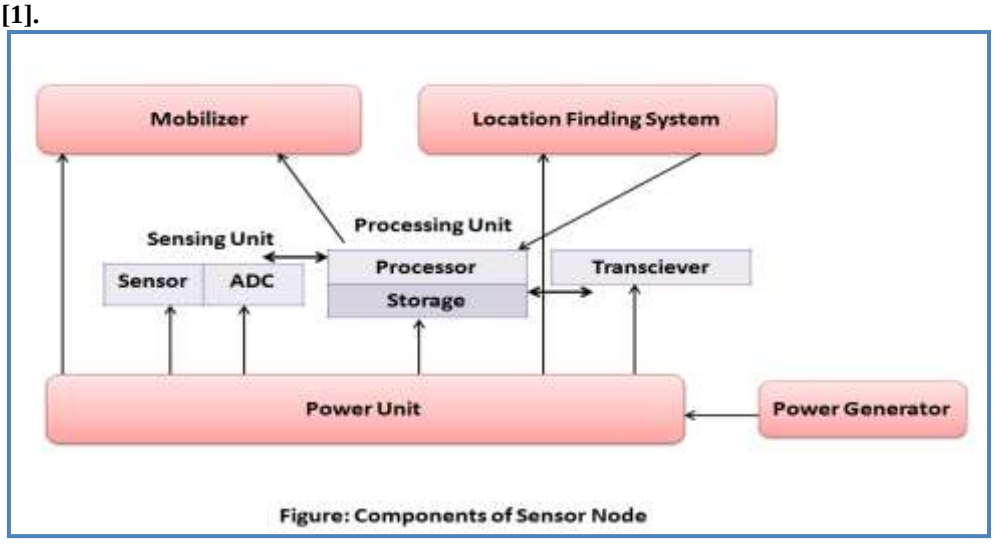

Figure 2: Components of Sensor Node

Currently, wireless sensor networks are starting to develop at an accelerated rate. It is not unjust to expect that in the next 10-15 years the world will be covered by wireless sensor networks that can be accessed through the Internet. This could be equivalent to the Internet becoming a physical network. This new technology is exciting with unlimited potential for many areas of application, including environmental, medical, military, transportation, entertainment, crisis management, national defense and smart spaces. Since a wireless sensor network is a distributed system in real time, a natural question is how many system solutions distributed in real time can be used in these new systems?

Unfortunately, very few previous jobs can be applied and new solutions are needed in all areas of the system. The main reason is that the set of assumptions underlying the previous work has changed drastically. Most of the investigations of distributed systems in the past have assumed that the systems are wired, have unlimited power, are not real-time, have user interfaces such as screens and mice, have a fixed set of resources, treat every node in the system as very important and are independent of the position. In contrast, for wireless

sensor networks, systems are wireless, have low power, are real-time, use sensors and actuators as interfaces, have dynamically changing resource sets, aggregate behavior is important and position is critical. Many wireless sensor networks also use devices with minimal capacity, which creates further pressure on the possibility of using previous solutions. Although sensor networks are a special type of ad hoc networks, protocols designed for ad hoc networks can not be used as they are for sensor networks due to the following reasons:

a) The number of nodes in the sensor networks is very large and must scale more orders of magnitude than ad hoc networks and therefore requires different and more scalable solutions.

b) The data transmission rate is expected to be very low in WSN and is statistical in nature. But the ad hoc mobile network (MANET) is designed to carry rich multimedia data and is mainly implemented for distributed computing.

c) A single sensor network is usually implemented by a single owner, but MANET is usually performed by multiple independent entities. **[4]**

d) The sensor networks are data-centric, ie the queries in the sensor network are directed to nodes that have data that meet certain conditions and a single addressing is not possible, as they do not have global identifiers. But MANET is centered on nodes, with queries addressed to particular nodes specified by their unique addresses.

e) The sensor nodes are normally implemented once in their useful life and those nodes are usually stationary, with the exception of some mobile nodes, while the nodes in MANET move in an ad hoc way.

f) Like the nodes of the MANET sensors, they are also designed for autoconfiguration, but the difference in traffic and energy consumption requires separate solutions. With respect to ad hoc networks, sensor nodes have limited power supply and energy recharging is not practical, taking into account the large number of nodes and the environment in which they are implemented. Therefore, energy consumption in WSN is an important metric to consider.

g) Sensor networks are application specific. You cannot have a solution that is suitable for all problems.

II. Fault Tolerance

Fault tolerance or graceful degradation is the property that allows a system (often based on a computer) to continue functioning correctly in the event of a failure (or one or more errors within) of some of its components. Fault tolerance is the ability of a system to offer a desired level of functionality in the presence of failures **[8].** Nodes in WSN are prone to errors due to power failure, hardware failure, communication link failures, malicious attacks, etc. If its operational quality decreases completely, the decrease is proportional to the severity of the error, compared to a system with a naive design in which even a small error can cause a total interruption. Fault tolerance is particularly required in high availability or critical life systems. It is said that a WSN is fault tolerant if it remains functional despite the failures of the k-1 sensor, where k is network connectivity.

Another important aspect in the design of WSN is what is called detection coverage, a good indicator of the quality of surveillance in a field of interest **[6].** Some detection applications require complete coverage, as all field positions are covered by at least one sensor. Furthermore, in order to address the problem of faulty sensors, double coverage of the same region is desirable. Sensor redundancy is closely related to the degree of detection coverage required by sensing applications, ie the maximum number of sensors that simultaneously cover any position in the field. Keep in mind, however, that detection coverage and network connectivity are not entirely orthogonal concepts. While the detection coverage depends on the detection range, connectivity is related to the communication range of the sensors. The detection coverage loses significance if the sensors fail to exchange the detected data, then arrive at a central meeting point, called sink, for further analysis. Therefore, for a network to function properly, both detection coverage and network connectivity must be maintained.

Fault tolerance is not just a property of individual machines; It can also characterize the rules with which they interact. For example, the Transmission Control Protocol (TCP) is designed to enable reliable two-way communication in a packet-switched network, even in the presence of imperfect or overloaded communication links. To do so, communication endpoints must provide for loss, duplication, reordering, and corruption of packets, so that these conditions do not damage data integrity and only reduce the performance of a proportional amount.

2.1 The Need for Fault Tolerant Protocols and Design Issues

Sensor networks share common error problems (such as connection errors and congestion) with traditional wired and wireless distributed networks, as well as introducing new sources of faults (such as node failures). Fault tolerance techniques for distributed systems include tools that have become industry standards such as SNMP and TCP / IP, as well as more specialized and / or more efficient methods that have been extensively studied **[14].** Faults in sensor networks can not be resolved in the same way as traditional wired or wireless networks due to the following reasons:

a) traditional network protocols generally do not care about energy consumption, as wired networks have constant and ad hoc power, wireless devices can be recharged periodically;
b) traditional network protocols aim to achieve a reliable point to point, while wireless sensor networks agree with reliable event detection;
c) in sensor networks, error nodes occur more often than in wired networks, where it is assumed that servers, routers and client machines typically operate most of the time; This implies that a closer node health monitoring system is needed without incurring significant overheads;
d) traditional wireless network protocols based on functional-level protocols to avoid package collisions, the hidden terminal problem and channel errors using the operator's physical sense (RTS / CTS) and sense the virtual operator (channel monitoring).
Many detection algorithms recent failures are loosely defined defect patterns or failures too general definition. **[6],** lists briefly selected faults and develops a method for detecting transversal in line defects based on very broad definitions of validation errors. Looking beyond the techniques of detection and correction of errors, there has been significant work that frames our willingness to provide taxonomy fault.

2.2 Taxonomy of Fault Tolerant Techniques

Recent research has developed several techniques that deal with different types of errors in different layers of the network stack. To help understand the hypotheses, the approach and the insights behind the design and development of these techniques, the taxonomy of the different fault tolerance techniques used in traditional distributed systems **[15]** was given as
a) Failure prevention: to prevent or prevent failures;
b) Detection of faults: it is a matter of using different metrics to collect the symptoms of possible errors;
c) Fault isolation: this is to correlate different types of fault indications (alarms) received from the network and to propose different hypotheses of failure;
d) Fault identification: it is a matter of testing each of the hypotheses proposed in order to locate and accurately identify faults;
e) Fault recovery: this is to deal with failures, ie to reverse their negative effects.

The identification and isolation of faults are sometimes collectively defined as fault diagnostics. Keep in mind that there are some techniques that deal with a combination of all these aspects. In reality, these techniques operate at different levels of the network protocol stack. Most failure prevention techniques work at the network level, adding redundancy in routing routes; most fault detection and recovery techniques work in the transport layer; and some error recovery techniques are performed in the application layer, hiding failures during online data processing.

III. Literature Review

Fault detection is the first step in error handling, where the network system must correctly identify an unexpected error. The failure detection approaches in the WSN can be classified into two types: centralized and distributed approach.

3.1 Centralized Approach

The centralized approach is a common solution for identifying and identifying the cause of suspected errors or nodes in WSN. generally; a geographically or logically centralized sensor node (in terms of base station **[5, 17 and 18],** central controller or administrator **[4],** sink) assumes responsibility for monitoring and locating defective or non-compliant nodes in the network. Most of these approaches consider that the central node has unlimited resources (for example, energy) and is able to perform a wide range of error handling maintenance. They also believe that the useful life of the network can be extended if the complex administration works and the transmission of messages can be changed to the central node. The central node usually adopts an active detection model to restore network performance states and individual sensor nodes by periodically injecting requests (or queries) into the network. Analyze this information to identify and locate faulty or suspected nodes. In **[17],** the base station uses marked packages (containing geographical information of origin and destination locations, etc.) to detect sensors. It is based on node response to identify and isolate suspect nodes in routing paths when excessive packet fall is detected or compromised data is detected. In addition, the central administrator provides a centralized approach to prevent potential failures by comparing the current or historical states of the sensor nodes with the general information models of the network (ie the topology map and the energy map). In summary, the centralized approach is efficient and accurate to identify network failures in certain ways.

3.2 Distributed Approach

The distributed approach favors the concept of local decision-making, which evenly distributes error handling in the network. The goal is to allow a node to take certain levels of decision before communicating with the central node. He believes that the more a decision can be made by a sensor, the less information needs to be delivered to the central node. In other words, the control center should not be informed unless there has actually been a failure in the network. Others face the use of the decision-making merger center (ie, several merger nodes across the network) to make definitive decisions on suspicious nodes in the network **[11, 12, 14, 16].**

(i) Node Self-Detection

Numerous researchers have proposed a self-determination model to control the malfunction of the physical components of a sensor node through the hardware and software interface. The self-determination of the node failure is somewhat simple because the node simply observes the binary outputs of its sensors when it compares with the default fault patterns. In data dissemination protocols that provide large segments of data to the entire network (or part of the network), destination nodes are responsible for detecting missing packets or missing packets and for communicating source feedback through messages NACK.

(ii) Neighbor Coordination

Fault detection through neighbors coordination is another example of the error handling distribution. The nodes are coordinated with their neighbors to detect and identify network faults (ie a suspicious node or anomalous sensor readings) before consulting the central node. For example, in a decentralized fault diagnostics system **[12],** a sensor node can execute a phased diagnostic algorithm to identify the causes of a fault. Furthermore, a node can also consult the diagnostic information of its neighbors (in the communication interval of a jump). This allows the decentralized diagnostic framework to be easily resized on much larger and denser sensor networks, if needed. Alternatively, suspicious (or failed) nodes can be identified by comparing the readings of their sensors with the average readings of neighbors. With this motivation **[9],** he developed a localized algorithm to identify a suspicious node whose sensor readings have a big difference compared to neighbors. Although this algorithm works for large sensor networks, the probability of sensor failures must be reduced. If half of the sensor neighbors are faulty and the number of neighbors is even, the algorithm can not detect failures as efficiently as expected. Furthermore, this approach also requires each sensor node to be aware of its physical location with expensive GPS or other technology without GPS. **[7, 8]** solves the accuracy of fault detection through a two-phase coordination scheme. Similar approach in **[6],** where a node can listen to its neighbor using WATCHDOG. If the data packets have not been successfully transmitted by neighbors of a node that is currently being routed, faults or neighbors errors can easily be detected.

(iii) Clustering Approach

Clustering **[14]** has become an emerging technology to create scalable and energy-balanced applications for WSN. **[18],** derived an efficient fault detection solution using a cluster-based communication hierarchy to simultaneously achieve scalability, integrity and accuracy. They divide the entire network into different groups and then distribute the error handling in each individual region. The intracluster heartbeat is used to identify failed nodes in each group. Meanwhile, **[13]** adopts event-based sensing through a management agent model supported by the MANNA management architecture **[3].** In this approach, agents are run on cluster heads with more resources than common nodes. An administrator is outside the WSN, where he has a global view of the network and can perform complex administrative and analytical tasks that would not be possible within the network. Each node controls its energy level and sends a message to the administrator or agent whenever a change of status occurs. The administrator then uses this information to create a topological map and a network power model to monitor and detect possible network failures in the future. Furthermore, random distribution and the limited transmission capacity of common nodes and group headers do not guarantee that each common node can be connected to a group header. Furthermore, transmission costs for polling network status were not considered in this approach.

(iv) Distributed Detection

The basic idea of distributed detection is to ensure that each node decides on failures (usually, binary data from the anomalous reading of the sensor). This approach is particularly energy efficient and ideal for data-driven sensor applications. However, many research challenges remain to achieve a better balance between the accuracy of fault detection and the use of network power. In general, the efficiency of such fault detection schemes is counted in terms of node communication costs, accuracy, detection accuracy, and the number of faulty sensor nodes that can be tolerated in the network. In Clouqueurs' work **[15],** fusion sensors (in terms of

node managers) coordinate with each other to ensure that they get the same global information on the network before making a decision, as faulty nodes can send them inconsistent information.

IV. Conclusions

Mobile computing is an emerging trend in distributed computing for different applications. Mobile host mobility (MH), limited battery power in the HD, limited wireless bandwidth, noisy wireless environment, limited transfer and storage (or lack of stable storage space in the HD) present problems difficult to provide fault tolerance to such mobile processing systems.

Due to the possible implementation in uncontrolled and difficult environments and due to the complex arc, wireless sensor networks are and will be subject to numerous malfunctions. The objective of this document is to identify the most important types of faults, the techniques for their detection and diagnosis and to summarize the first techniques to guarantee the efficiency of failure resilience mechanisms. In addition to an overview of fault tolerance techniques in general, and in particular in sensor networks, techniques to ensure fault resilience during sensor fusion and the heterogeneous fault tolerance approach were also analyzed. Integrated automatic repair.

Table 1: Existing Chart for Fault Tolerance Techniques in WSN

Name of Technique	Working Principle	Advantages	Dis-advantages
Online Fault Detection	Approach applied on arbitrary type of fault model, with probability based identification of faulty nodes	Accuracy in presence of Gaussian noise even for relatively sparse networks.	Effort restricted only to faults in sensors rather than taking other communication and computation units of a node into consideration.
Centralized Fault Detection	Centralized sensor node takes responsibility of identifying and locating the failed or misbehaved node.	Accurate and Fast for identifying faulty node	Central node becomes single point of data traffic concentration and also causes high volume of message and quick energy depletion
Sympathy	Message flooding approach to pool event data and current states from sensor nodes to a Sympathy node which further transmits to sink node	Fetches data to a sympathy node rather than each node sending directly to sink node.	Message broadcasting creates redundancy of data at sympathy node.
WATCHDOG	A node can listen on its neighbor if data packets have not been transmitted properly by its neighbors it is currently routing to.	Encourages concept of local decision making. More decision a node makes the less will be required to deliver to sink node	Slow and error prone as it is always difficult to keep an eye on all its neighbors.
FT-DSC Protocol	Clustered based approach in which CH receives info from members only when event of interest occurs	Energy saving by not delivering messages to CHs in every time slot of a frame	Selection of cluster head is always done on basis of level of energy remaining.
FREM	Only requires the touch set on the destination node for quick restart, the remainder of image is transferred after process is restarted on estination.	Allows fast restart of a failed process without requiring the availability of entire checkpoint image.	Issues with this are how to accurately identify the touch set, how to set the tracking window, how to load partial image on destination node.

References

[1]. Ian F. Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci, "A Survey on Sensor Networks", proceedings of IEEE Communications Magazine, August 2002.

[2]. Ian F. Akyildiz, Ismail H. Kasimoglu, "Wireless sensor and actor networks research challenges", Elsevier Ad Hoc Networks2, pp. 351–367, 2004.

[3]. Jennifer Yick, Biswanath Mukherjee, Dipak Ghosal, "Wireless sensor network survey", Elsevier Computer Networks 52, pp. 2292–2330, 2008.

[4]. L. Paradis and Q. Han, "A survey of fault management in wireless sensor networks", Journal of Network System Management, pp. 171-190, 2007.

[5]. N. Ramanathan, K. Chang, R. Kapur, L. Girod, E. Kohler, and D. Estrin, "Sympathy for the sensor network debugger", in SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems, pp. 255-267, 2005.

[6]. A. Mahmood, E. J. McCLUSKEY, "Concurrent Error Detection Using Watchdog Processors", IEEE TRANSACTIONS ON COMPUTERS, pp. 160-174, 1988.
[7]. F. Koushanfar, M. Potkonjak, and A. Angiovanni-Vincentell, "Fault tolerance techniques for wireless ad hoc sensor networks", Sensors 2002, Proceedings of IEEE, pp. 1491-1496, 2002.
[8]. S. Harte, A. Rahman, and K. Razeeb, "Fault tolerance in sensor networks using self- diagnosing sensor nodes", Intelligent Environments, 2005, The IEEE International Workshop, pp. 7-12, June 2005.
[9]. W. L. Lee, A. Datta, and R. Cardelloliver, "Winms: Wireless sensor network-management system, an adaptive policy-based management for wireless sensor networks", School of Computer Science and Software engineering, University of Western Australia, 2006.
[10]. A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "Spins: security protocols for sensor networks", Wireless Networks, pp. 521-534, 2002.
[11]. Y. Sankarasubramaniam, O. B. Akan, and I. F. Akyildiz, "Esrt: event-to-sink reliable transport in wireless sensor networks", in MobiHoc '03: Proceedings of the 4th ACM International symposium on Mobile ad hoc networking & computing, pp. 177-188, ACM, 2003.
[12]. Q. Han, I. Lazaridis, S. Mehrotra, and N. Venkatasubramanian, "Sensor data collection with expected reliability guarantees", Pervasive Computing and Communications Workshops, pp. 374-378, March 2005.
[13]. L. B. Ruiz, I. G. Siqueira, L. B. e. Oliveira, H. C.Wong, J. M. S. Nogueira, and A. A. F. Loureiro, "Fault Management in Event-Driven Wireless Sensor Networks", in MSWiM '04: Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems, pp. 149-156, ACM, 2004.
[14]. [14] Ramakrishna Gummadi, Todd Millstein, and Ramesh Govindan, "Declarative Failure Recovery for Sensor Networks," AOSD '07, March 12-16 2007.
[15]. Youngbae Kim, James S. Plank, Jack J. Dongarra, "Fault Tolerant Matrix Operations for Networking of Workstations Using Multiple Checkpointings", High Performance Computing on the Information Superhighway, HPC Asia '97 IEEE, pp. 460-450, 1997.
[16]. Rana Ejaz Ahmed, and Abdul Khaliq, "On the Role of Base Station in FaultTolerant Mobile Networks", Electrical and Computer Engineering, Canadian Conference 2004, pp. 473-476, 2004.
[17]. Rana Ejaz Ahmed, and Abdul Khaliq, "A Low-Overhead Checkpointing Protocol for Mobile Networks", Electrical and Computer Engineering, IEEE CCECE 2003, pp. 1779-1782, 2003.
[18]. Yawei Li, Zhilling Lan, "A Fast Restart Mechanism for Checkpoint/Recovery Protocols in Networked Environments", Dependable Systems and Networks with FTCS and DCC, 2008, pp. 217-226, 2008.
[19]. Anas Abu Taleb, Dhiraj K. Pradhan, Taskin Kocak, "A Technique to Identify and Substitute Faulty Nodes in Wireless Sensor Networks", Sensor Technologies and Applications, SENSORCOMM'09,pp. 346-351, 2009.

Mohammed Bakhtawar Ahmed" Techniques for Fault Detection in Wireless Sensor Networks" International Journal of Engineering Science Invention (IJESI), vol. 07, no. 12, 2018, pp 65-71