A Review of Fault Detection Techniques for Wireless Sensor Networks

Er. Saurabh, Dr. Rinkle Rani Aggarwal

Department of Computer Science & Engineering, Thapar University, Patiala.

ABSTRACT

Today wireless sensor networks (WSNs) emerge as a revolution in all aspects of our life. WSNs have unique specifications of themselves that describe them different from other networks. Fault tolerance is one of the most significant of many challenges in these networks. Five key features need to be considered when developing WSN solutions: scalability, security, reliability, self-healing and robustness. In this paper the main objective is to provide a comparative study of fault detection techniques using different approaches. Sensor nodes have various energy and computational constraints. To provide quality service by coverage protocols, there arises a need for developing protocols to provide fault tolerance, event reporting, and maintain energy efficiency.

Key words, of the Abstract- wireless sensor network (WSN); fault tolerance; cluster head; fault tolerant systems; fault diagnosis;

1. INTRODUCTION

1.1. Motivation

The reliability of computer, communication, and storage devices was recognized in the initial times as one of the key issues in computer systems. Since the 1950's, techniques that enhance the reliability of computer and communication systems were developed both in academia and industry. It has been also recognized that as computers complexity and number of communication devices increases, fault-tolerance will be in great demand. Surprisingly, fault tolerance has never been the major design objective. While there are a number of reasons for this situation, the most important is that the reliability of individual components has been increasing at a much more rapid pace than it was expected. The rapid growth of the Internet in the last 10 years was the first major facilitator of the renewed interest in fault tolerance and related techniques such as self-repair. Internet requires the constant mode of operation and therefore special effort has been placed to develop fault tolerant data centers. Emergence of wireless sensor networks will further increase the importance of fault tolerance. At the same time, wireless sensor networks will impose a number of unique new conceptual and technical challenges to fault-tolerance researchers. There are at least three major groups of reasons why research in fault tolerant sensor networks should receive a significant attention.

The first one is related to the technology and implementation aspects. Two components of a sensor node, sensors and actuators, directly interact with the environment and will be subject to a variety of physical, chemical, and biological

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

forces. Therefore, lower intrinsic reliability is expected than integrated circuits in fully enclosed packaging. Wireless sensor networks will be often deployed as consumer electronic devices that will put significant constraints on the cost and therefore, quality of used components. More importantly, nodes operate under strict energy constraints that will make energy budget dedicated to testing and fault tolerance very limited. The second reason is that applications will be equally as complex as the involved technology and architectures. More importantly, sensor networks will often operate in an autonomous mode without a human in the loop. In addition, security and privacy concerns will often prevent extensive testing procedures. Lastly, and maybe most importantly, many applications of sensor networks will be safety critical and can have very adverse impact on humans and the environment, in particular when the actuators are used. The final reason is that wireless sensor networks themselves are a new scientific and engineering field and it is not still quite clear as to what is the best way to address a particular problem. At this level, it is also difficult to accurately predict the best way to treat fault tolerance within a particular wireless sensor network approach. Additionally, both technology and applications for wireless sensor networks are changing at a rapid pace. Therefore, with respect to fault tolerance, it is important to consider schemes that conduct error detection using only local information at their own level or, to design fault tolerant techniques that do not significantly increase the communication overhead. On the other hand if the computation energy is significantly higher than the communication requirements, it is a good idea to support communication resources at one node with the computation resources at other nodes. It is preferable to develop fault tolerant sensor fusion approaches that require little additional computation regardless of any additional communication requirements.

1.2 Sensor Network

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 possibly 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

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

nodes are anticipated. Such systems can revolutionize the way we live and work. Advances in sensor technology and wireless communications have enabled the design and development of inexpensive, large-scale sensor networks that are suitable for different applications, such as health monitoring, environmental monitoring, and battlefields surveillance. A fundamental aspect in the design of WSNs is to keep them functional as long as possible. Because of scarce battery power (or energy), sensors may entirely deplete energy or have remaining energy below some threshold that is required for the sensors to function properly. Those sensors are called faulty as they cannot perform any monitoring task properly. A WSN is said to be functional if at any time there is at least one communication path between every pair of non faulty sensors in the network. The existence of communication paths between pairs of sensors, however, is related to another fundamental property of WSNs, called vertex-connectivity (or simply connectivity). In general, sensing applications are required to be fault-tolerant, where any pair of sensors is usually connected by multiple communication paths. Therefore, network functionality and hence network fault tolerance strongly depends on connectivity. Figure 1 below represents the common architecture of Wireless Sensor Networks and their nodes.

The Wireless Sensor Networks are capable of sensing and forwarding the sensed data, and performing reactions based on received data appropriately. The WSN's consists of sensor nodes and sink nodes. The sensor nodes usually have low costs, limited energy supply and limited transmission range; they are responsible

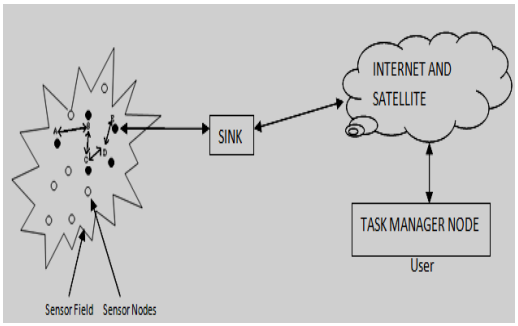

Figure 1: WSN communication Architecture

for detecting events or sensing environmental data. The sink nodes are resource-richer nodes with abundant energy sources, higher communication and computation capability, and the ability to perform powerful reactions. When the sink node performs some action then theses nodes are called actor nodes. When a sensor node detects some data to be delivered in its monitoring area, it will transmit the event to neighboring nodes, which in turn will forward the event one hop further. The hardware components of a sensor node have been shown in figure 2. In this way, the event reaches the sink. Once the sink node receives the data, it will perform corresponding reactions appropriately. WSNs enable some realistic applications, such as military, phenomenon monitoring, and attack detection [1].

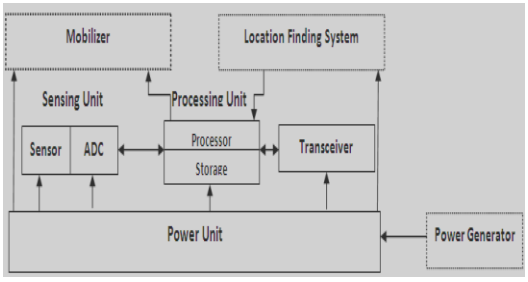

Figure 2: Components of a sensor node

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

Currently, wireless sensor networks are beginning to be deployed at an accelerated pace. It is not unfair to expect that in coming 10-15 years world will be covered with wireless sensor networks having access to them via the Internet. This can be equivalent being Internet becoming a physical network. This new technology is exciting with unlimited potential for numerous application areas including environmental, medical, military, transportation, entertainment, crisis management, homeland defense, and smart spaces. Since a wireless sensor network is a distributed real-time system a natural question is how many solutions from distributed and real-time systems can be used in these new systems? Unfortunately, very little prior work can be applied and new solutions are necessary in all areas of the system. The main reason is that the set of assumptions underlying previous work has changed dramatically. Most past distributed systems research has 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 each node in the system as very important and are location independent. In contrast, for wireless sensor networks, the systems are wireless, have scarce power, are real-time, utilize sensors and actuators as interfaces, have dynamically changing sets of resources, aggregate behavior is important and location is critical. Many wireless sensor networks also utilize minimal capacity devices which places a further strain on the ability to use past solutions. Even though sensor networks are a special type of ad hoc networks, the protocols designed for ad hoc networks cannot be used as it is for sensor networks due to the following reasons:

a) The number of nodes in sensor networks is very large and has to scale to several orders of magnitude more than the ad hoc networks and thus require different and more scalable solutions.

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

c) A sensor network is usually deployed by a single owner but MANET is usually run by several unrelated entities. [4]

d) Sensor networks are data centric i.e. the queries in sensor network are addressed to nodes which have data satisfying some conditions and unique addressing is not possible as they do not have global identifiers. But MANET is node centric, with queries addressed to particular nodes specified by their unique addresses.

e) Sensor nodes are usually deployed once in their life time and those nodes are generally stationary except a few mobile nodes, while nodes in MANET move in an ad hoc manner.

f) Like MANET sensor nodes are also designed for self configuration, but the difference in traffic and energy consumption require separate solutions. In comparison to ad hoc networks, sensor nodes have limited power supply and recharge of power is impractical considering the large number of nodes and the environment in which they are deployed. Therefore energy consumption in WSN is an important metric to be considered.

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

g) Sensor networks are application specific. One can't have a solution that fits for all the problems.

1.2.1 WSN Design Factors

There are number of design factors for designing an effective and efficient wireless sensor networks. Some of them have been discussed here: [1]

- Fault Tolerance
- Scalability
- Production Costs
- Hardware Constraints
- Sensor Network Topology
- Environment
- Transmission Media
- Power Consumption

1.3 Fault Tolerance

Fault-tolerance or **graceful degradation** is the property that enables a system (often computer-based) to continue operating properly in the event of the failure of (or one or more faults within) some of its components. Fault tolerance is the ability of a system to deliver a desired level of functionality in the presence of faults [8]. Nodes in WSNs are prone to failure due to energy depletion, hardware failure, communication link errors, malicious attack, and so on. If its operating quality decreases at all, the decrease is proportional to the severity of the failure, as compared to a naïvely-designed system in which even a small failure can cause total breakdown. Fault-tolerance is particularly sought-after in high-availability or life-critical systems. A WSN is said to be fault tolerant if it remains functional in spite of $\kappa - 1$sensor failures, where κ is network connectivity. Another important issue in the design of WSNs is what is called sensing coverage, a good indicator of the quality of surveillance of a field of interest [6]. Some sensing applications demand full coverage here every location in the field is covered by at least one sensor. Moreover, to cope with the problem of faulty sensors, duplicate coverage of the same region is desirable. Sensor redundancy is strongly related to the degree of sensing coverage requested by sensing applications, that is, the maximum number of sensors simultaneously covering any location in the field. Notice, however, that sensing coverage and network connectivity are not totally orthogonal concepts. While sensing coverage depends on the sensing range, connectivity relates to the communication range of the sensors. Sensing coverage becomes meaningless if the sensed data cannot be exchanged by the sensors so they reach a central gathering point, called the sink, for further analysis. Thus, for a network to function properly, both sensing coverage and network connectivity should be maintained.

Fault-tolerance is not just a property of individual machines; it may also characterize the rules by which they interact. For example, the Transmission Control Protocol (TCP) is designed to allow reliable two-way communication in a packet-switched network, even in the presence of communications links which are imperfect or overloaded. It does this by requiring the endpoints of the communication to expect packet loss, duplication, reordering and corruption, so that these conditions do not

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

damage data integrity, and only reduce throughput by a proportional amount.

1.3.1 Fault Tolerance at Different Levels

Five levels of fault tolerance were discussed in [14]. They are physical layer, hardware layer, system software layer, middleware layer, and application layer. On the basis of study, we classify fault tolerance in WSNs into four levels from the system point of view. More specifically, fault tolerance in a WSN system may exist at hardware layer, software layer, network communication layer, and application layer.

Hardware Layer

Faults at hardware layer can be caused by malfunction of any hardware component of a sensor node, such as memory, battery, microprocessor, sensing unit, and network interface (wireless radio).

Software Layer

Software of a sensor node consists of two components: system software, such as operating system, and middleware, such as communication, routing, and aggregation. Software bugs are a common source of errors in WSNs.

Network Communication Layer

Faults at network communication layer are the faults on wireless communication links. Link faults can be caused by surrounding environments or by radio interference of sensor nodes.

Application Layer

Fault tolerance can be addressed also at the application layer. For example, finding multiple node-disjoint paths provides fault tolerance in routing. The system can switch from an unavailable path with broken links to an available candidate path.

1.3.2 The Need for Fault Tolerant Protocols and Design Issues

Sensor networks share common failure issues (such as link failures and congestion) with traditional distributed wired and wireless networks, as well as introduce new fault sources (such as node failures). Fault tolerant techniques for distributed systems include tools that have become industry standard such as SNMP and TCP/IP, as well as more specialized and/or more efficient methods that have been extensively researched [14]. The faults in sensor networks cannot be approached in the same way as in traditional wired or wireless networks due to the following reasons:

a) traditional network protocols are generally not concerned with energy consumption, since wired networks are constantly powered and wireless ad hoc devices can get recharged regularly;

b) traditional network protocols aim to achieve point-to-point reliability, whereas wireless sensor networks are concerned with reliable event detection;

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

c) in sensor networks, node failures occur much more frequently than in wired, where servers, routers and client machines are assumed to operate normally most of the time; this implies that closer monitoring of node health without incurring significant overhead is needed;

d) traditional wireless network protocols rely on functional MAC layer protocols that avoid packet collisions, hidden terminal problem and channel errors by using physical carrier sense (RTS/CTS) and virtual carrier sense (monitoring the channel).

Many of the recent fault detection algorithms have either vaguely defined fault models or an overly general fault definition. [6], briefly listed selected faults, and develop a cross validation method for online fault detection based on very broad fault definitions. Looking beyond fault detection and correction techniques, there has been relevant work that frames our thrust to provide fault taxonomy.

1.3.3 Taxonomy of Fault Tolerant Techniques

Recent research has developed several techniques that deal with different types of faults at different layers of the network stack. To assist in understanding the assumptions, focus, and intuitions behind the design and development of these techniques, the taxonomy of different fault tolerant techniques used in traditional distributed systems [15] was given as:

a) Fault prevention: this is to avoid or prevent faults;

b) Fault detection: this is to use different metrics to collect symptoms of possible faults;

c) Fault isolation: this is to correlate different types of fault indications (alarms) received from the network, and propose various fault hypotheses;

d) Fault identification: this is to test each of the proposed hypotheses in order to precisely localize and identify faults;

e) Fault recovery: this is to treat faults, i.e., reverse their adverse effects.

Fault identification and isolation, sometimes are collectively referred to as fault diagnosis. Note that there do exist some techniques that address a combination of all these aspects. In fact, these techniques operate at different layers of the network protocol stack. Most fault avoidance techniques operate in the network layer, adding redundancy in routing paths; a majority of fault detection and recovery techniques operate at the transport layer; and a few fault recovery techniques perform at the application layer, concealing faults during online data processing.

2. RELATED WORK

2.1 Fault Detection: An Overview

Fault detection is the first phase of fault management, where an unexpected failure should be properly identified by the network system. The existing failure detection approaches in WSNs can be classified into two types: centralized and distributed approach.

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

2.1.1 Centralized Approach

Centralized approach is a common solution to identify and localize the cause of failures or suspicious nodes in WSNs. Usually; a geographically or logically centralized sensor node (in terms of base station [5, 17, and 18], central controller or manager [4], sink) takes responsibility for monitoring and tracing failed or misbehavior nodes in the network. Most these approaches consider the central node has unlimited resources (e.g. energy) and is able to execute a wide range of fault management maintenance. They also believe the network lifetime can be extended if complex management work and message transmission can be shifted onto the central node. The central node normally adopts an active detection model to retrieve states of the network performance and individual sensor nodes by periodically injecting requests (or queries) into the network. It analyzes this information to identify and localize the failed or suspicious nodes. In [17], the base station uses marked packets (containing geographical information of source and destination locations etc) to probe sensors. It relies on nodes response to identify and isolate the suspicious nodes on the routing paths when an excessive packet drops or compromised data has been detected. In addition, the central manager provides a centralized approach to prevent the potential failure by comparing the current or historical states of sensor nodes against the overall network information models (i.e. topology map, and energy map). As a summary, the centralized approach is efficient and accurate to identify the network faults in certain ways.

2.1.2 Distributed Approach

Distributed approach encourages the concept of local decision-making, which evenly distributes fault management into the network. The goal of it is to allow a node to make certain levels of decision before communicating with the central node. It believes the more decision a sensor can make, the less information needs to be delivered to the central node. In the other word, the control centre should not be informed unless there is really a fault occurred in the network. Others address the use of decision fusion centre (i.e. several fusion nodes across the network) to make the final decisions on suspicious nodes in the network [11, 12, 14, 16].

*** Node Self-Detection**

A self detection model to monitor the malfunction of the physical components of a sensor node via both hardware and software interface has been proposed by number of researchers. Self-detection of node failure is somehow straightforward as the node just observes the binary outputs of its sensors by comparing with the pre-defined fault models. In data dissemination protocols which deliver large segments of data to the entire (or part of the) network, the destination nodes are responsible for detecting the missing packet or the window of missing packets, and communicating the feedback to the source using NACK messaging.

*** Neighbor Coordination**

Failure detection via neighbor coordination is another example of fault management distribution. Nodes coordinate with their neighbors to detect and identify the network

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

faults (i.e. suspicious node or abnormal sensor readings) before consulting with the central node. For example, in a decentralized fault diagnosis system [12], a sensor node can execute a localized diagnosis algorithm in steps to identify the causes of a fault. In addition, a node can also query diagnostic information from its neighbors (in one-hop communication range). This allows the decentralized diagnostic framework to scale easily to much larger and denser sensor networks if required. Alternatively, suspicious (or failed) nodes can be identified via comparing its sensor readings with neighbor's median readings. With this motivation [9], developed a localized algorithm to identify suspicious node whose sensor readings have large difference against the neighbors. Although this algorithm works for large size of sensor networks, the probability of sensor faults needs to be small. If half of the sensor neighbors are faulty and the number of neighbors is even, the algorithm cannot detect the faults as efficient as expected. In addition, this approach also requires each sensor node to be aware of its physical location by equipped with expensive GPS or other GPS-less technology. [7, 8] address the accuracy of failure detection via a two-phase neighbor coordination scheme. Similar approach in [6], where a node can listen on its neighbor using WATCHDOG. If data packets have not been transmitted properly by the neighbors of a node it is currently routing to, fail or misbehaving neighbors can be easily detected.

*** Clustering Approach**

Clustering [14] has become an emerging technology for building scalable and energy balanced applications for WSNs. [18], derived an efficient failure detection solution using a cluster-based communication hierarchy to achieve scalability, completeness, and accuracy simultaneously. They split the entire network into different clusters and subsequently distribute fault management into each individual region. Intracluster heartbeat diffusion is adopted to identify failed nodes in each cluster. While, [13] adopt an event-driven detection via a manager-agent model supported by management architecture MANNA [3]. In this approach, agents are executed in the cluster-heads with more resources than common nodes. A manager is located externally to the WSN where it has a global vision of the network and can perform complex management tasks and analysis that would not be possible inside the network. Every node checks its energy level and sends a message to the manager or agent whenever there is a state change. The manager then uses this information to build topology map and network energy model for monitoring and detecting the potential failure of the network in future. Furthermore, random distribution and limited transmission range capability of common-node and cluster-heads provides no guarantee that every common-node can be connected to a cluster head. In addition, the transmission costs for network state polling has not been considered in this approach.

Distributed Detection

The basic idea of Distributed Detection is to have each node make a decision on faults (typically binary data of abnormal sensor reading). This approach is especially energy-efficient and ideal for data centric sensor

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

applications. However, there remain various research challenges in order to achieve a better balance between fault detection accuracy and the energy usage of the network. Usually, the efficiency of such failure detection schemes is counted in terms of node communication costs, precision, detection accuracy and the number of faulty sensor nodes tolerable in the network. In

Clouqueurs work [15], fusion sensors (in terms of manager nodes) coordinate with each other to guarantee that they obtain the same global information about the network before making a decision , as faulty nodes may send them inconsistent information.

3. CONCLUSIONS

Mobile computing is an emerging trend in distributed computing for several applications. The mobility of mobile hosts (MHs), limited battery power on MH, limited wireless bandwidth, noisy wireless environment, handoff, and limited (or lack stable storage on MH present challenging problems in providing fault-tolerance to such mobile computing systems. Due to the potential deployment in uncontrolled and harsh environments and due to the complex arch, wireless sensor networks are and will be prone to a variety of malfunctioning. The goal of this paper is to identify the most important types of faults, techniques for their detection and diagnosis, and to summarize the first techniques for ensuring efficiency of fault resiliency mechanisms. In addition to a comprehensive overview of fault tolerance techniques in general, and in particular in sensor networks, techniques that ensure fault resiliency during sensor fusion as well as the approach for heterogeneous built-in-self-repair fault tolerance were also discussed.

Figure 3: Comparative Chart for Existing Fault Detection Techniques in Wireless Sensor Networks

Name of Technique	Working Principle	Advantages	Disadvantages
On-line 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 [5]	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 [6]	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 [17]	Only requires the touch set on the destination node for quick restart, the remainder of image is transferred after process is restarted on destination.	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.

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

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. Cardell-oliver, “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:

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.

Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems, pp. 149-156, ACM, 2004.

[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 Fault-Tolerant 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.

Copyright (c) 2013 International Journal of Computer Science Issues. All Rights Reserved.