Cluster Based Fault Detection and Diagnosis Scheme for Wireless Sensor Network

Deviprasad Mishra[1] and Ramesh Kumar[2]
[1]Research Scholar, CSVTU Bhilai, CG, India
[2]Professor Department of Computer Science & Engineering, CSVTU
[1]mishradprasad@gmail.com, [2]rk_bitd@rediffmail.com

Abstract

Wireless Sensor Network (WSN) is emerged as revolution in all the aspect from past few years, WSN gained attention of lots of researchers for using them in different applications. WSN is having unique specification of their own that distinguishes them from other network. Fault tolerance is one of the most significant and challenging area for WSN, since sensor nodes are prone to various types of attacks and failures due to hardware, battery power, malicious attacks, etc. Faulty sensors are likely to report arbitrary readings that do not reflect the true state of observed physical process. These faulty sensors nodes should be recognized and timely excluded from the data collection process in order to ensure the overall data quality, so while designing and developing WSN based solutions, it is highly recommended to accomplish five key features in WSN solutions: scalability, security, reliability, self-healing and robustness. This paper will discuss different mechanisms used for fault detection, fault recovery in WSN, and propose cluster based recovery technique.

Keywords: Wireless Sensor Networks, Fault Recovery algorithm, Data Fault Detection, Functional Fault, Cluster Head

1. Introduction

Wireless Sensor Networks have emerged as an important new area in wireless technology. A wireless network consisting of tiny devices, which monitor physical or environmental conditions such as temperature, pressure, motion or pollutants etc. at different areas. Such networks may be used for variety of applications like environmental, commercial, civil, military applications such as surveillance, vehicle tracking, climate and habitat monitoring, intelligence, medical, and acoustic data gathering. The key limitations of wireless sensor networks are the storage, power and processing [1]. These limitations and the specific architecture of sensor nodes call for energy efficient and secure communication protocols. The key challenge in sensor network is to maximize the lifetime of sensor nodes and the accuracy of data is very important to the whole system's performance, detecting faulty node is main challenge in network management. The accuracy of individual node's readings is crucial; the readings of sensor nodes must be accurate to avoid false alarms and missed detection. There are certain applications, which are designed to be fault tolerant to some extent, by removing faulty nodes from a system with some redundancy or by replacing them with good ones, will significantly improve the whole system's performance and prolong the lifetime of the network. To overcome the burden of after deployment maintenance (e.g., remove and replace), it is essential to investigate methods for detecting faulty nodes.

2. Faults in Wireless Sensor Network

Wireless sensor networks consist of a large number of tiny sensor nodes deployed in harsh environment for unattended operation to sense and forward some data to base station through single-hop or multi-hop transmission since sensor nodes have self-organized capabilities[2]. Since most of the sensor network operates in unattended environment, there is the possibility of fault due to hardware failure, energy utilization, security attacks and signal strength / signal obstacle [3]. **Fault** is an unintended defect that ultimately channelizes to the cause of an error. **Error** is an indication of false (incorrect) state of the system. Imperfection quality of the system state caused by error, ultimately leads to the failure. A **failure** is the condition where the system becomes ineffective to perform the intended regulated functionalities, due to error.

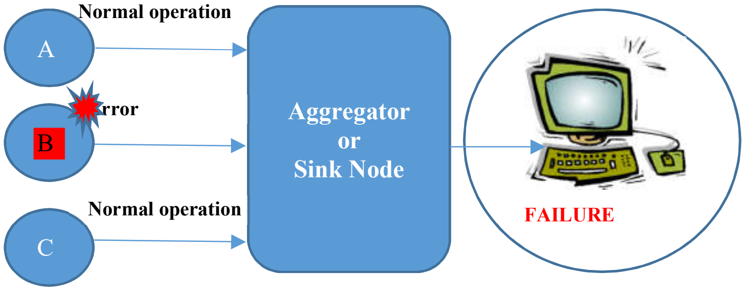

Figure 1. Relation between fault, error and failure

Fig. 1 depicts the basic difference between fault, error, and failure. The principle operation of sensor node A, B and C are reporting periodical sensed data to the gateway node, which aggregates different generic sensor data's for future analysis. Each sensor service is normal until node B suffers a fault. Thus, the immediate occurrence of fault (any) causes an error in performing normal service by node B. Due to the occurrence of fault on node B, it provide an errored service to the gateway node. These errored services contain inappropriate information for the analysis of entire application/system. The faulty service provided by node B results as cause of system failure. Fig. 2 shows sensor form, in which node no.13 is not responding, that isolates other part of network that results in collapse of application.

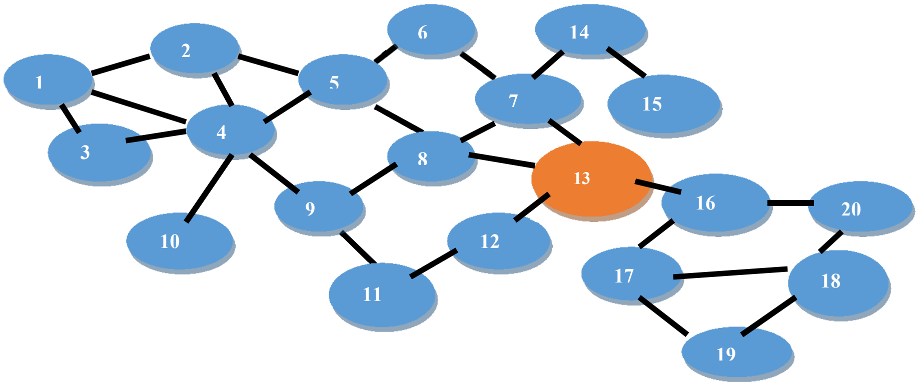

Figure 2. Network examples with faulty/failure

2.1 Representation of WSN Fault

A system is said to fail when it cannot meet promises. In particular, if a WSN system which is designed to provide number of services to the intended but its not meeting desired expectation or not providing the result, the system is said to be failed when one or more of those services cannot be (completely) provided. An error is a part of a system's state that may lead to a failure. Fig. 3 is shows the stages of fault.

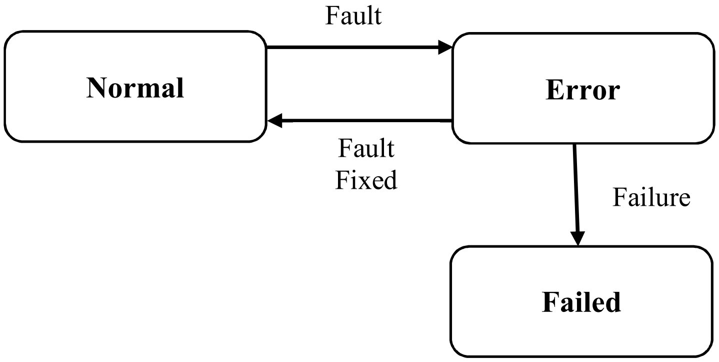

Figure 3. Representation of fault

The identification of a faulty member node in WSN, who is not delivering the promised services within period. There are two main ways to detect a fault in a distributed system: active and passive. The former is sending "AreYouAlive" messages to each other and the latter is waiting until a process send you a message informing that it is still alive. In general, there are many problems lying in the fault detection. One of them is the attempt to reduce the generation false positives. The false positive usually are generated by using a timeout mechanism in an unreliable network. Another issue is that the aforementioned fault detection mechanisms do not provide enough information about the fault.

2.2 Types of Faults

Almost all the WSN researchers are asking a common typical question - "What will be the most vigorous causes and deep impact of fault on WSN?" There are different possible answers for this question. From [4], it's conceptually expressed or assumed that under any circumstance, entire functionality of WSN should not be disturbed as a whole in order maintain and ensure high reliability. First step to build a WSN fault tolerant system will closely relate various faults; inspect the variety and nature of faults. WSN faults are categorized into three major categories and they are Sensor reading faults, Software faults and Hardware faults. Each of these categories are elaborately depicted in Fig. 4.

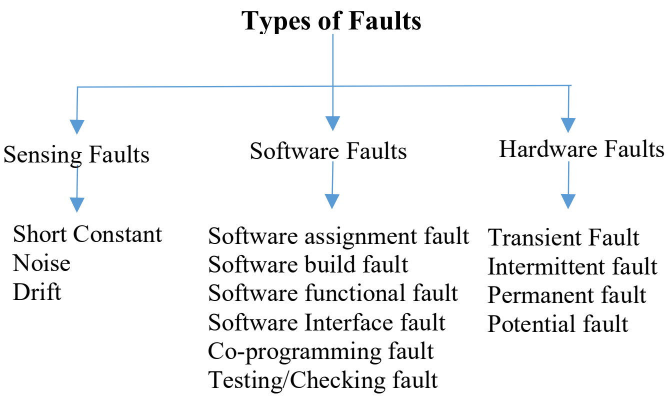

Figure 4. Network examples with fault node

2.3 Generic lifecycle of Fault Tolerance:

Increasing fault tolerance potentiality of WSN depends on continuous well-organized multi-operational procedures of three phases (prevention, diagnosis and recovery), that are involved in FT management. On following analysis with three phases, a generic lifecycle has been furnished, which is depicted in Fig. 5.

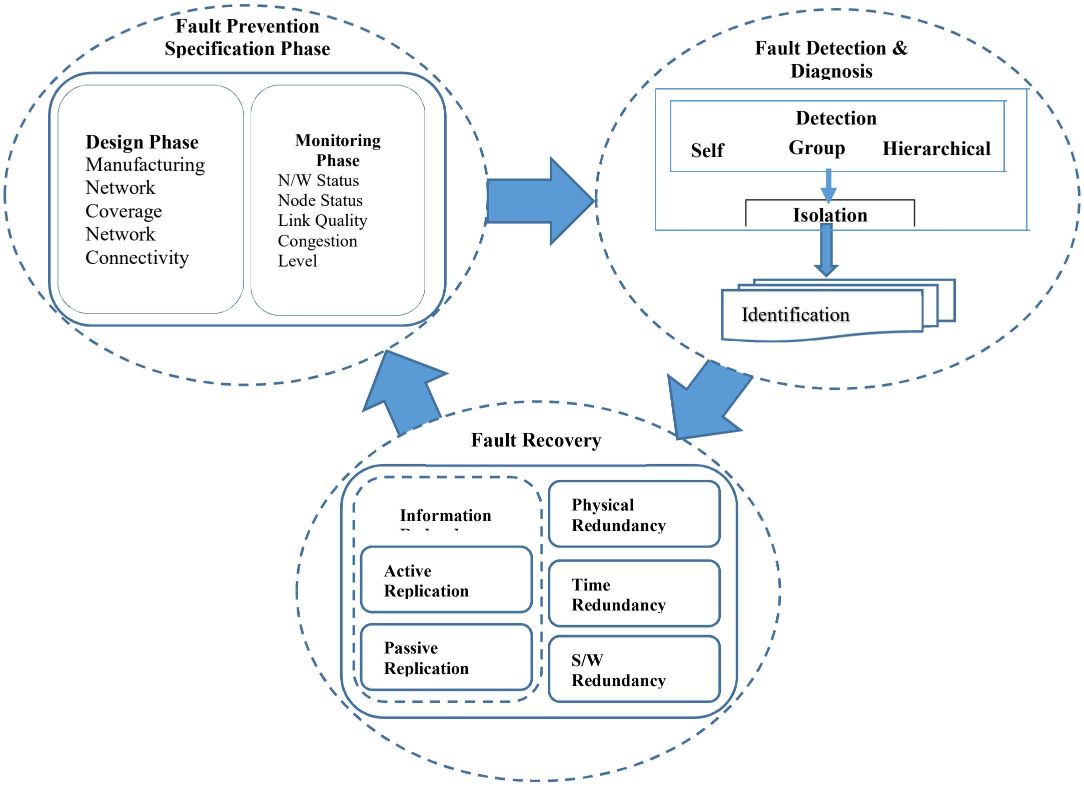

Figure 5. A generic life cycle of Fault Tolerance

3. Basics of Clustering

Low-energy adaptive clustering hierarchy (LEACH) [5] is, a clustering based protocol that includes the features like –

- Randomized adaptive self-configuring cluster formation.
- Localized control for data transfers.
- It reduces the energy required for media access and data processing task like aggregation.

LEACH randomly selects a few sensor nodes as cluster heads (CHs) and rotates this role to evenly distribute the energy load among the sensors in the network. The entire iteration specific to selection of CHs is called a round. The operation of LEACH is split into two phases: Set up & Steady

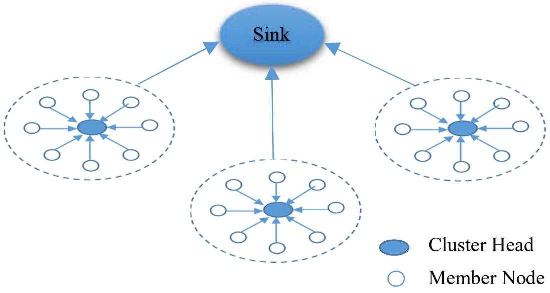

Figure 6. Leach Architecture

During the setup phase, a predetermined fraction of nodes, p, elect themselves as CHs as follows. Sensor node selects a random number, r, between 0 and 1. If the selected random number is less than a threshold value, T(n), then the concern node becomes a cluster-head for the current round. Threshold value T (n) is calculated with formula given below.

$$T(n) = \frac{P}{1 - p(r \, mod \left(\frac{1}{p}\right))} \quad if \; n \; \in G$$

Where

P: is the desired percentage of nodes, which are CHs,
r: is the current round, and
G: is the set of nodes that has not been CHs in the past 1/P rounds.

During steady state phase, data transmission takes place based on TDMA schedule and the CHs perform data aggregation through local computation. The BS receives only aggregated data from cluster-heads, leading to energy conservation. After a certain time, the network goes back into the setup phase again and enters another round of selecting new CH. Each cluster communicates using different CDMA codes to reduce interference from nodes belonging to other clusters.

3.1 Proposed Mechanism

There are four phases in this scheme – Advertising, Data Transmission, Fault Detection and Fault Recovery, which is depicted in Fig. 7.

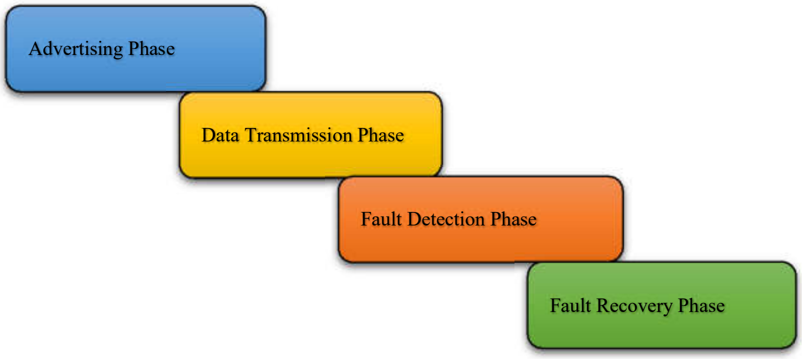

Figure 7. Four Phases of proposed Mechanism

As shown in Fig. 7 in First phase i.e. advertising phase, the clusters are prepared and selection of cluster heads (CHs) is done. After selection, the CHs advertise their selection to all neighboring or remaining nodes[6]. All concerned nodes select their nearest CH based on the received signal strength during advertisement. Later on concern, CHs assign a TDMA schedule to their cluster members.

The second phase, data transmission phase, all subordinate nodes can start sensing and transmitting data to the cluster-head. After receiving data, the cluster-head aggregate it before sending it to the Base-Station (BS).

The third phase is the fault detection phase. In hostile environments, unexpected failure of CH may partition the network or degrade application performance. If no response comes from CH to BS or subordinate nodes within a time interval, BS marks or put flag for concern

CH as a faulty node and forwards this information to the rest of the network and initiate fault recovery process.

In the final phase, cluster head immediately starts fault recovery process after detection. When a faulty CH node is identified, all the cluster members associated with it are gradually informed about the CH failure. For the CH recovery operation, the sink node chooses a new CH from the cluster members, based on cluster member's sensor nodes residual energy. According to this scheme, simply replace the faulty cluster-head by the next highest energy node in the cluster.

3.2 Fault Detection Algorithm

```
Step1. Initialize CH1 & CH2 & subordinates
Step2.  IF no response comes within a TDMA slot Then
Step3. Set CH1 as Faulty Step Else
Step4. For CH2
Step5. IF no ping message comes periodically Then
Step6. Set CH2 as Faulty
```

3.3 Fault Recovery Algorithm

```
Step1.   Start
Step2.   Initialize CHs & subordinates
Step3.   Compare residual energy of current CH (CHR) and each subordinate in the
cluster.
            IF CHR less than each subordinate,      then
            Replace CHR with next highest energy node.  Else
            Set CHR as CH for next setup round.
Step4.   Stop.
```

3.4 Performance evaluation

The energy model used is a simple model shown in transmitter, receiver dissipates energy to run the power amplifier to run the radio electronics. In the simple radio model [7], the radio dissipates Eelec = 50 nJ/bit to run the transmitter or receiver circuitry and Eamp = 100 (pJ/bit)/m2 for the transmit amplifier in-order to get acceptable signal-to-noise ratio. We have used MATLAB Software as the simulation platform[8] and utlised simulation parameters specified in Table2

Table 2 – Simulation Parameters

Simulation Parameter	Value
Terrain Dimension	1 KM^2
Total number of nodes in terrain, N	100 – 1500
Transmission range	100 – 450m
Cluster size limit, s	10 – 50
Supported degree, D	3 – 10

3.5 Characteristics of the Clusters

Fig. 8 depicts the percentage of cluster heads observed with varying cluster range. The cluster range was varied from 200 to 400. The size limit, S in our algorithm was set to 50 with admissible degree, D set to 3. The percentage of cluster heads was observed and noted for about 10 runs of the clustering algorithm. The percentage of cluster heads does not increase or decrease over various rounds of the algorithm. This is because for a total number of N nodes in terrain, the limit S is set to 50 leading to N/50 cluster heads or clusters. Due to this limitation the results do not having variation in terms of decrease or an increase in the cluster heads. Even though the percentage of cluster heads is not changing, the responsibility of cluster head is delegated or exchanged with the nodes in the network

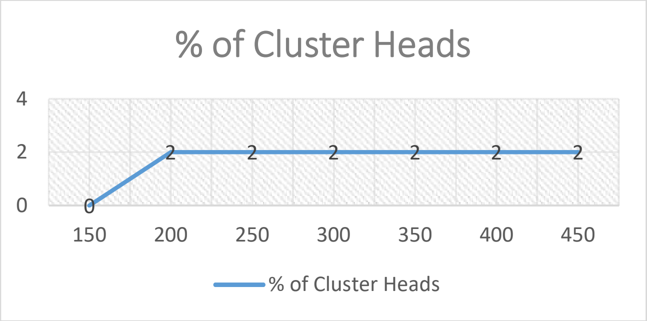

Figure 8. Percentage of cluster heads observed with varying cluster range

3.6 Energy characteristics in Clusters Fig. 9 depict the energy drain during the cluster formation. Energy drain is the loss of energy in all the node after cluster formation and operation. Energy loss is based on the relation in the first order radio model. Total energy loss would be the energy loss due to transmission added to the loss due to receiving. Energy utilization depends on parameters used in first order radio model, distance and the number of bits, k. Energy consumption is also dependent on the no. of concerned nodes i.e. transmitting to and receiving from. In clustering algorithm the distance is sensing range, which is about 50 % of the transmission range. Also the number of nodes each node would handle is D. These two factors make energy loss regular and uniform

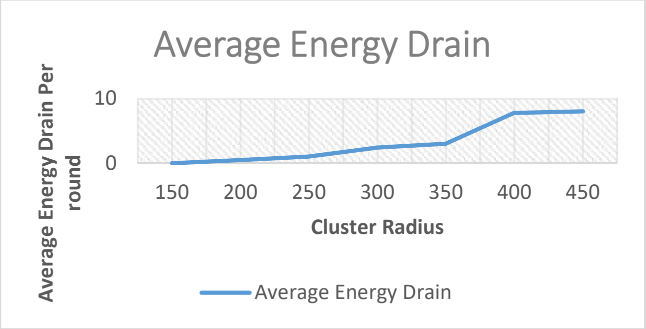

Figure 9. Ratio of average balance energy drain per round with varying cluster radius

4. Evaluation of proposed Algorithm

We compared our work with that of algorithm [9], which is based on recovery due to energy exhaustion. Where the nodes in the cluster are categorized in four categories: boundary node, pre-boundary node, internal node and the cluster head. Boundary nodes does not require any recovery but pre-boundary node, internal node and CH will take appropriate actions to connect the cluster. Usually, if node energy becomes below a threshold value, it will send a fail_report_msg to its parent and children. This will initiate the failure recovery procedure in order to maintain the connectivity of failing node parent and children to the cluster. A join_request_mesg is sent by the healthy child of the failing node to its neighbors. All the neighbors with in the transmission range respond with a join_reply_mesg/join_reject_mesg messages. The healthy child of the failing node selects a suitable parent by verifying that selected neighbor is not one among the children of the failing node.

In proposed mechanism, normal nodes does not require any recovery but they switch them-self to lower computational mode by informing their cell managers. In existing algorithm [10], CH failure results in children to exchange energy messages. Important aspect over here is failed children are not considered for the new cluster-head election. The healthy node/child with the maximum residual energy is selected as the new cluster head and and responsible for sending a final_CH_mesg to its members. After the new cluster head is selected, the other children of the failing cluster head are attached to the new cluster head and new CH becomes the parent for these children. CH failure recovery procedure requires more messages to be exchanged to select the new cluster head that require more energy to exchange series of messages. Also,in case of failing CH require appropriate steps to get connected to the cluster, which is time consuming as well abrupt network operations. In our proposed algorithm, back up secondary cluster heed is employed which replace the cluster will heed in case of failure.

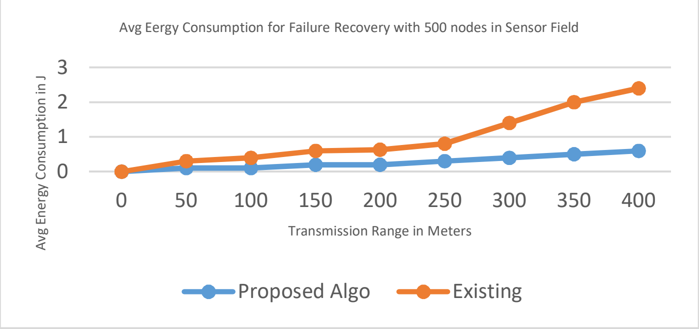

Figure 10. Average time for cluster head recovery

No further messages are required to send to other cluster members to inform them about the new cluster heed Fig. 10 and 11 compare the average energy loss during failure recovery of different algorithms. It can be observed from Fig. 10 that when the transmission range increases, after analyzing the greedy algorithm with Gupta algorithm [11] and the proposed algorithm it observed that greedy algorithm expends the maximum energy. However, from Fig. 11, we may say that the Gupta algorithm spends the more energy as compared to other algorithms when the number of nodes in sensor field increases.

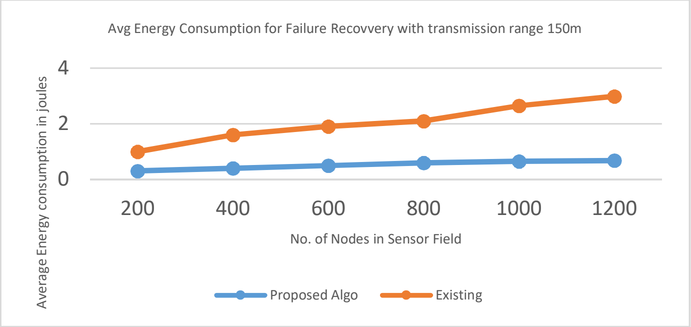

Figure 11. Average time for cluster head recovery

5. Conclusion and Future Work

In this paper, we have explained about the issues specific to network disruption due to cluster head failures in wireless sensor networks and we have tried to find a solution for that. We have proposed a fault management mechanism for wireless sensor network to diagnose faults, respond appropriately to recover sensor network from failures. We have compared our algorithm with the algorithm, is recent approach of fault detection and recovery in wireless sensor networks and proven to be more efficient than few existing algorithms. It is more energy efficient when compared with Gupta and Greedy Therefore; we conclude that our proposed algorithm is also more efficient than Gupta and Greedy [11] in term of fault recovery. The faster response time of proposed algorithm provides uninterrupted operation and healthy lifetime for the prolonged operation of the WSN. In future, we would incorporate the mobility and autonomic fault management aspect in the context WSN fault tolerant system

References

[1] K. Xie *et al.*, "An efficient privacy-preserving compressive data gathering scheme in WSNs," *Inf. Sci. (Ny).*, vol. 390, pp. 82–94, 2017.

[2] T. Bokareva, N. Bulusu, and S. Jha, "SASHA: Toward a self-healing hybrid sensor network architecture," *Second IEEE Work. Embed. Networked Sensors, EmNetS-II*, vol. 2005, pp. 71–78, 2005.

[3] K. E. Ackermann and C. Nita-rotaru, "Mitigating Attacks against Virtual Coordinate Based Routing in Wireless Sensor Networks," 2008.

[4] D. Kim, Y. Kim, D. Li, and J. Seo, "A new maximum fault-tolerance barrier-coverage problem in hybrid sensor network and its polynomial time exact algorithm," *Ad Hoc Networks*, vol. 63, pp. 14–19, 2017.

[5] C. Chen and T. Chan, "An Efficient Non-ACK Routing Protocol in Wireless Sensor Network," no. 1, pp. 2–4, 2008.

[6] S. Dasgupta and P. Dutta, "A Novel Game Theoretic Approach for Cluster Head Selection in WSN," no. 3, pp. 40–43, 2013.

[7] I. Chatzigiannakis, T. Dimitriou, S. Nikoletseas, and P. Spirakis, "A probabilistic algorithm for efficient and robust data propagation in wireless sensor networks," *Ad Hoc Networks*, vol. 4, no. 5, pp. 621–635, 2006.

[8] D. P. Mishra, S. Csvtu, and R. Kumar, "Qualitative Analysis of Wireless Sensor Network Simulators," *Int. J. Comput. Appl.*, pp. 975–8887, 2015.

[9] M. H. Megahed, D. Makrakis, and B. Ying, "SurvSec: A new security architecture for reliable network recovery from Base Station failure of surveillance WSN," *Procedia Comput. Sci.*, vol. 5, pp. 141–148, 2011.

[10] V. Mhatre and C. Rosenberg, "Design guidelines for wireless sensor networks: Communication, clustering and aggregation," *Ad Hoc Networks*, vol. 2, no. 1, pp. 45–63, 2004.

[11] L. Schwiebert, S. Fowler, and S. K. S. Gupta, "e3D : An Energy-Efficient Routing Algorithm for Wireless Sensor Networks."