**A HYBRID FAULT-TOLERANT ROUTING BASED ON GAUSSIAN NETWORK FOR WIRELESS SENSOR NETWORK**

**CHAPTER -1**

**ABSTRACT**

In this paper, we have proposed a hybrid fault tolerant routing to solve fault-tolerant issue in wireless sensor networks (WSNs) based on hierarchical topology. The hierarchical topology is a combination of clustering and the labelling of sensor nodes as Gaussian integers. Accordingly, the network area is divided into small square grids, the cluster head of each grid is represented by a Gaussian integer. These cluster heads are connected together to create a Gaussian network. Through node symmetry, and the shortest distance in the Gaussian network, as well as the advantages of multi-path routing, this paper proposes a hybrid fault-tolerant clustering routing protocol based on Gaussian network for wireless sensor network (FCGW). The purpose of FCGW is to improve fault tolerance, increase data reliability and reduce energy consumption for wireless sensor networks. The experimental results of the proposed scheme show that FCGW protocol has high data reliability.

Keywords—Clustering, fault-tolerance, Gaussian network, multi-path routing, wireless sensor network.

**CHAPTER-2**

**INTRODUCTION**

I N wireless sensor network (WSN), the metrics such as data reliability, optimal energy consumption, memory limit, and data latency are major challenges in effective implementation of network [1]–[3]. In particular, in WSN, due to limited resource of sensor nodes, as well as harsh communication environments such as rain, wind, snow and water, always leads to faulty connections as in [4]. Therefore, improving fault tolerance in the network will improve the quality of service and availability of the WSN. Since then, addressing fault tolerance issues has been an important requirement in WSN design. Currently, there are several fault-tolerant techniques that have been proposed to improve fault tolerance. These fault tolerant techniques will be classified into three main categories: redundancy based mechanisms, clustering based mechanisms, and deployment-based mechanisms as in [5]. Specifically, each mechanism has different advantages and disadvantages - Redundancy based mechanisms: The fault tolerance is improved by redundancy factors such as redundant routing paths, redundancy time, redundant data [6]. In this mechanism, the main fault recovery approach includes [7] active replication and passive replication. In active replication, all fault recovery requests will be processed by all replicas. On the other hand, in passive replication, a fault recovery requests will be processed by a single replica. If this replica fails, another replica will be continued. Therefore, this mechanism increases data recovery capacity and the data reliability. However, the cost of deployment and maintenance of the network is very large. - Clustering based mechanisms: This mechanism will create small clusters, making it easy to manage and expand the network [8], [9]. Therefore, this mechanism avoids the random failures of nodes as well as improves energy efficiency through the cluster head node (CH). However, when a CH fails, it will affect its cluster. So, improving the fault tolerance efficiency of CH is essential. Based on the redundancy of CH or re-election of CH after a period, several approaches have been proposed for CH fault tolerance, lead to more energy consumption and increased data latency. - Deployment based mechanisms: WSNs are self-organizing networks that are randomly deployed or pre-located. Hence, it is crucial to evade connection failure to maintain a strong network topology which is essential to avoid connection failure [10]. Network topology optimizations based on virtual backbone is an example to avoid connection failure using the connected dominating set (CDS) [11]. This mechanism optimizes data transmission by a virtual backbone, avoids information interference and reduces energy loss. Even though optimizing the network structure can solve the NP-hard problems, this will increase the complexity of the solution. Through the summary above, it has been shown that the optimizing the network topology and clustering efficiency have more advantages for fault tolerance in the wireless sensor networks. In addition, our previous research [12] has proposed a modelling of WSN based on a Gaussian network to optimize the management of network topology as well as improve energy consumption by the shortest path routing protocol. Therefore, this paper continues to expand previous research and propose a solution using the Gaussian network model to solve fault tolerance issue in WSN. The main fault tolerance idea in this proposal is to exploit the efficiency in clustering routing and symmetry of Gaussian network based on multipath routing .

Our main contributions are as follows:

• The random sensors will be divided into virtual grids (clusters) using geographic adaptive fidelity (GAF) algorithm [13]. Each cluster will select one active node as a CH node at a time to improve consumption energy efficiency and prolong network lifetime.

• Each CH node will be represented as a Gaussian integer and connected together to form a Gaussian network, so WSN-management will be optimized as a Gaussian network.

• This paper proposes a fault-tolerance mechanism for CH nodes. Based on the symmetric link in the Gaussian network, if the main routing path fails, it is easily replaced by the redundant routing paths, thus increasing the data reliability as well as optimized energy consumption.

1. **Comprehend Fault Tolerance**

Fault tolerance is one of the important characteristics of wireless sensor networks. Accordingly, fault tolerance is improved to ensure the reliability and availability of the network in the harsh environment and limited energy of the nodes. Therefore, in order to classify and understand fault tolerance mechanisms, some related concepts have been defined in the previous literature [4], [7], [14], [15], [16], as follows:

• **A Fault**: A fault is an unexpected change or abnormal condition in a system. It is any kind of defect that leads to an error. A fault in any component of a system can lead to a system failure.

• **Fault detection**: To provide any countermeasures, the first step a system must perform is to detect that specific faults occurred in the network.

• **Fault diagnosis:** Is the process of determining the factors affecting the faults and diagnosing the types of causes that result in faults.

• **Fault recovery:** After the system has detected a fault, the next step is to prevent further faults or recover from the detected fault. The main technique to achieve this goal is to replicate components when an error occurs in the system. In WSN, the fault occurs for a variety of reasons in network due to in hardware, software, communication environment, power limit, timeout, malicious nodes etc. On the other hand, based on fault detection and fault diagnosis approaches [15], [17], the fault tolerance approaches will be divided into three types as below:

• **Centralized approach**: This approach is mainly based on the use of a centralized node to perform fault management. The status of all nodes in the network will be determined by this centralized node by receiving and processing information from all nodes. Therefore, the centralized approach is ineffective in large-scale WSNs. Accordingly, the energy consumption for network management and data transmission is huge, as well as increasing data latency, which greatly affects the efficiency of fault detection and fault recovery in WSNs.

• **Distributed approach**: In this approach, the status of each node will be determined by collecting and analysing information from neighbouring nodes. The distributed approach will reduce data transmission to the central node, which helping to reduce energy consumption, prolong lifetimes and reduce data latency. Therefore, this approach can be effectively applied to large-scale WSNs. However, this approach depends heavily on the density of neighbouring nodes. Because over time the node density decreases, which inevitably has a negative effect on fault detection accuracy.

• **Hybrid approach**: The main idea of this approach is to rely on both the centralized node to collect information and the neighbour node to exchange information to determine the state of the nodes. The hybrid approach addresses the large-scale network problem of centralized methods and node density problem in distributed approach by adding additional equipment (super nodes, mobile sink). However, installing additional equipment will increase costs and make it difficult to deploy actual networks. According to these fault tolerance approaches, there are several fault routing protocols have been proposed to achieve greater fault recovery efficiency such as clustering based protocols; multi-path based protocols; hybrid based protocols. The details of some common fault tolerance routing will be discussed in the next section.

1. **The Fault-Tolerant Routing in Literatures**

In this section, some fault-tolerant routing protocols will be introduced in detail. These routing protocols will be incorporated in our proposal as well as during the performance evaluation process in Section V.

1. **Fault-tolerant Clustering Routing Protocols**: In WSN, the clustering protocol has proven to be an energy-efficient protocols as in [18]. In addition, clustering protocols also greatly improve the fault tolerance such as several routing protocols FT-LEACH [20], HEED [21], and FTS [22], etc.

• The FT-LEACH protocol reduces the CH faults in LEACH [19] by randomly selecting the CH nodes based on the residual energy of the nodes. In FT-LEACH, the main idea for fault tolerance is that each common member (CM) node will only send data to CH when its value differs from the value of the previous time. The advantage of this proposal is that it avoids the CH nodes processing repetitive data and prevents the CH node from failing due to energy outage. However, the author does not clearly explain the energy consumption parameters for the data exchange.

• The HEED protocol periodically selects CH nodes based on a hybrid of the residual energy of nodes and a secondary parameter, such as node proximity to its neighbours or node degree. Each node will calculate a CHprob (CH probability) based on the residual energy of the node to elect as CH nodes. The CHprob helps the HEED protocol to avoid the election of low energy nodes as CH which leads to the network that will quickly lose connection. However, HEED depends on the density of high-energy nodes in the network, so if the fault nodes density is increased, it will lead to low fault tolerance performance.

• The FTS protocol is proposed for fault diagnosis and fault recovery. In FTS, each cluster will elect a spare CH (SCH) node based on the remaining energy of nodes and its distance to the CH node to fault diagnosis of CH node. This algorithm is highly effective in fault tolerance. However, due to FTS dependency on information exchanged between nodes, therefore, over time, reducing the node density will lead to a decrease in routing performance.

1. **Fault-tolerant Multi-path Routing Protocols**: Improving fault tolerance in wireless sensor networks based on multi-path routing is an important approach. The multi-path routing will build a few routing paths from source node to sink node. After that, the data will be transmitted through one of these paths, leading to, the data reliability can be increased. In case the primary routing path fails, it can be replaced by another path [23]. This is a good approach to fault tolerance. The benefits of multi-path routing in WSN such as fault tolerance, load balancing, quality of service, etc. are also mentioned in the literature [24] to [26]. The fault tolerance based on multi-path routing can be mainly divided into three categories as in [24].

• Alternate path routing: a few multi-path routing have been proposed for alternative path routing as in literature [27], etc. The purpose of these protocols is to ensure that multiple paths are redundant. The multiple paths are built by the sink node to broadcast messages to other nodes in the network. Therefore, these protocols rely heavily on sink nodes resulting in low efficiency and passive construction of alternatives. • Reliable data transmission: the protocols have proposed to increase the reliability of data by creating redundant packets during data transfer as in [28], [29]. The advantage of these protocols is that it does not maintain routing table. However, the routing costs and the delay time will increase to determine redundancy paths in case of a fault. • Efficient network resource utilization: the protocols such as HEED-FT [30], I2MR [31], etc., have introduced multi-path protocols for resource efficiency. Accordingly, routing maintains some redundant routing paths in some cases to balance resource efficiency and restore routing path when a disconnection occurs on the primary path.

1. **Hybrid Fault-tolerant Routing Protocols**:

Multi-path routing improves data reliability and optimizes fault connections, while clustering ensures high energy efficiency and prolongs network life. Therefore, several fault-tolerant routing protocols have been proposed based on a combination of multipath routing and cluster routing. A protocol called robust energy-efficient distributed clustering (REED) for the k-connectivity network has been proposed in [32]. The benefit of REED is that it finds and replaces the faulty connection quickly and simply. So, that it optimizes the cost of restoring fault connections. However, this algorithm applies only to the CH node, the CM node fault has not been addressed in this proposal. In addition to this proposal depends on the density of neighbouring nodes and the high cost of maintaining other routing paths in the network. The authors in [30] have presented a Fault-tolerant multipath routing protocol for WSN based on HEED (HEEDFT). The aim of HEED-FT is to increase data reliability and load balancing based on multi-path routing and clustering. In HEED-FT, to improve fault tolerance, each cluster will select a deputy cluster head node (SCH) based on the energy weighted centre of parameters (EWNC, energy-weighted node centrality). Accordingly, the data will be transmitted on one of two paths established based on SCH or CH. However, the maintenance of SCH nodes, as well as highly complex algorithms results in lower energy efficiency over time.

1. **The Gaussian Network and Benefits**

The authors in [34][35] introduced a Gaussian network (Gaussian graph), in which nodes are labelled using a subset of the complex numbers. The Gaussian network is represented by a set of Gaussian integers Z[i] = {x+yi|x, y ∈ Z}, where Z is the set of integers and i 2 = −1, that is defined as follows: Definition: Let 0 6= α = a + bi ∈ Z[i], Z[i]α is set of Gaussian integers modulo α, we define the Gaussian network as Gα(V, E) where: 1) V = Z[i]α is the node set, 2) E = {(β, γ) ∈ V × V |(β − γ) ≡ ±1, ±i( mod α)} is the edge set, where β, γ are two Gaussian integer. We call Gα the Gaussian network generated by α. In a Gaussian network, every node is connected to four adjacent nodes in four different directions: North, East, South, and West. In addition, the adjacent links are undirected and node symmetric. In this paper, we represent a Gaussian network Gα with α = a + bi, 0 ≤ a ≤ b and the greatest common divisor gcd(a, b) = d > 1 as a mesh network in a rectangle. In the case gcd(a, b) = 1 is presented in [34]. Accordingly, the nodes in Gα are arranged in a rectangle of size (d × f) where f = (a 2+b 2 )/d. Node 0 is located at the bottom left of the graph shown. The connections between adjacent nodes are explained in Theorem 2.2 of the literature [35]. In the Gaussian network Gα=a+bi, for two nodes β = x1 + y1i and γ = x2 + y2i. The distance between β and γ is given by formula (1). For Example, in Gaussian network G4+4i , and β = 0+ 0i, γ = 1+ 1i, then D(0, 1+i) = 2, since D(0, 1+i) = min{|x| + |y| |x + yi = (1 + i) − 0 + δ(4 + 4i)}, when δ = 0, x + yi = 1 + i.

D(β,γ) = min{|x| + |y| |x + yi = γ − β + δα}, (1)

where δ ∈ Z[i] is an arbitrary Gaussian integer.

**CHAPTER-3**

**LITERATURE REVIEW**

**S. Sharma, K. B. Rakesh, and B. Savina:** Recent advances in technology has made researchers quite optimistic towards the feasibility of Wireless sensor networks (WSNs). These are being deployed for various applications and have huge potential for research. However, owing to the multidisciplinary nature of this field, researchers have to face many a technical hitches. In this paper, an overview of the broad research issues and challenges involved in the design of WSNs are presented. Energy conservation emerges as one the most critical aspect in hardware and software related design issues, and puts a question mark on the overall practicability of WSNs. Besides, other related main issues include specialized hardware, software and operating system, synchronization, QoS, security, architecture and data collection related aspects with minimum communication and computation costs. This paper provides an insight into various such design issues for the better understanding of this field for the overall benefit of the research community working in this area.

**Summary:** Studied about issues and challenges in the wireless sensor Networks .

**[2] Indu and D. Sunita:** Wireless sensor network (WSN) is an emerging technology that shows great promise for various futuristic applications both for mass public and military. The sensing technology combined with processing power and wireless communication makes it lucrative for being exploited in abundance in future. The inclusion of wireless communication technology also incurs various types of security threats. The intent of this paper is to investigate the security related issues and challenges in wireless sensor networks. We identify the security threats, review proposed security mechanisms for wireless sensor networks. We also discuss the holistic view of security for ensuring layered and robust security in wireless sensor networks.

**Summary:** Studied about Wireless sensor issues.

**[3] M. A. Kafi, J. B. Othman, and N. Badache:** Wireless Sensor Network (WSN) applications have become more and more attractive with the miniaturization of circuits and the large variety of sensors. The different application domains, especially critical fields of WSN use, make the reliability of data acquisition and communication a hot research field that must be tackled efficiently. Indeed, the quality of largely used, cheap-cost wireless sensors and their scarce energy supply support these reliability challenges that lead to data loss or corruption. For solving this problem, the conception of a reliability mechanism that detects these shortcomings and recovers to them becomes necessary. In this article, we present a survey on existing reliability protocols conceived especially for WSNs due to their special features. The deep classification and discussion in this study allow for understanding the pros and cons of state-of-the-art works in order to enhance the existing schemes and fill the gaps. We have classified the works according to the required level of reliability, the manner to identify the origins of the lack of reliability, and the control to recover this lack of reliability. Across the discussion along this study, we deduce that the cross-layer design between MAC, routing, and transport layers presents a good concept to efficiently overcome the different reliability holes.

**Summary:** Studied about Protocols in the wireless sensor Network.

**[4] Mehdi Afsar:** In the last decade, wireless sensor networks (WSNs) have increasingly gained the attention of researchers. Depending on the applications of WSNs, the sensor nodes are usually dispersed in harsh environments, which are prone to different types of faults. Hence, fault tolerance seems as an essential characteristic that should be considered in the architecture level of these networks. On the other hand, WSNs are battery-powered so that there is a trade-off between preserving the energy and meeting the quality of service requirements of the network. In this paper, we target these challenges through proposing a fault-tolerant scheme (FTS) for clustered sensor networks. First, all the nodes are grouped into some clusters, and then the FTS alongside the main operation of the network, that is, data acquisition, is performed by the cluster heads. The main idea of the FTS is to efficiently use different kinds of the redundancy, including the hardware, time, and space. We validate the FTS through simulation and probabilistic analysis

**Summary:** Studied about Fault tolerant fame work for Wireless sensor network.

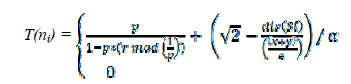
**[5] K. Gholamreza, G. Savita, and S. Sukhwinder:** Fault tolerance is one of the critical issues in Wireless Sensor Network (WSN) applications. The problem of missing sensor node, communication link and data are inevitable in wireless sensor networks. WSNs experience failure problems due to various factors such as power depletion, environmental impact, radio interference, asymmetric communication links, dislocation of sensor node and collision. Many researchers have proposed fault tolerant mechanisms that are able to achieve higher data reliability, accuracy, energy saving, enhance network lifetime and minimize failure of components of wireless sensor network. This paper presents a critical analysis of various fault tolerance mechanisms in wireless sensor networks such as redundancy based mechanisms, clustering based mechanisms and deployment based mechanisms to identify the strengths and weaknesses of each one of these mechanisms. Finally, the paper presents conclusion and suggests some future research directions that will be helpful for researchers who are working in this field.

**Summary:** Studied about Fault tolerant.

**CHAPTER-4**

**EXISTING METHOD**

A new algorithm named S-LEACH is used to extend the existence of the system. Cluster Head (CH) and Secondary Cluster (SC) are two options for creating a network. Secondary Cluster Head (SCH) during each round's sensor configuration process. According to previous studies, the shorter the gap, the better. Among CH and BS, the BS has a longer lifespan and is more energy consuming. It's the network. We chose to pick the basis equation. The closest node to the BS is designated as CH, and the closest node to the CH is designated as SCH, taking into account the energy and distance parameters of nodes. Depending on this suggestion, if the CH dead the cluster will not cut off the communication with the sink and the secondary cluster head replace the dead cluster head and pronounces itself as a cluster head. Rather than that, the cluster continuously connecting the sink as long as the active node alive in the cluster. The threshold in the proposed method is defined as follows:



The threshold is defined above because we want node energy criteria and node-sink distance to be effective in determining CH and SCH. Although Select secondary cluster head costs more time in the setup phase for the same round, it keeps the cluster communicate with the BS in case it has alive nodes. Below Equation what we call the distance coefficient,



increases the likelihood of nodes being shorter than the sink. dis(si) compute the distance of the node i to the sink. This value can be calculated by relying on the Euclidean distance (the distance between two nodes is the line length between them) and as shown in below equation

d(p,q)=

Where, points p = (p1, p2) and q = (q1, q2) are the locations of the two nodes. The number of CH should equal the number of SCH (one SCH for each CH) at the beginning of the round. Returning to reducing data transmission time, we suggest that the position of BS should be in the center of the WSN as shown in Figure surrounding with four equal-area squares in order to optimize energy consumption for communicating between clusters and BS.



Figure : Architecture platform of the proposed algorithm

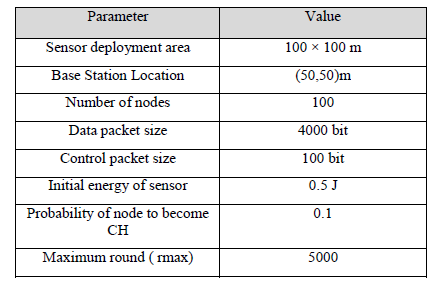
From the perspective of making the network with high latency. we applythe algorithm of node density in the cluster. Some nodes are not able to join the cluster because they have not the range ratio of any cluster, So they connect to BS directly without electing CH by the protocol which called Direct transmission (DTx).

Considering the major issues of the Wireless Sensor Networks, We focus on the most common algorithm. LEACH protocol, which enhanced the performance of the WSN such as Energy efficient, Transmission time, transmission rate, optimal position of the sink, the optimal position of the cluster head, node density and network lifetime. These are the main performance enhancement factors that we figure out in this paper. Figure 4 shows the initial LEACH protocol with homogenous network topology using MATLAB (Version:9.2.0,518641) with an area of 100 × 100m. BS is in the center of the network (50,50), with 100 nodes are deployed randomly and certain nodes are randomly selected as cluster heads. We fixed a maximum number of clusters to be 10, a maximum number of rounds is 5000 and the initial energy equal 0.5Joules. Then, we set the data packet size equal to 4000 bits and the number of packets from each node to be 10 each round. We prefer to save the same parameters for all algorithms to gain fair comparison



Figure : Nodes density in a cluster; LEACH protocol connection and

DTx protocol connection with BS



Prolong network lifetime still a major issue in wireless sensor networks .So, we tend to extend the network lifetime and implement the proposed algorithm with the same parameter used for the basic LEACH protocol. proposed S-LEACH protocol simulation parameters are shown in above Table. We used MATLAB (Version: 9.2.0,518641) simulation tool with a simulation time of 2002.291161 seconds. Network lifetime measurement factor is used to monitor the network life cycle, in this paper we focus on the last dead node in the whole network concurrently with data packet transmission through the network

100 nodes were deployed randomly in the 100\*100\*m^2 WSN sensing area size

Calculation of

Election of CH (CLUSTER HEAD)

Steady State Phase

Setup Phase of Each Round

If the Energy of CH is Zero then choose the Secondary cluster Head (SCH) Closest node to CH

Data Transmission to BS

Determination of Alive Nodes

Determination of Energy Consumption for Nodes

**Block diagram of Existing method**

**DISADVANTAGES:**

* If any CH will be dead then, the packets from that particular cluster will never reaches the BS resulting in loss of data of that cluster.
* There exists an unbalanced energy consumption when the rounds goes on.
* There is no optimal limit for the number of nodes in a cluster.

**CHAPTER-5**

**PROPOSED METHOD**

1. **The Proposed Hierarchical Network Model**
2. **WSN Model**: In this paper, we consider a WSN model that consists of a base station node (BS) or a sink node and n identical sensor nodes that are randomly allocated over the rectangular area S = X ×Y . Here, based on the geographical position of the nodes, the network area will be divided into some virtual square grids. Accordingly, each grid will choose an active node as the CH node to aggregate and transmit data as in Fig. 2 by GAF algorithm (geographic adaptive fidelity) [13]. The GAF enhanced versions as well as GAF benefits are also introduced in [40]. The GAF algorithm divides the network into virtual square grids with unit r = R/√ 5, where R is the sensor communication range. The nodes in each grid will receive one of three statuses, namely: discovery, sleep or active at a time. Initially, all nodes start with the discovery state and set a predefined time Td (time period for discovery state). When a node finish Td time, it broadcasts a message to all nodes in grid. A node will be switch to active state if the energy level is greater than a threshold and it does not receive any other discovery message and set a time of operation Ta, else it will be switch to sleep state and set a time of sleep Ts, details as in [13]. Each virtual grid has only one active node at a time. The active node is selected as a CH of cluster (grid), which will collect the packets in intra-clusters and transmits packets to BS through single-hop or multi-hops. In our approach, after each cluster has selected a CH node, the Gaussian network (Gα) of CH nodes will be built through connections between CH nodes [12]. Accordingly, each grid will only have a CH node and it is represented by a Gaussian integer (β = x + yi). The Gα of CH nodes (CH-Gaussian network) as shown in Fig. 1, which will be discussed in detail in the next section. 2) The CH-Gaussian Network: Our previous work has proposed a hierarchical topology based on Gaussian network connection properties to efficiency of WSN topology [12]. In that work, we presented three connection models corresponding to the four adjacent links of each Gaussian node in the wireless sensor network. Accordingly, based on the WSN model in this work, we have a Gaussian network Gα of CH nodes generated by a Gaussian integer α = a + bi, with 0 ≤ a ≤ b, and the largest common divisor gcd(a, b) = d > 1. Thus, the total number of Gaussian nodes in the Gα is N as in formula (2). After the WSN divided into a set of virtual square grids (clusters) Γ(G) = N, the CH node in each cluster is connected together to form a Gaussian network of CH nodes in the network area S = X ×Y as in formula (3). Hence, each CH node in each cluster is equivalent to a node in the Gaussian network as in Fig. 3.

N = {d × f |f = (a 2 + b 2 )/d} (2)

S = {X × Y |X = d × r, Y = f × r} (3)

According to the set of clusters, each cluster will be assigned with an ID. since, corresponding to each cluster ID (Gr ⊂ Γ(G) ) will only have one CH node and only one Gaussian integer representation β = x + yi, which the relationship between Gr and β is shown in formula (4). Therefore, in our proposal, we based on the cluster ID to calculate the corresponding Gaussian integer.

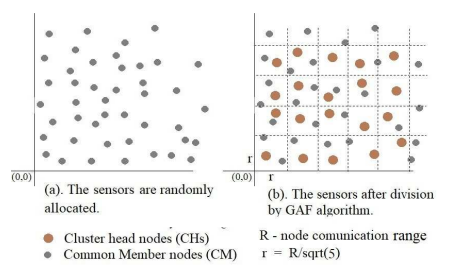


Figure: The network are is divided into several square virtual grids

1. Shortest Path Routing Protocol In our network model, the network is divided into clusters, which will optimize the energy efficiency of the wireless sensor network. However, to increase routing efficiency more, in [12] based on the shortest distance in the Gaussian network, we proposed the shortest path routing protocol. Accordingly, the length of the path from CH(γ = xs + ysi) to BS(β = xd + ydi) is the shortest distance between γ and β, D(β, γ) = |Xmin| + |Ymin| as in formula (1). Therefore, the data transmission path P(β,γ) from CH node to BS node consists passing through Xmin steps in the real axis (x-axis) and Ymin steps in the imaginary axis (y-axis). A simple presentation of a routing path as shown in formula (5), in which the packets pass through nodes on the x-axis, then the nodes on the y-axis. In addition, the CH nodes are represented by a Gaussian integer, so during routing, nodes will rely on the Gaussian integer to forward the packet without performing any additional work such as flooding to find the next node, which results in reducing costs for routing.

Since, in this paper, through improving the shortest path routing, we have proposed a fault-tolerant approach for wireless sensor networks. The details of the fault tolerance approach will be presented in the next section.

P(β,γ) = { |X Xmin| m=0 ((xs±m)+ysi) ∩ |Y Xmin| n=0 (xs+(ys±n)i)} (5)

**Fault Tolerance Mechanism**

According to the hierarchical topology based on a Gaussian network of WSN, this paper propose a hybrid fault-tolerant routing protocol based on a combination of clustering and multi-path routing. Our proposal is called fault tolerance clustering based on a Gaussian network for wireless sensor network protocol (FCGW). In FCGW, the fault tolerance mechanism focuses on fault detection and fault recovery for CH nodes. Therefore, our fault tolerance mechanism includes two phases: fault detection phase and fault recovery phase. Accordingly, based on the symmetric links of the Gaussian network and the shortest path routing as in formula (5), the fault recovery process will be optimized as multiple path routing.

1. **Fault Detection Phase**

In wireless sensor networks, there are many causes for faults such as software failure, hardware failure, natural factors, harsh device deployment environment, resulting in network connection failures. In this paper, we propose a fault detection mechanism for CH nodes based on clustering and self diagnostics by following the information exchanged between neighbors CH. Accordingly, after selecting CHs, the CHs periodically broadcast the grid-ID (grid index) and CH-ID (the CH index) for the CHs in four adjacent clusters. Therefore, each CH node will update a CH neighbor table (NeigT ableCH), which includes adjacent grid-ID, CH-ID, and CH states of adjacent clusters (Fig. 4). In a given period of information exchange time TheathCH (TheathCH < Ta), if a CH does not receive information from the adjacent CHs then it will update the state of the corresponding CH in the CH neighbor table as “false”. According to the CH status in the CH neighbor table, our proposal will diagnose CH node fault or not. B. Fault Recovery Phase In our network model, the faults may occur at the CH nodes or CM nodes, which would cause the link between the CH nodes (extra-connections) or between CH and CMs (intraconnections) to be lost.

• In case of the CMs fault, CM nodes are automatically removed from the cluster, and

• In case of the CHs fault, the fault recovery mechanism is executed until another CH is re-selected in the cluster. In fault recovery mechanism, one CH before forwarding the packet, it will get the next CH state from NeigT ableCH.

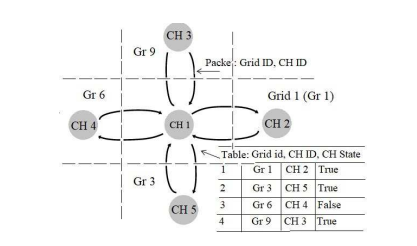


Figure:Definition of the CH neighbor table (NeigT ableCH) a

If this next CH fails, the packets will be automatically forwarded to other next CH nodes.

Accordingly, the routing paths of packets from CH(β = xs + ysi) node to BS(η = xd + ydi) node include D(η, β) = |Xmin|+|Ymin| hops (1). In this case, the packet can be transmitted in real dimension (real axis) or imaginary dimension (imaginary axis), this means that at node β = xs + ysi, the packets can be transmitted to next nodes β1 = (xs+1)+ysi or β2 = xs+(ys+1)i . Therefore, in our fault recovery mechanism, the number of redundant routing paths to send data from the source node to the destination node is Npath as formula (6), where Npath is a 2-combination of a set D(η, β). Therefore, during the routing process, if the next CH fault on the real axis (β1 is faulty), it will be replaced by the next CH node in the imaginary axis (β2 ), similarly, if the CH fault on the imaginary axis (β2 is faulty) will be replaced by β1 as shown in formula (7). , we introduced in detail the fault detection and fault recovery mechanism in new WSN network model, it is simple and easy to implement.

Npath = D(η, β)! /2!(D(η, β) − 2)! (6)

P(β,η) →formula(5),if there is no fault,

β1(xs±1,ys) ,if fault in imaginary axis,

β2(xs,ys±1), if fault in real axis

**CHAPTER-6**

**ADVANTAGES AND APPLICATIONS**

**Advantages:**

1. All the CHs are connected through Gaussian Network making chance of faults occurring less.

2. CHs nodes are connected through GN, making the chances of fault tolerance high.

3. Fault tolerance is higher compared to remaining algorithms.

**Applications:**

1. Industrial control

2. Environmental monitoring,

3. Military surveillance,

4. Intelligent transportation systems and medical field.

5. Furthermore, it can function independently in harsh or high-risk places where human presence is not possible

6. Disaster relief operations.

7. Biodiversity mapping

8. Monitoring of temperature, pressure, and humidity.

**CHAPTER-7**

**MATLAB**

**7.1 INTRODUCTION TO MATLAB**

**What Is MATLAB?**

MATLAB is an elite dialect for specialized registering. It incorporates calculation, representation, and programming in an easy to-utilize condition wherein issues and preparations are communicated in herbal numerical documentation. Run of the mill utilizes comprise

• Math and calculation

• Algorithm advancement

• Data obtaining

• Modeling, re-enactment, and prototyping

• Data examination, investigation, and representation

• Scientific and designing illustrations

• Application advancement, including graphical UI building

MATLAB is an intuitive framework whose important statistics aspect is an show off that does not require dimensioning. This allows you to tackle several specialized processing issues, particularly those with framework and vector info, in a small quantity of the time it'd take to compose a program in a scalar non intuitive dialect, as an instance, C or FORTRAN.

The call MATLAB stays for grid studies facility. MATLAB changed into first of all composed to present easy access to framework programming created by way of the LINPACK and EISPACK ventures. Today, MATLAB motors fuse the LAPACK and BLAS libraries, inserting the cutting side in programming for network calculation.

MATLAB has advanced over a time of years with contribution from several customers. In university situations, it's far the usual academic apparatus for early on and propelled guides in mathematics, designing, and science. In enterprise, MATLAB is the tool of choice for excessive-profitability studies, advancement, and exam.

MATLAB highlights a collection of more utility-specific arrangements known as tool booths. Important to most clients of MATLAB, device kits permit you to learnandapply particular innovation. Tool compartments are exhaustive accumulations of MATLAB capacities (M-records) that reach out the MATLAB condition to take care of precise training of problems. Territories in which tool stash are reachable include flag coping with, manipulate frameworks, neural structures, fluffy reason, wavelets, pastime, and severa others.

**The MATLAB System:**

The MATLAB system consists of five main parts.

**Development Environment:**

 This is the set of tools and centres that help you operate MATLAB features and files. Many of that gear are graphical person interfaces. It includes the MATLAB desktop and Command Window, a command history, an editor and debugger, and browsers for viewing assist, the workspace, files, and the hunt direction.

**The MATLAB Mathematical Function:**

This is a great collection of computational algorithms ranging from standard capabilities like sum, sine, cosine, and complex arithmetic, to extra sophisticated features like matrix inverse, matrix eigen values, Bessel functions, and speedy Fourier transforms.

**The MATLAB Language:**

This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create complete large and complex application programs.

**Graphics:**

MATLAB has considerable centres for displaying vectors and matrices as graphs, as well as annotating and printing those graphs. It consists of high-stage functions for 2-dimensional and 3-dimensional records visualization, photograph processing, animation, and presentation graphics. It also consists of low-stage capabilities that will let you absolutely customise the appearance of graphics as well as to construct complete graphical person interfaces for your MATLAB programs.

**The MATLAB Application Program Interface (API):**

This is a library that allows you to put in writing C and Fortran applications that have interaction with MATLAB. It consists of facilities for calling workouts from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for studying and writing MAT-documents.

**7.2 MATLAB WORKING ENVIRONMENT:**

## MATLAB DESKTOP:

Matlab Desktop is the principle Matlab application window. The desktop consists of five sub windows, the summon window, the workspace program, the existing catalog window, the order records window, and at the least one figure home windows, which can be proven simply while the consumer suggests a sensible.

The order window is the area the customer sorts MATLAB orders and expressions at the initiate (>>) and wherein the yield of these fees is shown. MATLAB characterizes the workspace because the association of factors that the customer makes in a work session. The workspace software demonstrates these elements and some statistics approximately them. Double tapping on a variable within the workspace application dispatches the Array Editor, which may be applied to get data and salary instances modify sure homes of the variable.

The present Directory tab over the workspace tab demonstrates the substance of the existing registry, whose way is seemed within the present index window. 1For case, within the windows running framework the manner may be as consistent with the subsequent: C:MATLABWork, demonstrating that registry "paintings" is a subdirectory of the primary catalog "MATLAB", which is delivered in pressure C. Tapping on the bolt inside the present index window demonstrates a rundown of as of past due utilized approaches. Tapping at the seize to one aspect of the window enables the client to exchange the existing catalog.

MATLAB utilizes an inquiry way to discover M-data and different MATLAB related documents, which might be sort out in catalogs within the PC file framework. Any file keep strolling in MATLAB must dwell inside the ebb and go with the flow registry or in an index that is on are trying to find manner. Of direction, the statistics supplied with MATLAB and math works device kits are included into the inquiry way. The least stressful method to look which indexes are at the inquiry manner. The handiest method to peer which catalogs are soon the quest way, or to encompass or regulate an inquiry manner, is to pick set manner from the File menu the computer, and after that utilization the set way exchange container. It is exquisite exercise to add any typically utilized catalogs to the pursuit way to hold a strategic distance from again and again having the exchange the existing index.

The Command History Window contains a record of the orders a client has entered in the charge window, including both present and past MATLAB sessions. Already entered MATLAB orders can be chosen and re-executed from the charge history window by right

tapping on a summon or arrangement of orders. This activity dispatches a menu from which to choose different choices notwithstanding executing the orders. This is helpful to choose different choices notwithstanding executing the summons. This is a valuable component while trying different things with different orders in a work session

**Using the MATLAB Editor to create M-Files:**

The MATLAB manager is both a word processor unique for making M-statistics and a graphical MATLAB debugger. The proofreader can display up in a window without everybody else, or it could be a sub window in the laptop. M-facts are intended by means of the expansion .M, as in pixelup.M. The MATLAB editorial manager window has various draw down menus for errands, for instance, sparing, seeing, and troubleshooting documents. Since it plays out a few basic checks and furthermore utilizes shading to separate between exclusive additives of code, this content device is suggested as the equipment of selection for composing and changing M-capacities. To open the proofreader, sort regulate at the incite opens the M-report filename.M in a supervisor window, organized for altering. As referred to before, the record has to be inside the momentum catalog, or in an index within the pursuit manner.

**Getting Help:**

The important technique to get help on line is to utilize the MATLAB assist application, opened as a exclusive window both via tapping at the query mark image at the computing device toolbar, or by using writing help program on the provoke within the order window. The help Browser is an internet application coordinated into the MATLAB computing device that shows a Hypertext Markup Language (HTML) statistics. The Help Browser contains of two sheets, the assistance pilot sheet, used to find out data, and the show sheet, used to look the statistics. Clear as crystal tabs aside from pilot sheet are applied to play out a pursuit. Second, within the motion pictures taken via transferring camera setup, the state of affairs becomes extra complex because the heritage may additionally exchange by using shifting shot, we cannot tune item motion exactly inside the sum of distinction map. Therefore, in this situation, the purpose is executed through reusing the previous seam and applying it to the cutting-edge body. In order to discover the seams, we use the preceding seam from previous body to look the modern-day seam in contemporary frame. our method is using a seam computed in frame1 (in crimson) to go looking a comparable seam in frame2. For the pixels close by the area of previous seam, we decide how a lot the selected pixel might vary from the pixel of preceding seam. We use difference of the 2 pixels as the degree of temporal coherence. If the distinction value of first seam pixel is over the threshold, we can keep to go looking the next seam pixel on three feasible pixels (in yellow, blue and brown) in subsequent row, until we discover 5 consecutive pixels that also exceed the threshold.

When we can't search the matching seam, we recalculate the energy for a new seam. We assume a seam 𝑆l-1 has been calculated inside the previous body, and a seam must be calculated for the contemporary frame. For preserving the temporal coherence, we want to make a new seam close to the previous seam with the identical index. We use the distinction among preceding seam and all pixels at the current body as the measure

Thus we upload temporal coherence price Tc(i,j) to the strength map earlier than calculating a seam 𝑆L. The price Tc is zero while the body pixels have the equal fee as previous seam pixels. Using our temporal coherence price, we will calculate the seam which has least electricity and is more close to the preceding seam in previous frame. Consequently, we will decrease the jittery artifacts inside the films.

**COMMUNICATION:**

Communications System Toolbox™ offers algorithms and gear for the layout, simulation, and analysis of communications systems. These capabilities are furnished as MATLAB ® features, MATLAB System gadgets™, and Simulink ® blocks. The machine toolbox includes algorithms for source coding, channel coding, interleaving, modulation, equalization, synchronization, and channel modeling. Tools are supplied for bit blunders charge evaluation, producing eye and constellation diagrams, and visualizing channel characteristics. The machine toolbox additionally provides adaptive algorithms that allow you to version dynamic communications structures that use OFDM, OFDMA, and MIMO techniques. Algorithms support fixed-point facts arithmetic and C or HDL code era.

**Key Features**

▪ Algorithms for designing the physical layer of communications systems, which includes supply coding, channel coding, interleaving, modulation, channel fashions, MIMO, equalization, and synchronization

▪ GPU-enabled System objects for computationally intensive algorithms together with Turbo, LDPC, and Viterbi decoders

▪ Interactive visualization equipment, consisting of eye diagrams, constellations, and channel scattering capabilities

▪ Graphical tool for evaluating the simulated bit mistakes rate of a machine with analytical outcomes

▪ Channel models, consisting of AWGN, Multipath Rayleigh Fading, Rician Fading, MIMO Multipath Fading, and

LTE MIMO Multipath Fading

▪ Basic RF impairments, along with nonlinearity, section noise, thermal noise, and section and frequency offsets

▪ Algorithms available as MATLAB features, MATLAB System objects, and Simulink blocks

▪ Support for fixed-point modeling and C and HDL code technology

**System Design, Characterization, and Visualization:**

The layout and simulation of a communications gadget requires analyzing its reaction to the noise and interference inherent in real-world environments, reading its behavior the usage of graphical and quantitative manner, and determining whether the resulting overall performance meets requirements of acceptability. Communications System Toolbox implements a selection of obligations for communications machine layout and simulation. Many of the functions, System objects™, and blocks inside the device toolbox perform computations associated with a specific thing of a communications gadget, consisting of a demodulator or equalizer. Other talents are designed for visualization or evaluation.

**System Characterization**

The system toolbox offers several standard methods for quantitatively characterizing system performance:

▪ Bit error rate (BER) computations

▪ Adjacent channel power ratio (ACPR) measurements

▪ Error vector magnitude (EVM) measurements

▪ Modulation error ratio (MER) measurements

Because BER computations are fundamental to the characterization of any communications system, the system toolbox provides the following tools and capabilities for configuring BER test scenarios and accelerating BER simulations:

**BER tool**— A graphical user interface that enables you to analyze BER performance of communications systems. You can analyze performance via a simulation-based, semi analytic, or theoretical approach.

**Error Rate Test Console** — A MATLAB object that runs simulations for communications systems to measure error rate performance. It supports user-specified test points and generation of parametric performance plots and surfaces. Accelerated performance can be realized when running on a multi core computing platform.

**Multi core and GPU acceleration** — A capability provided by Parallel Computing Toolbox™ that enables you to accelerate simulation performance using multi core and GPU hardware within your computer.

**Distributed computing and cloud computing support** — Capabilities provided by Parallel Computing Toolbox and MATLAB Distributed Computing Server™ that enable you to leverage the computing power of your server farms and the Amazon EC2 Web service. Performance Visualization. The system toolbox provides the following capabilities for visualizing system performance:

**Channel visualization tool** — For visualizing the characteristics of a fading channel

**Eye diagrams and signal constellation scatter plots** — for a qualitative, visual understanding of system behavior that enables you to make initial design decisions

**Signal trajectory plots** — for a continuous picture of the signal’s trajectory between decision points

**BER plots** — for visualizing quantitative BER performance of a design candidate, parameterized by metrics such as SNR and fixed-point word size

**Analog and Digital Modulation**

Analog and digital modulation strategies encode the facts circulation into a sign this is appropriate for transmission. Communications System Toolbox presents some of modulation and corresponding demodulation abilities. These talents are available as MATLAB features and gadgets, MATLAB System Modulation sorts provided by the toolbox are:

**Source and Channel Coding**

Communications System Toolbox affords source and channel coding talents that can help you develop and compare communications architectures fast, enabling you to discover what-if eventualities and avoid the need to create coding competencies from scratch.

**Source Coding**

Source coding, also referred to as quantization or signal formatting, is a manner of processing facts a good way to lessen redundancy or prepare it for later processing. The system toolbox offers a diffusion of styles of algorithms for imposing source coding and interpreting, inclusive of:

▪ Quantizing

▪ Companding (*µ*-law and A-law)

▪ Differential pulse code modulation (DPCM)

▪ Huffman coding

▪ Arithmetic coding

**Channel Coding**

▪ orthogonal area-time block code (OSTBC) (encoder and decoder for MIMO channels)

▪ Turbo encoder and decoder examples

The gadget toolbox offers application functions for developing your personal channel coding. You can create generator polynomials and coefficients and syndrome deciphering tables, in addition to product parity-take a look at and generator matrices.

The system toolbox additionally presents block and convolutional interleaving and deinters leaving functions to reduce facts errors as a result of burst mistakes in a conversation machine:

**Block,** including General block interleaver, algebraic interleaver, helical scan interleaver, matrix interleaver, and random interleaver.

**Convolutional,** including General multiplexed interleaver, convolutional interleaver, and helical interleaver

**Channel Modeling and RF Impairments**

Channel Modeling

Communications System Toolbox provides algorithms and tools for modeling noise, fading, interference, and different distortions which might be commonly found in communications channels. The system toolbox supports the subsequent styles of channels:

▪ Additive white Gaussian noise (AWGN)

▪ Multiple-enter multiple-output (MIMO) fading

▪ Single-enter single-output (SISO), Rayleigh, and Rician fading

▪ Binary symmetric

A MATLAB channel object provides a concise, configurable implementation of channel models, enabling you to

specify parameters such as:

▪ Path delays

▪ Average path gains

▪ Maximum Doppler shifts

▪ K-Factor for Rician fading channels

▪ Doppler spectrum parameters

For MIMO systems, the MATLAB MIMO channel object expands these parameters to also include:

▪ Number of transmit antennas (up to 8)

▪ Number of receive antennas (up to 8)

▪ Transmit correlation matrix

▪ Receive correlation matrix

To combat the effects noise and channel corruption, the system toolbox provides block and convolutional coding and decoding techniques to implement error detection and correction. For simple error detection with no inherent correction, a cyclic redundancy check capability is also available. Channel coding capabilities provided by the system toolbox include:

▪ BCH encoder and decoder

▪ Reed-Solomon encoder and decoder

▪ LDPC encoder and decoder

▪ Convolutional encoder and Viterbi decoder

****

**RF Impairments**

To model the effects of a non-ideal RF front end, you can introduce the following impairments into your communications system, enabling you to explore and characterize performance with real-world effects:

▪ Memory less nonlinearity

▪ Phase and frequency offset

▪ Phase noise

▪ Thermal noise

You can include more complex RF impairments and RF circuit models in your design using SimRF™.

****

**Equalization and Synchronization**

Communications System Toolbox lets you discover equalization and synchronization strategies. These techniques are usually adaptive in nature and tough to design and symbolize. The machine toolbox affords algorithms and tools that will let you swiftly select the proper approach on your communications machine. Equalization To compare one-of-a-kind techniques to equalization, the device toolbox offers you with adaptive algorithms which include:

▪ LMS

▪ Normalized LMS

▪ Variable step LMS

▪ Signed LMS

▪ MLSE (Viterbi)

▪ RLS

▪ CMA

These adaptive equalizers are available as nonlinear decision feedback equalizer (DFE) implementations and as

Linear (symbol or fractionally spaced) equalizer implementations.

**Synchronization**

The device toolbox provides algorithms for each service segment synchronization and timing phase synchronization. For timing section synchronization, the machine toolbox presents a MATLAB Timing Phase Synchronizer object that offers the following implementation techniques:

▪ Early-late gate timing method

▪ Gardner’s method

▪ Fourth-order nonlinearity method

**Stream Processing in MATLAB and Simulink**

Most verbal exchange structures cope with streaming and frame-primarily based statistics using a aggregate of temporal processing and simultaneous multi frequency and multichannel processing. This form of streaming multidimensional processing can be visible in superior communication architectures consisting of OFDM and MIMO. Communications System Toolbox enables the simulation of advanced communications structures via helping move processing and frame-based simulation in MATLAB and Simulink. In MATLAB, circulate processing is enabled by way of System items™, which use MATLAB objects to symbolize time-based and facts-driven algorithms, sources, and sinks. System objects implicitly manipulate many information of flow processing, including information indexing, buffering, and management of set of rules state. You can mix System gadgets with fashionable MATLAB functions and operators. Most System items have a corresponding Simulink block with the identical abilities. Simulink handles circulation processing implicitly with the aid of coping with the float of information thru the blocks that make up a Simulink model. Simulink is an interactive graphical environment for modeling and simulating dynamic systems that uses hierarchical diagrams to symbolize a machine version. It includes a library of widespread-reason, predefined blocks to represent algorithms, resources, sinks, and device hierarchy.

**Implementing a Communications System**

Fixed-Point Modeling Many communications systems use hardware that requires a fixed-point representation of your design.

Communications System Toolbox supports fixed-point modeling in all relevant blocks and System objects™ with tools that help you configure fixed-point attributes.

Fixed-point support in the system toolbox includes:

▪ Word sizes from 1 to 128 bits

▪ Arbitrary binary-point placement

▪ Overflow handling methods (wrap or saturation)

▪ Rounding methods: ceiling, convergent, floor, nearest, round, simplest, and zero

Fixed-Point Tool in Simulink Fixed Point™ facilitates the conversion of floating-point data types to fixed point. For configuration of fixed-point properties, the tool tracks overflows and maxima and minima.

**Code Generation**

Once you've got advanced your set of rules or communications device, you can robotically generate C code from it for verification, rapid prototyping, and implementation. Most System gadgets, functions, and blocks in Communications System Toolbox can generate ANSI/ISO C code the use of MATLAB Coder™, Simulink Coder™, or Embedded Coder™. A subset of System gadgets and Simulink blocks also can generate HDL code. To leverage present highbrow belongings, you can choose optimizations for specific processor architectures and integrate legacy C code with the generated code.

You can also generate C code for both floating-point and fixed-point data types.

DSP Proto typing DSPs are used in communication system implementation for verification, rapid prototyping, or final hardware implementation. Using the processor-in-the-loop (PIL) simulation capability found in Embedded Coder, you can verify generated source code and compiled code by running your algorithm’s implementation code on a target processor. FPGA Prototyping

FPGAs are used in communication systems for implementing high-speed signal processing algorithms. Using the FPGA-in-the-loop (FIL) capability found in HDL Verifier™, you can test RTL code in real hardware for any existing HDL code, either manually written or automatically generated HDL code.

**CHAPTER -8**

**HARDWARE & SOFTWARE REQUIREMENTS:**

**Software:**

• Matlab R2018a.

**Hardware:**

**Operating Systems:**

• Windows 10

• Windows 7 Service Pack 1

• Windows Server 2019

• Windows Server 2016

**Processors:**

Minimum: Any Intel or AMD x86-64 processor

Recommended: Any Intel or AMD x86-64 processor with four logical cores and AVX2 instruction set support

**Disk:**

Minimum: 2.9 GB of HDD space for MATLAB only, 5-8 GB for a typical installation

Recommended: An SSD is recommended a full installation of all Math Works products may take up to 29 GB of disk space

**RAM:**

Minimum: 4 GB

Recommended: 8

**CHAPTER-9**

**RESULTS**

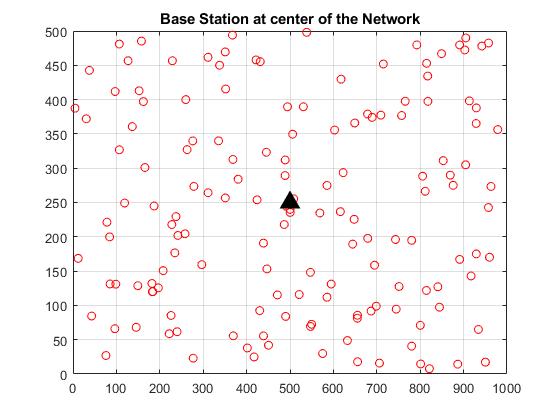


Fig. Base Station Locating At center of the Network.

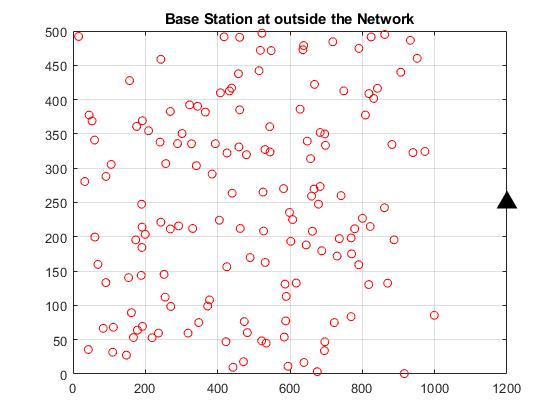


Fig. Base station locating outside of the network

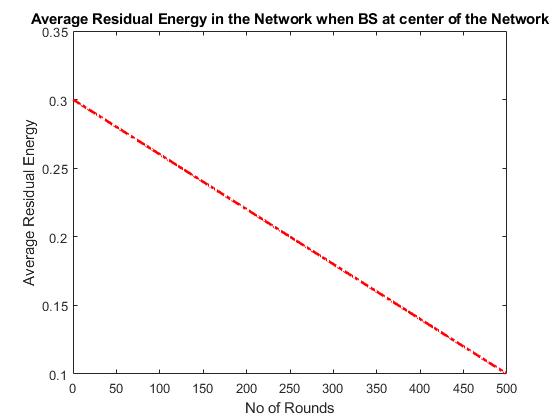


Fig. Average Residual energy when base station locating at the center

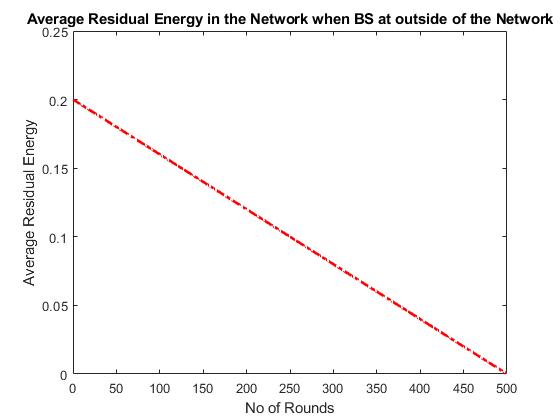


Fig. Average Residual energy when base station locating outside of the network

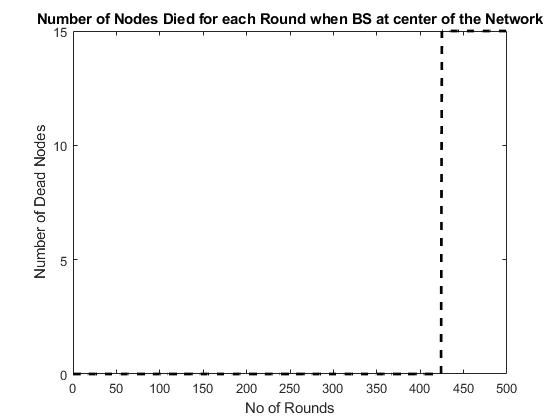


Fig. No of Nodes Died for each Round when base station locating center of the network

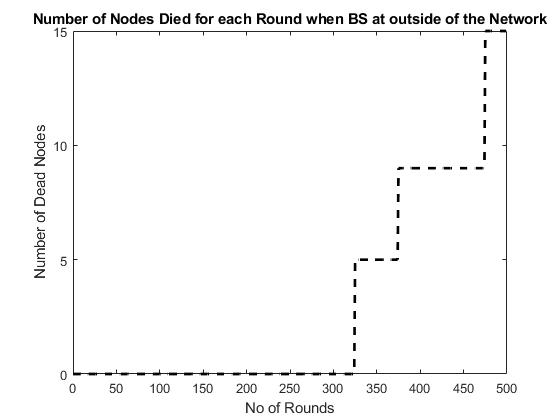


Fig. No of Nodes Died for each Round when base station locating outside of the network

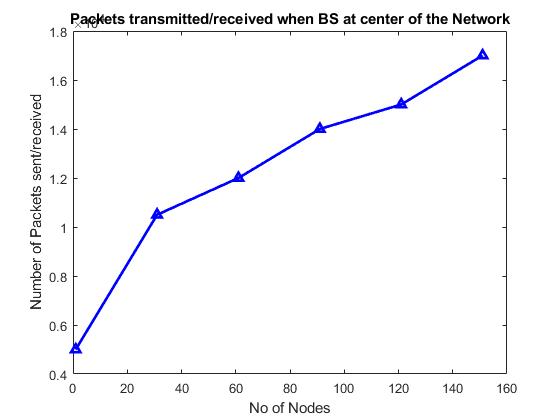


Fig. Packets Transmitted or Received when base station locating center of the network

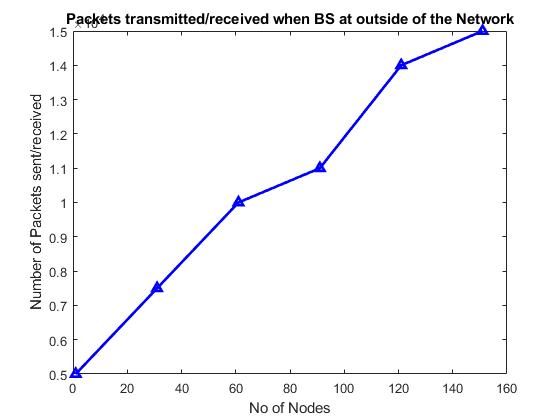


Fig. Packets Transmitted or Received when base station locating Outside of the network

**CHAPTER-10**

**CONCLUSION**

In this paper, we have constructed a hierarchical topology for wireless sensor network using the Gaussian network connection properties and clustering routing. According to our approach, the sensors are randomly distributed in a rectangular area and clustered into several square grids. The CH nodes are connected together to form a Gaussian network, so this approach has improved fault tolerance through symmetric links and multi-path routing. In addition, the CH nodes are represented as Gaussian integers, which is used in the routing protocol to reduce the complexity of routing algorithms.

**CHAPTER-11**

**REFERENCES**

[1] S. Sharma, K. B. Rakesh, and B. Savina, “Issues and challenges in wireless sensor networks,” in Proc. IEEE ICMIRA, 2013.

[2] Indu and D. Sunita, “Wireless sensor networks: Issues and challenges,” Int. J. Comput. Sci. Mobile Comput., vol. 3, no. 6, pp. 681–685, June 2014.

[3] M. A. Kafi, J. B. Othman, and N. Badache, “A survey on reliability protocols in wireless sensor networks,” ACM Comput. Surveys, vol. 50, no. 2, pp. 31:1–31:47, June 2017. [4] Mehdi Afsar, “A comprehensive fault tolerant framework for wireless sensor networks,” Security Commun. Netw., vol. 8, no. 17, pp. 3247–3252, Mar. 2015.

[5] K. Gholamreza, G. Savita, and S. Sukhwinder, “A survey on fault tolerance techniques in wireless sensor networks,” in Proc. IEEE ICGCIoT, 2015.

[6] N. Verma, and D. Singh, “Data redundancy implications in wireless sensor networks,” Procedia Comput. Sci., vol. 132, pp. 1210–1217, 2018.

[7] M. Sushruta, J. Lambodar, and P. Aarti “Fault tolerance in wireless sensor networks,” Int. J. Adv. Research Comput. Sci. Softw. Eng., vol. 2, no. 10, pp 146–153, Oct. 2012.

[8] S. Hu and G. Li, “Fault-tolerant clustering topology evolution mechanism of wireless sensor networks,” IEEE Access, vol. 6, pp. 28085–28096, 2018.

[9] F. Fanian and M. K. Rafsanjani, “Cluster-based routing protocols in wireless sensor networks: A survey based on methodology,” J. Netw. Comput. Appl., vol. 142, pp. 111–142, Sept. 2019.

[10] Y. Mohamed, I. F. Senturk, A. A. Kemal, L. Sookyoung, and F. Fatih,“Topology management techniques for tolerating node failures in wireless sensor networks: A survey,” Comput. Netw., vol. 58, pp. 255– 283, Jan. 2014.

[11] Z. Jiao et al., “Fault-tolerant virtual backbone in heterogeneous wireless sensor network,” IEEE/ACM Trans. Netw., vol. 25, no. 6, pp. 3487–3499, Dec. 2017. [12] D. N. Quoc et al., “Energy efficiency clustering based on Gaussian network for wireless sensor network,” IET Commun., vol. 13, no. 6, pp. 741–747, 2019.

[13] J. Heidemann, D. Estrin, and Y. Xu, “Geography-informed energy conservation for Ad Hoc routing,” in Proc. ACM MobiCom, 2011.

[14] A. Munir, J. Antoon, and A. Gordon-Ros, “Modeling and analysis of fault detection and fault tolerance in wireless sensor networks,” ACM Trans. Embedded Comput. Syst., vol. 14, no. 1, pp. 3:1–3:43, Jan. 2015.

[15] M. Arunanshu and M. K. Pabitra, “Fault diagnosis in wireless sensor networks: A survey,” IEEE Commun. Surveys Tuts., vol. 15, no. 4, pp. 2000–2016, Mar. 2013.

[16] M. Shyama and Anju S. Pillai, “Fault-tolerant techniques for wireless sensor network: A comprehensive survey,” Innova. Elec. Commun. Eng., vol. 51, pp. 261–269, Feb. 2019.

[17] Z. Zhang et al., “A survey on fault diagnosis in wireless sensor networks,” IEEE Access, vol. 6, pp. 11349–11364, Feb. 2018.

[18] O. O. Olayinka and S. A. Attahiru, “A survey on an energy-efficient and energy-balanced routing protocol for wireless sensor networks,” Sensors, vol. 17, no.5, pp. 1084–1135, May 2017.

[19] K. S. Sunil, K. Prabhat, and P. S. Jyoti, “A survey on successors of LEACH protocol,” IEEE Access, vol. 5, pp. 4298–4328, Feb. 2017. [20] M. N. Cheraghlou and M. Haghparast, “A novel fault-tolerant leach clustering protocol for wireless sensor networks,” J. Circuits, Syst. Comput., vol. 23, no. 3, pp. 145–162, Mar. 2014.