Research and Postgraduate Studies (RPGS) Unit

Thesis Title: "Design of novel high density, fault tolerant Protocol for cluster based routing in Ad-hoc Networks"

Student's Name: MINANI Frodouard

College of Science and Technology

School of Information and Communication Technology (SoICT)

Master of Science in Information and Communication Technology, Option of Operational Communication (MSc ICT-OC)

2017

Thesis Title: "Design of novel high density, fault tolerant Protocol for cluster based routing in Ad-hoc Networks"

By

Name: MINANI Frodouard

Registration Number: 11111443

A dissertation submitted in partial fulfilment of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION AND COMMUNICATION TECHNOLOGY

In the College of Science and Technology

Supervisor: Prof.Santhi Kumaran

June 2017

DECLARATION

I hereby declare that all information in this document is original and it has never been presented in any University or other Institutions of Higher Learning. I also declare that, as required by rules I have fully cited and referenced all material and results that are not original to this work.

Student Name and Number

MINANI Frodouard

11111443

Signed: ………………………

Date : ..../......../.........

CERTIFICATE

*This is to certify that the project work entitled "*Design of novel high density, fault tolerant Protocol for cluster based routing in Ad-hoc Networks*" is a record of original work done by MINANI Frodouard with Reg no: 11111443 in partial fulfillment of the requirements for the award of Masters of Science in Information and Communication Technology of College of Science and Technology, University of Rwanda during the academic year 2015-2016.*

Prof. Santhi Kumaran
…………………………......

Supervisor

Mr. Henry Kirya
......………………

Head Dept of IT

ACKNOWLEDGEMENTS

First of all, I would like to thank the Almighty God who gave me the grace to accomplish this work. I would like to express my sincere and special thanks to my supervisor Prof.Santhi Kumaran for her tireless guidance and supervision in the achievement of this work from which I gained the invaluable knowledge. Secondary, I wish to thank Department of Information Technology, COLLEGE OF SCIENCE AND TECHNOLOGY, UNIVERSITY OF RWANDA, especially the UR authorities for allowing me to pursue this Masters and for the skills they gave to me. I also wish to thank my family member who has done great sacrifice by providing me all the necessary means to have a good education. I also thank my classmates especial those we were together in groups in all academic activities for their direct and indirect contribution to the achievement of my objectives. Lastly, I would like to communicate my appreciation to the lecturers in my department for the good job done during the 2 years period of our courses. My acknowledgement goes to everyone who supported me in my everyday life to reach this moment.

May the Almighty God bless you all!

ABSTRACT

Clustering is a widely used approach to ease implementation of various problems such as routing and resource management in Ad-hoc networks. Today, Ad hoc wireless networks perform the difficult task of multi-hop communication in an environment without a dedicated infrastructure, with mobile nodes and changing network topology. After determining the cluster members and cluster head, distance and energy for each node are carried out per cluster head and cluster and data gathered from the cluster members are relayed to the main server. The result has to show that the proposed algorithm has less number of dropped packets, uses high energy during the path discovery than the widely used Ad hoc On-Demand Distance Vector (AODV) and handles the fault tolerance problem during the packets transmission.

In this project, I proposed a new algorithm for clustering in Ad-hoc networks that merges networks nodes to form higher level clusters by increasing their levels, their energy and their density. I showed the operation of the algorithm and analyzed its time and message complexities and provided results in the simulation environment of *ns*2. The results confirmed that the algorithm proposed is scalable and has a lower time and message complexities than the other algorithms.

Keywords: Routing Protocol, Ad hoc networking, Clustering, Cluster Density, Fault Tolerance in Ad-hoc Networks, Performance metrics, Dropped packets and Throughput.

LIST OF ACRONYMS AND ABBREVIATIONS

3G: Third Generation

3GPP: 3rd Generation Partnership Project

4G LTE: Fourth Generation Long Term Evolution

AODV: Ad hoc On-Demand Distance Vector

CCI: Cluster Contention Interval

CH : Cluster Head

CSMA: Carrier Sense Multiple Access

D2D: Device-to-Device

DEST: Destination

DSDV: Destination-Sequenced Distance-Vector Routing

DSR: Dynamic Source Routing

E2FT: End-to-End Fault Tolerant Routing

FDD: Frequency Division Duplexing

FTCH: Fault-tolerance Cluster Head Based

FTRA: Fault Tolerant Algorithm

HSPA: High Speed Packet Access

INs: Intermediate Nodes

LAN: Local Area Network

LCA: Linked Cluster Architecture

LTE: Long Term Evolution

LTE-A: Long Term Evolution Advanced

MANET: Mobile Ad-hoc Network

MMMH: Max-min multi-hop routing protocol

NS-2: Network Simulation 2

QoS: Quality of Service

RAM: Random Access Memory

SRC: Source

TACA: Topology adaptive clustering algorithm

TDD: Time Division Duplexing

TDMA: Time Division Multiple Access

LIST OF SYMBOLS

ρ : Node Density

n : Number of nodes

A : Size of Network Area

d_{min}: Minimum degree of nodes

P : Probability of the connectivity,

π= 3.142 represents the circumference

r_0^2 : Transmission radius.

μ : The expected number of neighbors of a node

E_n: Energy consumed by node in cluster in Ad-hoc network

E_{amp}: Amplifier coefficient

k: A number of transmitted data bits

d: distance between a sensor node and its respective cluster head or between a CH to another cluster head nearer to the BS or between CH and BS.

PDR: Packet Delivery Ratio

PRCD: Packets Received by Cluster Destination

PSCS: Packets Sent by Cluster Source

EED: End-to-End Delay

TSn: Time at which data packets n has been sent

TRn: Time at which data packets n has been received

N : Total number of data packets received

P: Number of packets successfully received at the cluster Destination

T : Unit time

DPD: Data Packets Delivered

RCP: Routing Control Packets.

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

CHAPTER ONE: GENERAL INTRODUCTION

1.1. Introduction

Ad hoc network is a local network (LAN) that is built spontaneously as devices connect. Instead of relying on a base station to coordinate the flow of messages to each node in the network, the individual network forward packets to and from each other. For that reason, the Ad-hoc Networks play an important role, as they provide communication without any fixed infrastructure. The Ad-hoc network consists of a number of wireless nodes that are self-organized and do not require a constant structure. The movement of the nodes enables them to generate multiple routes [1]. Thus, accurate routes must be determined for these nodes. A new algorithm must therefore be developed to design a routing protocol that adapts to network topology changes. In order to support large number of devices, ad hoc networks should be organized hierarchically. Clustering organize the ad hoc networks hierarchically and create clusters of ad hoc nodes which are geographically adjacent. Each cluster is managed by a cluster head (CH) and other nodes may act as cluster gateway or cluster member. Each node in the network or cluster also acts as a router, forwarding data packets for other nodes. A central challenge in the design of ad hoc networks is the development of dynamic routing protocols that can efficiently find routes between two communicating nodes. The routing protocol must be able to keep up with the high degree of node mobility that often changes the network topology drastically and unpredictably. The ad-hoc network contains many kinds of routing protocols, namely, reactive (on demand), proactive (table driven), and hybrid routing protocols. In the reactive routing protocol (AODV), a route is created only as needed. In the proactive routing protocol (DSDV), the route is prepared in advance, and the details are listed in a specific table [2]. The hybrid routing protocol combines the two previous types of routing protocols. This type of routing protocol is used in clustering. A cluster uses the DSDV to locate internal paths, whereas the AODV is employed to determine routes to other clusters [3]. The cluster density is defined as the expected number of clusters per unit area, and the cluster order is the number of nodes belonging to a cluster. In the ad hoc network, due to node mobility the topology of the network can change frequently, nodes can move away from the transmission range. So, there may be a chance of node failure or link failure and node have to consume more energy to transfer the packets from source to destination. The performance of the routing is decreased because of the failures. Node failure occurs when lack of power it causes route failure in the network. One of the main issues is the nodes movement and the dynamic change that occurs in the network topology. This project is aiming at designing a novel high density, fault tolerant cluster based routing algorithm in ad hoc network with varying network node densities which supports fault tolerance when there is node failure in the cluster. Because Fault tolerance is now a relatively mature field of computing that can rely on a significant body of theoretical and experimental results, this new algorithm must support the addition and deletion of dynamic nodes, ensuring the continuation of service despite the presence of different types of failures and the transparency of the process proposed for the users. In this thesis, the design, simulation using simulation tool such NS-2 network simulator has been performed together with the results.

1.2. Background

Currently wireless networks have grown significantly in the field of telecommunication networks. In the last few years, the use of mobile devices and mobile networks has grown dramatically and with the emergence of low cost wireless networking, a wide range of applications are been incorporated in laptops, mobile phones etc. Today, wireless communication between mobile users is becoming more popular than ever before. This is due to the recent technological advances in laptop computers and wireless communication devices, such as wireless modems and wireless LANs [4]. This has led to lower prices and higher data rates, which are the two main reasons why mobile computing continues to enjoy rapid growth. There are two distinct approaches for enabling wireless communication between two hosts. The first one is to let the existing cellular network infrastructure carry data as well as voice. The second approach is to form ad-hoc network among all users wanting to communicate with each other. This means that all users participating in the ad-hoc network must be willing to forward data packets to make sure that the packets are delivered from source to destination. This form of networking is limited in range by the individual nodes transmission ranges and is typically smaller compared to the range of cellular systems [4].The Ad-hoc networks have several advantages compared to traditional cellular systems. These advantages include:

- On demand setup
- Fault tolerance

The Ad-hoc networks do not rely on any pre-established infrastructure and can therefore be deployed with no infrastructure. Because nodes are forwarding packets for each other, and the nodes can be grouped together to form a cluster, some sort of routing protocol is necessary to make the routing decisions within that formed cluster [4]. There exist many routing algorithms in ad-hoc network. This thesis looks at some problems and tries to evaluate one of the routing algorithms within ad-hoc networks called “Novel high density, fault tolerant Protocol for cluster based routing in Ad-hoc Networks”.

1.3. Problem Statement

Ad hoc networks require a highly adaptive routing scheme to deal with the frequent topology .As of today; the last decade has witnessed an unprecedented growth in both the amount of mobile broadband traffic and the user demand for faster data access. The problems identified in this study on Ad hoc networks with varying node densities are: Low packet delivery, low throughput, high end to end delay and potential broadcast storm problems due to unmanaged network broadcasting soliciting very high number of routing overheads in highly dense areas of the network and discovering the system failure (node failure). To solve this problem, a Novel High Density and Fault tolerant cluster based routing: Algorithm, Design and system level performance is proposed. In this thesis also, I propose to find more relationships between these new metrics (Density and Fault tolerance) and other metrics for the designed algorithm. In addition to this, comparative tests between node density and connectivity should be performed to see if one of the metrics should be performed over the other.

1.4. Project Objectives

1.4.1. General Objective

To design, analyze and simulate a cluster based routing protocol in Ad hoc network with high density, and supports fault tolerance and connectivity metrics.

1.4.2 Specific Objectives

- To run a simulation using a simulation tool such as NS-2 Simulator and analyze the results.
- To test the parameter metrics such as the Density and Fault tolerance for the proposed algorithm.

- To develop an algorithm with high density that handles fault tolerance mechanism in case of node failure within the clusters or nodes.

1.5. Scope and Limitation of the Project

This research project will be mainly focused on the Design of novel high density, fault tolerant protocol for the cluster based routing in Ad hoc network. The routing scheme will be realized by clustering the terminals (nodes) according to some predefined physical layer parameters (such as density of nodes, Fault tolerance and connectivity metrics). It will also limit the number of broadcast reiterations performed based on the density of the nodes in the same area. This cluster based routing algorithm is not going to analyze and compare performance of the other cluster based routing protocols like AODV,DSDV but it is about designing the new one. Current broadband wireless technologies like LTE, LTE-Advanced are not included in this project, it will concentrate only on cluster based routing protocol in Ad Hoc Networks. This new algorithm will be analyzed theoretically and experimentally using a simulation tool.

1.6. Expected Results

From this project, I expected to design a new routing algorithm in ad-hoc networks with two factors: Cluster density and Fault tolerance. This is done by increasing the number of nodes in a given cluster. Designing this algorithm will help the communication among the nodes when the number becomes big and when the faults occurs in the cluster.

1.7. Justification of the Study

In many networks especially in cellular networks, clustering and identification of routing protocols is difficult. That is why there is a need to identify a new routing algorithm for a cluster. Some of the clustering algorithms were proposed to mitigate the “energy-hole” problem in cluster. Different kinds of nodes with different initial energy are grouped into clusters. Three major factors are taken into accounts in the process of cluster formation: (1) the optimal number of cluster heads; (2) the balanced average energy consumption and (3) the requirement of no isolated nodes. Due to the problem that big networks are facing like packet collision, performance degradation, clustering is one of the most familiar mechanisms to handle this. It gathers numerous nodes into many sets called clusters to reduce loads in connections and to eliminate power consumption in large networks. In the clustering structure, each cluster has one node that is regarded as the CH. This node manages the selection of an appropriate path for any node in a particular cluster. By reducing the number of broadcast iterations according to the density of the network will help dense networks to reduce the amount of packet collisions that may occur due to a “broadcast storm”. That is why a novel high density protocol for cluster based routing in Ad hoc network has been proposed.

1.8. Organization of the Project

This work is articulated around four chapters:

- ✓ Chapter 1: General Introduction, this chapter focuses on Objectives of the project, Problem statement, project Scope, Hypothesis and Methodology of the project.
- ✓ Chapter 2: Literature review, this offers theoretical concepts regarding the related work done by the other researchers.
- ✓ Chapter 3: Research Methodology; this focuses on software development methodology that can be used on the project and the data gathering techniques.
- ✓ Chapter 4: Analysis and Design; this chapter focuses on the analysis and the design of the project.
- ✓ Chapter 5 focuses on the results of the projects and discussions
- ✓ Chapter 6: The last chapter is made up of the conclusion and recommendation for further improvements in the project.

1.9. Conclusion

In this chapter, the brief description of the project has stated including the objectives, problem statement and the techniques or methodologies that will be used to run a simulation for getting the results. It also states the main assumptions that the project are planning to achieve at the end of the chapter 4 which is design and implementation. The related works and literature review are detailed the following chapter.

CHAPTER TWO: LITERATURE REVIEW

2.1. Introduction

This chapter of literature review expresses the work done by other researchers, and then provides an overview of overview of definitions, characteristics of concepts and technologies used in the development of novel high density, fault tolerant Protocol for cluster based routing in Ad-hoc Network.

2.2. Overview of Density, Fault tolerance and connectivity for cluster in Ad-hoc networks

2.2.1. Ad-hoc network

A network in ad-hoc mobile network consists of the source (SRC), destination (DEST) or/and a number of intermediate nodes (INs). Movement by any of these nodes will affect the validity of the route. A source (SRC) in a route has a downstream link and when it moves out of its downstream neighbor's radio coverage range, the existing route will immediately become invalid. Hence, all the downstream nodes may have to be informed so as to erase their invalid route entries. Likewise when the destination (DEST) moves out of the radio coverage of its upstream neighbor, the route becomes invalid. However, unlike the SRC, the upstream nodes will have to be informed so as to erase their invalid route entries.

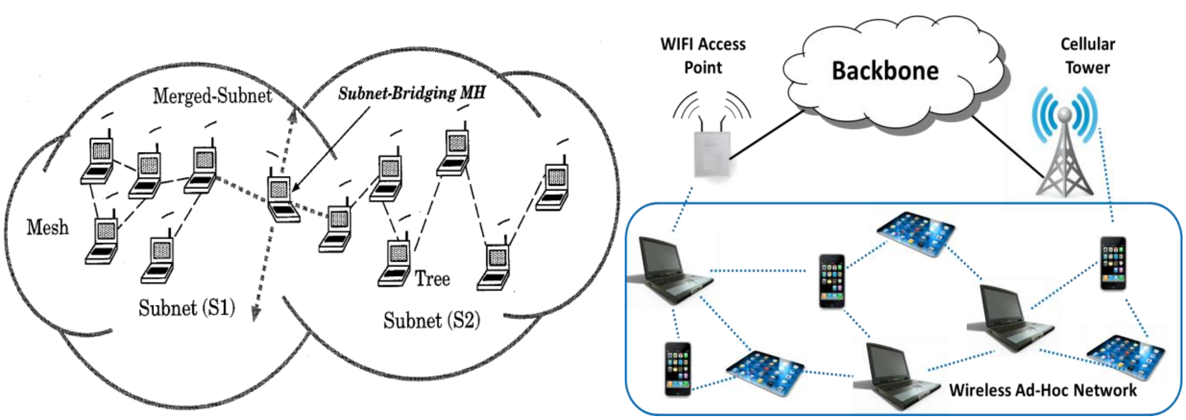

Figure 2.1: Ad-hoc mobile subnets merging and fragmenting and Wireless ad-hoc network

Lastly, similar to the source (SRC) and destination (DEST) nodes' movements, any movements by one of the INs supporting an existing route may cause the route to become invalid. All these nodes' movements cause conventional distributed routing protocols to respond in sympathy with the link changes, in order to update all the remaining nodes within the network, so that consistent routing information can be maintained. However, this involves broadcasting over the wireless medium which results in wasteful bandwidth and an increase in the overall network control traffic [8].In such networks, all the nodes connect to an access point which usually has a wired connection to the Internet. From the point of view of the network and higher layers, this first hop can be approximated as an Ethernet-type shared medium. In this scenario the nodes connected to the same wireless LAN communicate with each other only indirectly. There are, however, many important applications where this model is not applicable. First, even if the goal is Internet access, the access point might not be able to cover all the relevant mobile nodes due to limitations in transmission range, cost or access rights considerations. Another case is when Internet access is not desired (or is secondary importance), the main application being to communicate locally among a group of (potentially mobile) nodes [13]. In summary, the Ad-Hoc network is a group of nodes which form a temporary network without the aid of any fixed infrastructure or centralized administration[9][10].The configuration of this network can be static or dynamic.

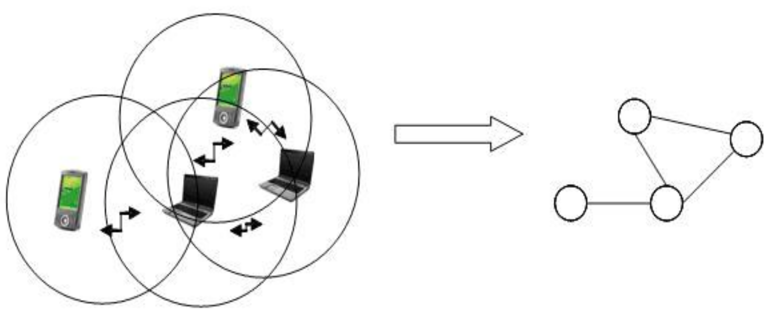

Figure 2.2: Ad-hoc network architecture

2.2.2. Density and Fault tolerance in cluster

2.2.2.1. Node Density

The node density for an entire network can be identified as the number of nodes that populate over a certain area or region of an Ad-hoc network [11]. The node density impacts routing evaluations since it determines, together with the mobility model, how many neighbors a node has [12]. Therefore, the node density can be equated to:

$\boldsymbol{\rho}$ Node Density

$\boldsymbol{n}$ Number of nodes

$\boldsymbol{A}$ Size of Network Area

$$\boldsymbol{\rho} = \frac{\boldsymbol{n}}{\boldsymbol{A}} \tag{2.1}$$

According to the formula (1), the node density is calculated as the ratio of the number of nodes in the cluster and the size of the network area.

2.2.2.2. Approaches of Fault tolerance in Ad-hoc networks

The Ad-Hoc networks where nodes move freely causing connection failures and frequent node failures. The failure of some links and nodes considered as criticism can divide the network into several partitions. The main objective in the design of a fault-tolerant routing protocol is that if a route breaks (due to node and/or link failures), there is always (at least) one more route available for a source destination pair. The main challenge that a fault-tolerant routing protocol should address is coping with node and/or link failures without incurring high overhead. This reduces the availability of data and leads to inconsistent data. Data replication in Ad-Hoc networks must address the additional problems resulting from the constraints imposed by the environment of Ad-hoc network [14] [15]. These problems are:

- Problem partitioning of the network: Partitioning reduces the chances of access to a data because mobile users may not be in the same partition as the node holding the data. Replication of data in the partitions before the occurrence of network partitioning can improve the availability of data. To do so, the replication define the response time of network partitioning may occur and replicate data in advance.
- Problem of energy consumption: In mobile communications, an important parameter is the energy under the battery, in fact the Ad-hoc networks are autonomous and independent so they do not opportunity to have an infinite energy as it once consumed cannot be recovered or reloaded. To increase the availability of data, the replication protocol must replicate a critical data on the nodes that can last for a long time.
- Problem scalability: As the network size increases, a request sent by a client node may have to traverse a long path to reach the data, thus increasing the cost and latency of motion. Moreover, the existence of a large number of customers request causes

controversy over access channel, which reduces the available bandwidth and increases the access delay of the channel. The replication protocol should be designed so that its performance is not greatly accepted if the number of nodes or the network size increases.

The model proposed for tolerating faults in the Ad-hoc networks is a model which is shown in Figure 2.3 where if there is a failure in one nodes, other continue to forward the packets.

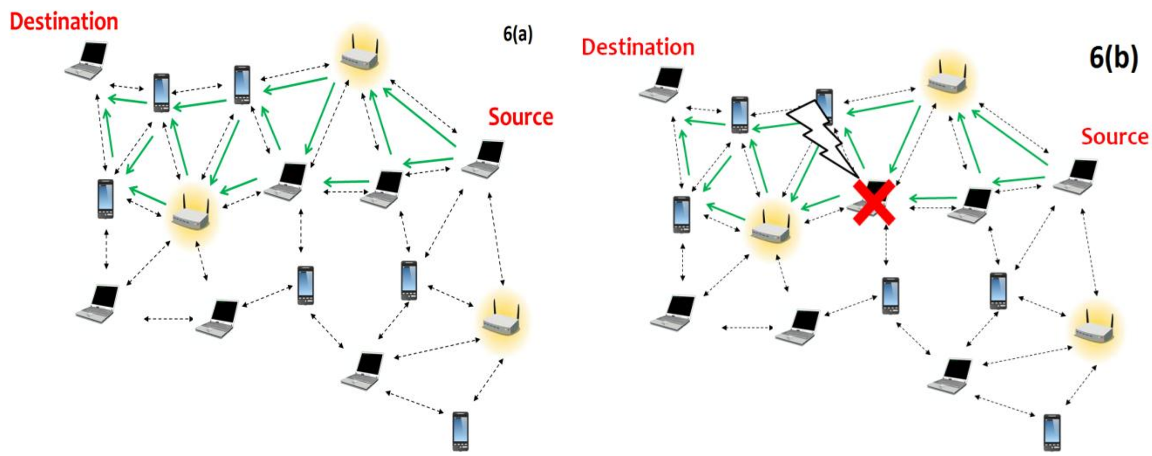

Figure 2.3: Fault tolerance in Ad-hoc Networks

2.3. Cluster based network architectures

Clustering is one of the most familiar mechanisms. It gathers numerous nodes into many sets called clusters to reduce loads in connections and to eliminate power consumption in large networks. The structure of the cluster based network that is used in ad hoc wireless networks is described below [40].In this system architecture, the entire network nodes are autonomously organized themselves into interconnected clusters where each cluster contains a cluster-head(CH) where this node possesses complete information about all of the nodes in the cluster and this information is stored in a member table, one or more gateways which is used to connect clusters**,** and zero or more ordinary nodes as shown in Figure 2.5. The cluster-head (CH) schedules transmissions and allocates resources within the clusters. There are also the Gateways that connect adjacent clusters. That gateway may directly connect two clusters as a member of both, or indirectly connect two clusters as a member of one and forming a link to a member of the other. Therefore architecture can contain both overlapping and disjoint clusters.

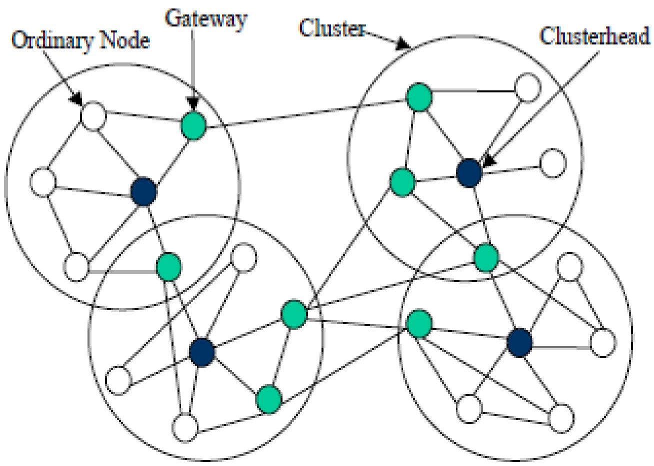

Figure 2.4: Link-Clustered Architecture

Nodes establish a link-clustered architecture over a physical network by the following [16]:

- Discovering neighbors with bi-directional connectivity by broadcasting a list of those neighbors that they can hear and by receiving broadcasts from neighbors.
- Electing cluster-heads (CH) and forming clusters.
- Agreeing on gateways between all the clusters.

2.3.1. Cluster-Head (CH)

There are two proposed algorithms for the selection of cluster-heads, namely called identifier-based clustering and connectivity-based clustering. The implementation of both algorithms can be either distributed or centralized. For centralized, the node with the lowest or highest numbered identifier (identifier based) or with the largest number of neighbors (connectivity based) is chosen as a cluster-head for the cluster containing that node and its one-hop neighbors. For the distributed version of identifier based clustering, a node elects itself if it has the lowest or highest numbered identifier in its neighborhood. With the distributed version of connectivity-based clustering, a node becomes a cluster-head if it is the most highly connected of all of its neighbors which are not elected as cluster-heads [16].

2.3.2. Node Mobility in Cluster based network

Because there is a movement of nodes about the network, cluster membership or nodes must be updated accordingly to ensure proper scheduling of transmissions. Clustering algorithms described by Baker, Ephremidis, Wieselthier, Gerla, and Tsai estimate cluster membership and cluster-head (CH) and the status of the gateway whenever a node moves in or out of a cluster. Identifier-based clustering methodology is more stable than connectivity-based clustering methodology in the cluster re-computation approach. This is because a node's cluster-head (CH) status may change more frequently with connectivity based clustering [16].The least cluster change algorithm reduces the number of changes in cluster-head (CH) status due to node movement. In this algorithm, a change in cluster-head (CH) status occurs only if two cluster-heads (CH) move within the range of each other and in this case one of them relinquishes its cluster-head (CH) or an ordinary node moves out of range of any other node and becomes a cluster-head (CH) for its own cluster [16].

2.3.3 Routing

The figure above of link-clustered architecture, it provides a natural routing backbone consisting of cluster-heads and gateways and the links between them. However, cluster-heads as points of traffic concentration may become congested and each cluster-head may become a failure point for communication across its cluster. Therefore, link clustered architecture is not used as routing control architecture in routing algorithms. Instead, each node distributes and collects routing information, and generates and selects routes. However, clusters exists to define regions for transmission management primarily and to form a routing backbone in link-clustered Architecture.

2.4. Related works for density and Fault tolerance cluster based routing protocols in Ad-hoc Networks

2.4.1. Related works for Routing Algorithms in Ad-hoc Network

2.4.1.1. Concepts of Routing Protocols in Ad-hoc Networks

The purpose is to apply the clustering solution to a routing problem encountered in the routing after doing the clustering. Wireless routing protocols generally focus on ad-hoc and mobile ad-hoc (MANET) routing [35]. Ad-hoc routing means that there are no routers positioned on network borders which handle routing between networks and there are no wires between nodes and routers that establish the network structure. Wireless routing algorithms are generally divided into proactive and reactive protocols [36, 37]. Proactive routing protocols (e.g. DSDV

[38]) build routing tables continuously, regardless of need [39]. In proactive routing protocol, when a network node is ready to send data, it can consult its pre-existing routing information and send the data. Reactive (or on-demand) routing protocols (e.g. AODV [39], DSR [42]) collect routing information as needed instead of continuously, that is, they react to a routing request and build the necessary routes [36]. This is not to say that a reactive routing protocol cannot cache collected information for later use (for instance [39], but that information is not discovered initially or updated except in response to a routing request. At request time, if the routing information is up-to-date, it can be used as is.

2.4.1.2. Related works for routing protocols

In the year 2003, P. Anju and M. Silpa proposed a paper on “A Distributed Efficient and Accurate Routing Protocol to Increase the Capacity of Hybrid Wireless Networks”[7], in a hybrid wireless network that combines a mobile ad-hoc network and an infrastructure network, efficient and reliable data routing is important for high throughput. Existing routing schemes that simply combine ad-hoc and infrastructure routings inherit the drawbacks of ad-hoc routing including congestion and high overhead for route discovery and maintenance. In the absence of the central control infrastructure, in mobile ad-hoc network the data is routed to its destination through the intermediate nodes in a multi-hop manner. The multi-hop routing needs on-demand route discovery or route maintenance. The messages are transmitted in wireless channels and through dynamic routing paths; the mobile ad-hoc networks are not reliable as the infrastructure wireless networks. Because of the multi-hop transmission feature, mobile ad-hoc networks are only suitable for local area data transmission. The infrastructure wireless network (e.g. cellular network) is the major means of wireless communication in our daily lives. In the inter-cell communication (i.e., communication between nodes in different cells) and internet access. It makes the support of universal network connectivity and ubiquitous computing by integrating all kinds of wireless devices into the network [7].

2.4.2. Related works for Clustering Algorithms in Ad-hoc Networks

There is a concept of partitioning of the random dynamic network into logical clusters (also called as the Linked Cluster Architecture (LCA) which was initially proposed by Baker and Ephremides [6]. The existing one-hop clustering algorithms emphasize either on minimizing number of cluster-heads [43,44] in the virtual back bone to reduce the routing delay or maximizing the cluster stability by un-altering the head nodes [45,54]. A small variation to LCA was proposed by Ephremides, Wieselthier and Baker in [45] as a lowest ID algorithm. In this algorithm, a node which has the lowest ID among its neighbors is selected as the head node (Cluster-head). It retains its utility as a point of reference for producing reasonably stable cluster control architecture as written by Gerla and Tsai in [46]. However, as the node ID is the only deciding factor for a node to be a cluster head, the lower ID nodes are influenced to become the heads all the time resulting in their faster energy drainage which may perturb the cluster stability. A modified version of LCA was proposed by Parekh [47] that aims to reduce the number of clusters in the network. If {Ni} represents the set of neighbors of a particular node *i*, then the degree of connectivity of *i* is represented as Di= |Ni|, where |Ni| is the cardinality of {N_i}.A higher degree of connectivity ensures lower delay in communication through cluster heads (CH). The mobility of nodes changes the degree of connectivity of the node very frequently which leads to more number of cluster head reelections as well as link updations resulting poor cluster stability.

Another routing methodology which is mobility based metric for Clustering version of lowest ID algorithm (MOBIC) was proposed by Basu, Khan and Little [48]. The algorithm uses mobility based metric for calculation of weights of the nodes by using the ratio of two consecutive signal strengths received by a node to know its relative motion with respect to its neighbors. When two cluster heads accidentally come within their transmission range, re-clustering is deferred for Cluster_Contention_Interval (CCI) period as per the LCC [49] algorithm. Though MOBIC provides a better cluster stability, but the need to collect relative speed information of a node from its neighbors degrades its performance. The author in [49, 50] proposed a weight based distributed mobility adaptive algorithm (DMAC) that removes the non-mobility assumption of the hosts during clustering setup and maintenance. However, when two head nodes come within the transmission range of each other; the node with the lower weight has to resign its role as head and is forced to become the member node of the node with higher weight. The combined metric clustering algorithms [51,52] use some node parameters like running average, degree of connectivity or mean connectivity, transmission power, available battery power or consumed battery power to find its suitability as a head. But obtaining so much of information to compute the combined weight for every node in the network needs a longer frozen period of motion before the cluster is actually formed. A large number of message exchanges take place globally to yield the node with lowest weight. The authors of [55] have proposed a topology adaptive clustering algorithm (TACA**)**. As the network is activated, clusters are formed with the election of volunteer cluster heads. When any of these volunteer head drains a threshold amount of its battery power then it selects a non-volunteer head within its own cluster so that it can hand over the responsibilities to the newly selected head locally. This avoids as much as possible to a global cluster head election procedure that demands considerable computation and communication overhead. A clustering algorithm known as Lowest-ID (cluster head selection based on ID) has been proposed in the past [57] and has been revisited later in the "multimedia multihop mobile networks" context [56]. In [5], Chiang et al. have shown that the Lowest-ID algorithm performs better than the maxconnectivity algorithm in terms of stability of clusters (measured by the number of cluster-head changes), and they have proposed a small change in the Lowest-ID algorithm to improve performance; the improved version is referred to as LCC which stands for "Least Cluster-head Change".

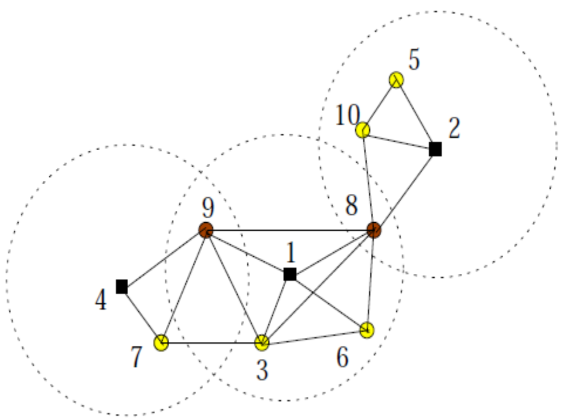

Figure 2.5: Lowest ID Clustering Algorithm

Lowest-ID clustering is one of the most popular clustering schemes used in the old [57] as well as recent ad hoc networks. The Lowest-ID algorithm proceeds as follows and results in the formation of clusters which are at most two hops in diameter:

- Each node is given a distinct ID and it periodically broadcasts the list of its neighbors (including itself).
- A node which only hears nodes with ID higher than itself is a "cluster-head" (CH).

- The Lowest-ID node that a node hears is its cluster-head, unless the Lowest-ID specifically gives up its role as a cluster-head (deferring to a yet lower ID node).
- A node which can hear two or more cluster-heads is a “gateway”.
- Otherwise, a node is an ordinary node.

A node is identified by its cluster-head’s ID. Figure 1 shows a schematic of the result of using Lowest-ID clustering. Associativity based routing (ABR) protocol that uses mobility for routing has been proposed for mobile ad hoc networks [58]. ABR does not define mobility metric but uses a route stability metric that is used by the routing protocol.

2.4.3: Related works for Fault tolerance algorithm in cluster based routing protocol

Due to the mobility of the nodes and the associated rapidly-changing topologies, the reliability of the correct transmission of messages is an important concern for Ad-hoc networks. To overcome the node failure many algorithms and protocols to be used in the Ad-hoc networks. Xue and Nahrstedt [17, 18] confirmed that devising a fault-tolerant routing algorithm for ad hoc networks is inherently hard. This is because the problem itself is NP-complete due to the unavailability of “correct” path information in these environments. In [17], they designed an efficient algorithm, called the End-to-End Fault Tolerant Routing (E2FT) Algorithm, which is capable of significantly lowering the packet overhead, while guaranteeing a certain packet delivery rate. Following the work of Xue and Nahrstedt [17, 18], Oommen and Misra [19] proposed weak-estimation learning based fault-tolerant routing protocol for Ad-hoc networks. Very recently, Misra et al. [20] also proposed a low overhead ant-swarm inspired routing protocol for Ad-hoc network. In [21] Zhou,Xia proposed an algorithm called location based fault tolerant routing algorithm (FTRA). In this algorithm based on geographical location information networks divided in to grid. Fault may occur the proposed algorithm select alternate route from unused at hop in normal routing path, the route selection depends upon location information of its neighbors grids. In [22] Qin and Pang proposed fault-tolerance cluster head based (FTCH) routing protocol reduce misbehaving node in the network. Faulty node occurs the proposed protocol provide packet delivery fraction guarantee and reduce routing overhead. The performance analysis of FTCH compared with MMMH, AODV and DSR. Rouzi et.al [23] introduces a graph called fault-tolerant 1-spanner. It is used to protect minimum energy paths. Also called as k-Fault-Tolerant 1-Spanner it generated minimum energy path tree for failed node set. To remove the failed nodes in the network the initial network will get energy form it. Modified Cluster-based QoS routing algorithm [24] with the goal of providing fault tolerance in Quality of Service (QoS). This algorithm evaluates node failure based on failure recovery time, throughput, dropped packet and flow bandwidth. Nazeeruddinet. al [25] propose a distributed agent based dynamic host auto-configuration protocol. Every node has address, the IP addresses can be allocate to the new node, without check with further address agents. This protocol handles message losses and node failure. Melamedet..al[26] proposed Octopus a fault-tolerant and efficient position-based routing protocol is suitable for failure prone environments. In this protocol frequently update node location using flexible state and achieves low location update overhead by using novel aggregation technique. Octopus applicable for fixed node density and not raise the network size. In [27]Khazaei and Berangi increase the data transfer and fault tolerance by creating backup path between source and destination during route reply, route maintenance and local recovery. Chandrasekaran et.al [28] proposed a model Trusted Fault Tolerant (TFT) used in Location Aided Routing (LAR) protocol. LAR focused link faults, high mobility, node congestion and capacity of buffers. The Location fault occurs the destination node move away from source. The proposed model considers that node is selfish or misbehaving node. It improves the location awareness and node trust level based on recovery of the lost packet. In [29]Rana and Ahamed proposed a new fault tolerant routing protocol extends of DSR protocol. In this protocol identify at least two paths between sources to destination. Link failure occurs in one path immediately select another path. Shaji et.al [30] [31] proposed a novel Self-eliminating Fault-tolerant based Un-interrupted reliable Service switching mobile Protocol (SFUSP) it includes the task of clustering and self-elimination to creates a reliable route in heterogeneous network and identify the link break in early. Mutual Exclusion (MUTEX) algorithm [32] used to tolerate link or host failure using the time out based method. In [33] investigates communication failure in grid based topology using distributed genetic algorithm also use simple retry and reroute protocol to solve the communication failure in the network. In [34] proposed a Trusted Fault Tolerant (TFT) model based on Location Aided routing protocol along with user recovery features. It covers link failures during packet forwarding and location failures.

2.4.4: Related works for Density based in Ad-hoc networks

The network node density for an entire network can be differentiated into physical density versus connectivity density. In this study network physical density is defined as dense when large number nodes are in proximity of one another within a particular area and vice versa for sparse. However, when determining density for a particular network, one should also consider the connectivity of the network in terms of transmission range that covers the particular area. Thus the network density determined in this study, is based on the number of nodes found in a particular area and the connectivity of the nodes. Therefore even though the number of nodes found in a small area may not be packed, given a high transmission range then it can be determined that the node in the area is dense. Otherwise given either a very large or low connectivity the node density could be determined as sparse. On the issue of connectivity density, studies by (Bettstetter and Zangl, 2002; Bettstetter, 2002) discussed determining the network connectivity based on the density of the numbers of neighboring nodes. The density is defined based on the transmission range of the nodes. The definition of connectivity density of a network based upon the study is as follow:

- ✓ The number of neighbors surrounding a node is denoted by its degree d
- ✓ A node that has a degree d = 0 is said to be isolated from the rest of the network
- ✓ The minimum degree of nodes d_{min} and is considered as the smallest degree of all nodes in the network.
- ✓ A network is said to be connected when every pair of nodes exists a path between them, otherwise it is disconnected.
- ✓ A connected network always has a minimum degree $d_{min}>0$ but the reverse implication is not necessarily true.
- ✓ A network is k-connected if for each pair of node exists k mutually independent paths connecting them.

P is the probability of the connectivity. The value n is the number of nodes located in the area. The value μ is represented by formula 5 where ρ is the density, π represents the circumference and r is the radius of the transmission:

According to the above authors, the density is calculated by the following formulas 4-5 which state that:

$$\mathbf{P(k\text{-}con) \approx (1-e^{-\mu})^{n}} \quad (2.2)$$

$$\mathbf{\mu = \rho \times \pi \times r_0^2} \quad (2.3)$$

$$\mathbf{\rho = n/A}$$

where P =Probability of the connectivity, **n** = Neighbor count, **A** = Pre-defined area size, **ρ**= density, ***π***= 3.142 represents the circumference and $\mathbf{r_0^2}$ = Transmission radius.

For their study, they set the value of k to1 which means that for any particular network which has the probability of the connection of P(k-con)≥0.95 where k = 1, the network area is considered dense and there exists 1 mutually independent path connecting the nodes in the particular network area. Thus the network is mentioned as 1-connected. This also implies that for any neighbors found within the transmission range of a particular node they are at most 2 hops away from each other. The node density of the network areas is calculated based on the formulae provided for P(k-con). Therefore an area is considered dense when a node identifies that:

- ✓ It neighbors are at most 2 hops away from it and it has a mutually exclusive path to other neighboring nodes that is independent of one another. Thus P(1- con)≥0.95

2.5. The Summary and Identified Gap

The following are the problem facing the existing clustering algorithms in Ad hoc network:

1. Difficult in selecting the cluster-head (CH) in the network and determining the paths done by the nodes in the cluster.
2. Some studied algorithms in the above do not deal with the node mobility. It means that if nodes change their location over time, they have to update their location estimates frequently in order to avoid inaccuracies resulting from using outdated location estimates. Moreover, node movement during the measurement of parameters needed for location computation can cause inaccuracies in the estimated location.
3. All the above researchers did not develop a cluster based routing algorithm and the following parameter metrics on that protocol:

- ✓ Novel High Density of the nodes within the cluster
- ✓ Fault tolerance when there is node failure within Ad-hoc networks

Therefore, the simulation and analysis of novel high density protocol for cluster base routing in ad-hoc networks will solve those problems.

2.6. Proposed Routing Algorithm

In Ad-hoc networks, especially for the clustering mechanism, the high mobility, the nodes failure and density of nodes within the cluster pose challenge to establish an efficient and stable route for propagating information packets which is High novel density and Fault tolerance protocol for cluster based routing in Ad-hoc networks. For reducing the node failure in the cluster and maintain the nodes connectivity in the cluster, the proposed approach has use the idea of clustering in addition to reducing the flooding, overhead effects and minimizing the rate of link breakages in the established cluster. To reduce the network overhead, cluster size limited to the n-hop, member's nodes of cluster having maximum n -hop count link from the head node where depends on the network size.

2.7. Conclusion

In this chapter, an overview of the background of this research area is discussed. Related work to this research is presented in this chapter. Similarities and contrasting features with important conference and journal papers are also discussed. Some of the important considerations for a comprehensive performance that were not considered by some previous researchers are pointed out. The proposed algorithm was detailed in this chapter where by the design and the research methodology will be given and explained in the next chapter.

CHAPTE THREE: RESEARCH METHODOLOGY

3.1 Introduction

This chapter describes how the research will be conducted in order to achieve the stated objectives. It will demonstrate the research design and procedures, population and sample selection, data collection techniques and instruments, data processing and analysis significance of the study and limitations that were encountered during the research process. The scientific methods for conducting research have been used which are both qualitative and quantitative approaches as data analysis. Different experimental research approach was also considered because several simulation results have been presented using the simulation tools such NS-2 simulator.

3.2: Research Approach and Design of the System

This part describes the overview of the research approaches and the steps involved in system development from the step of gathering the ideas to the final step of simulations and getting result.

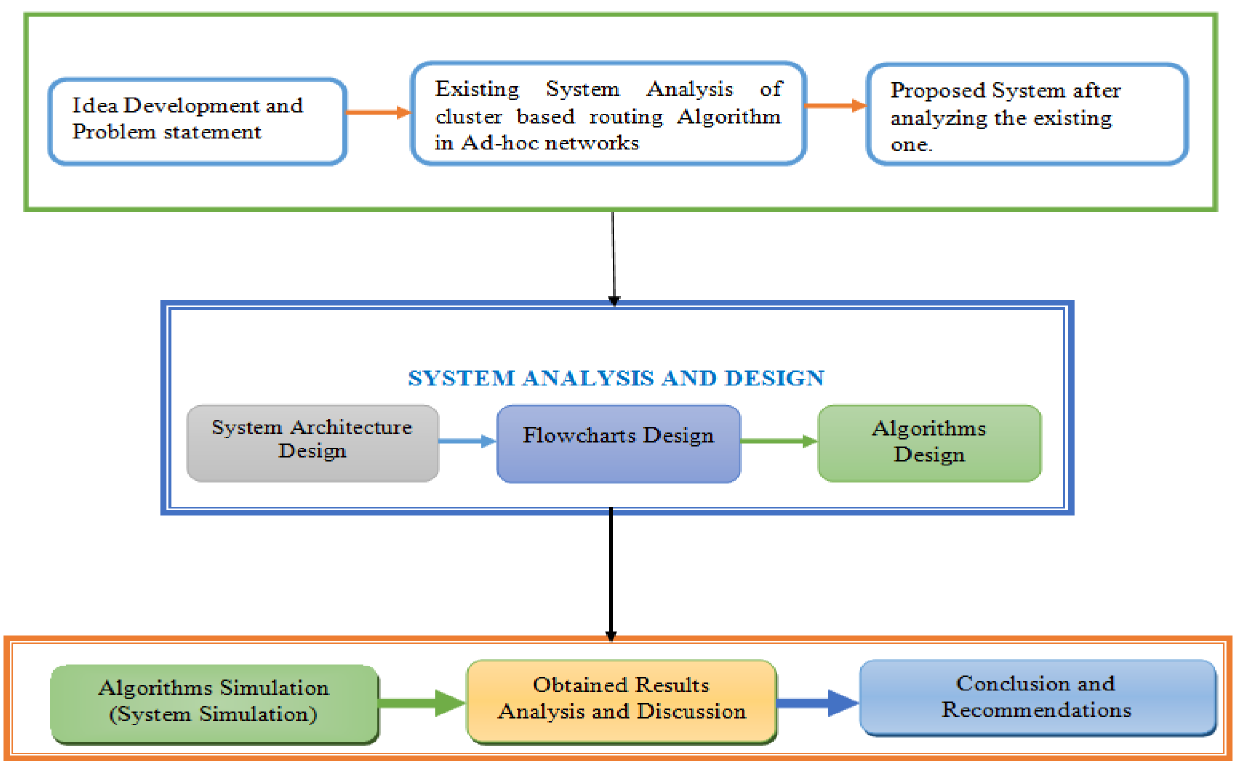

Figure 3.1: Overview of Research approaches development

The development approach in this research thesis has two types:

- The algorithm and flowcharts Design approach
- The simulation approach.

In this research thesis, the existing systems is analyzed and propose the new one where several formulae used in the algorithms were developed based on the Cluster density equation and inside the cluster, the Cluster head is elected based on different algorithms. Several flowcharts were designed to show the flow of information.

3.2.1: Scientific research methods

In this research thesis, I have used several scientific methods to conduct the research. The existing system has been found in qualitative research methods and design, analysis and simulation where an experimental approach was used. Firstly, the idea came with the objectives, problem statement and the proposition of the solution regarding the routing algorithm in Ad-hoc Networks. A qualitative approach have been used to do the evaluation of how existing system are working, what they don't do and which something to improve in the performance of the existing system for making the new system. The developed idea was that several nodes are grouped together and form clusters and in each cluster, a cluster head (CH) is elected based on algorithm and it is that CH which initiates the communication within the clusters. The routing algorithm is developed and simulation is done with different parameter as it will be discussed in the next chapter of this thesis.

3.2.2: Experimental approach

To be realistic, the experiments were done by using simulation tools such as NS-2 simulator. Under this method, discussion of the simulation results based on different performance metrics was done. Those performance metrics are such as: Packets Delivery Ration (PDR), Throughput, End-to-End delay, Number of dropped packets, Energy consumption, Packet sent and received. Some simulation tables and energy calculation of the cluster have been presented. In simulating this system, there were nodes formation (33 nodes) and derive clusters from those nodes (3 clusters) where in each cluster, a cluster head (CH) is elected based on different algorithms as shown in the following figure:

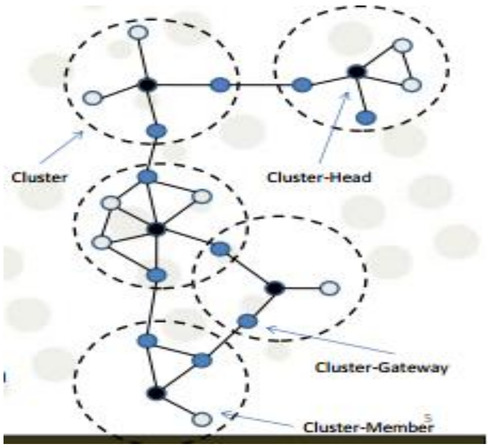

Figure 3.2: Cluster formation and Cluster head election

I tried to increase the number of nodes and test the performance of the propose system compared with the existing one and then plot the results in graphs called X-Graphs.

3.3: System Development Methodology

3.3.1: Prototyping Model

For the research methodology of this project, the chosen software development life cycle was Prototyping model which a systems development method (SDM) in which a prototype (an early approximation of a final system or product) is built, tested, and then reworked as necessary until an acceptable prototype is finally achieved from which the complete system or product can now be developed.

3.3.2: Steps of used SDM of my thesis (Prototyping Model)

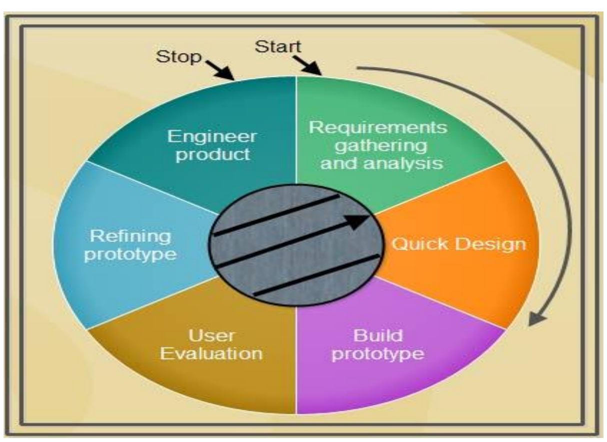

Figure 3.3: Steps of used SDM

1. Thesis Requirements gathering and analysis: For this thesis, I began with requirements analysis of the new routing algorithm for Ad-hoc networks. The details of the new system were defined in details. The users were interviewed in order to know the requirements of the system.
2. Quick design**:** After knowing the requirements of new routing algorithm, a preliminary design or quick design for the system was created which is shown in chapter 4. It is not a detailed design and includes only the important aspects of the system, which gives an idea of the system to the user. This quick design helped in developing the prototype (Algorithms and simulation).
3. Build prototype: All the Information gathered from quick design phase have been modified to form the first prototype (first routing algorithm in Ad-hoc networks, simulation results) which represented the working model of the required system. It is in this phase where also the simulation tools stated to be identified and being used.
4. User evaluation: Next step I did is presenting the first prototype to the user for thorough evaluation of the prototype to recognize its strengths and weaknesses such as what is to be added or removed. Comments and suggestions were collected from the users and provided to the developer who is me as researcher.
5. Refining prototype: After evaluating the prototype and if he/she is not satisfied, the current prototype was refined according to the requirements or comments or suggestions from the first presentation. A new prototype has been developed with the additional information provided by the user. The new prototype has been evaluated just like the previous prototype. This process continued until all the requirements specified by the user are met. A final system has been developed on the basis of the final prototype when the user was satisfied with the developed prototype.
6. Last in the used SDM is Engineer product: Once the requirements were completely met, the user has accepted the final prototype (final simulation results, algorithms).

3.4: Data Collection Methods

The following methods have been used during data collection: Interview and Documentation as my research methods. Through this, I was able to collect raw data on the new system**,** where information on the current system was obtained.

3.4.1: Documentation

In this method, I went to the library to search for books that contain the subject related to this topic. Not only the books used but also some electronic books (from the internet), the memoirs done, class notes, different journals and papers.

3.5: Data Analysis

1.1 After the data was collected, it was organized and analyzed. Simulation table was drawn and filled and from that, the data was put in simulation tool to get the results like the clusters, the X-Graphs and node energy.

3.6: Proposed system requirements

3.6.1: Functional Requirements

Novel high density, fault tolerant Protocol for cluster based routing in Ad-hoc Networks possesses some functional requirements such as:

- Formation of nodes when running the system.
- Grouping the nodes in clusters.
- Electing the CH in every formed cluster.
- Sending the packets from source to destination (node to node or cluster to cluster).
- Handling Fault tolerance problem within the Ad-hoc networks.
- Reporting the results using X-Graphs.

3.6.2: Non-Functional Requirements

The following are also the nun-functional requirements of the developed routing algorithm:
Performance: Response time, throughput, utilization.

- Scalability: This new routing algorithm is scalable because you can increase the number of nodes and clusters.

- Capacity: For this system, you can set its capacity in terms of number of nodes and clusters
- Usability: Every one with the knowledge of ICT can use this system.
- Recoverability: When something is wrong in the project, it is easily recoverable.

3.6.3: Hardware and Software Requirements

- ✓ **Hardware requirement**: The following are minimum hardware requirements for running novel high density, fault tolerant Protocol for cluster based routing in Ad-hoc Networks:
    - Computer with 2 GB RAM
    - 2 GB free space of Hard Disk
- ✓ **Software requirement**:
    - Unix/Linux/Ubuntu Operating System.
    - NS-2 (All versions are supported

3.7: Design process

The design process of new proposed Novel High Density, Fault Tolerant Cluster Based Routing: Algorithm, Design and System Level Performance in Ad-hoc Networks were very important phase after gathering the requirements and conducting the feasibility study by considering different factors. The design was specifically done after analyzing the existing schemes and then proposes a new scheme which was intended to archive our objectives.

3.8: Conclusion

This chapter of methodology covers the approach to carry out the research, methods of data collection, summary of data and its analysis and the processes to be undertaken when designing and simulating of the systems. By this stated methodology, I believe that the problem encountered in cluster based routing algorithms is going to be addressed. The proposed system analysis and design is going to be explained in the next chapter (Chapter 4).

CHAPTER FOUR: SYSTEM ANALYSIS AND DESIGN

4.1: Introduction

To analyse and design the Novel High Density, Fault Tolerant Cluster Based Routing: Algorithm, Design and System Level Performance in Ad-hoc Networks, some flowcharts and algorithms were used as prescribed in chapter 3 of the research methodology. In this project, there is only a single scenario which is composed of network nodes and the nodes are grouped together to form the cluster. Around 33 nodes and 3 clusters are enough to test the Algorithm where each cluster is composed of 11 nodes. For each cluster, a cluster head (CH) is elected which initiates the communication among the others. To elect a cluster head, the cluster energy of node is measured and calculated. As the objective was to design the fault tolerance and cluster density algorithm by increasing the number of nodes in the scenario, each elected cluster head starts the communication and when a faults occurs in the system, the algorithms designed searches for a new way so that the communication continues to happen by using the backup nodes. Every node in the network can communicate straightaway with the other nodes located in the transmission range. To overcome the node failure many algorithms and protocols to be used in the Ad-hoc networks has been proposed and designed. At the end, there is an analysis of the results to see if the cluster density and fault tolerance has been achieved.

4.2: System Model

As explained before in chapter 3, in this system, network nodes are grouped together to form the clusters. In my simulation scenario, only 33 nodes have been used which form 3 clusters where each cluster has 11 nodes. In each cluster, a cluster head is elected based on the algorithm. Each node in the group communicates regularly with the leader by sending his information: identifier, energy level, neighboring nodes, number of times he stopped, and the directory replicas that exist in this node. It's assumed that each node has a unique address, called the *ID_node,* a line refers to one given.

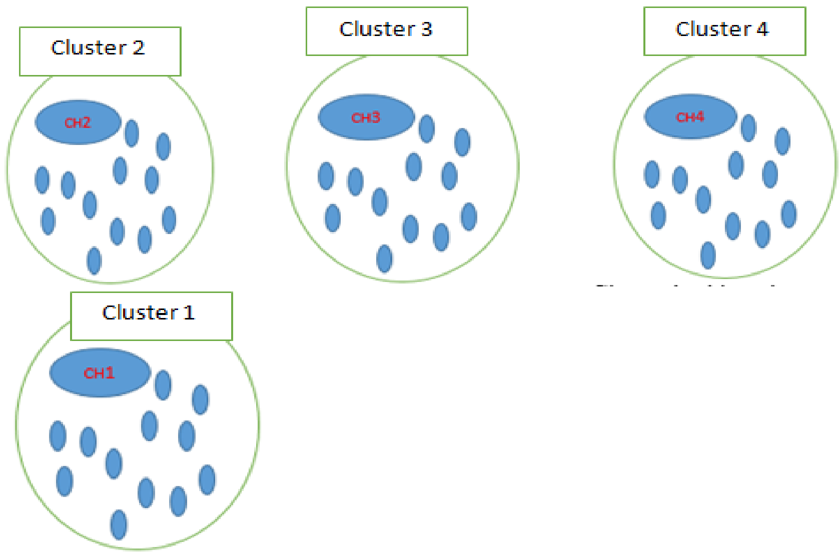

Figure 4.1: Cluster and CH formation within the Network

4.2.1: Approaches of Fault Tolerance in Ad-hoc Networks

The Ad-Hoc networks where nodes move freely causing connection failures and frequent node failures. The failure of some links and nodes considered as criticism can divide the network into several partitions. This reduces the availability of data and leads to inconsistent data. To increase the availability of data, a new protocol or algorithm has been designed to help the communication keeping goes on within the clusters. The model proposed for tolerating faults in the Ad-Hoc networks is a model which consists of two sub-services, which are presented in Figure 4.2 below:

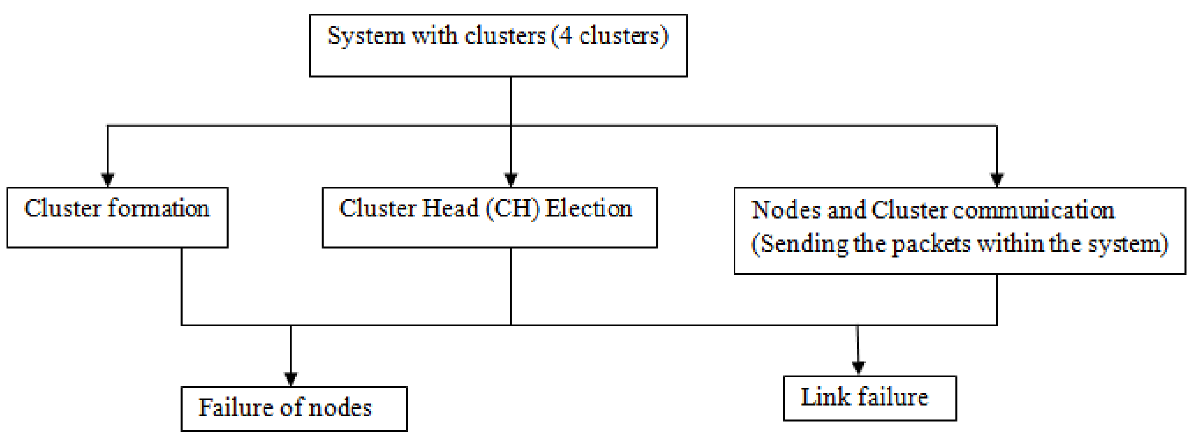

Figure 4.2: Proposed fault tolerance design

As shown in the figure 4.2, the failure can happen in two ways:

- ✓ Failure within the node
- ✓ Link failure

In this project, a fault tolerance routing algorithm is presented which applies the natural redundancy of Ad-hoc networks to increase the networks` fault tolerance. For this purpose, the proposed algorithm functions in two phases:

- ✓ Choosing the backup nodes and,
- ✓ Choosing the backup routes.

In the first step, the proposed algorithm uses the nodes with the same movement path as the backup nodes for the main nodes. Identifying and predicting the path of mobile nodes is conducted through the backup nodes` table. After selecting the backup nodes, the proposed algorithm starts fault tolerance. After that, from the source to the destination nodes, the backup path(s) between the backup nodes, which have already been identified by the proposed algorithm, usually route are created.

4.2.2: The Cluster Density

According to chapter 2 where we introduce the basic formulae to be use in this project to calculate the cluster in Ad-hoc network. The experiment showed that the cluster density, i.e., the expected number of cluster heads per unit area, is

$$\rho_c = \frac{\rho}{1+\mu/2} \qquad (4.1)$$

where μ denotes the expected number of neighbors of a node. Consequently, a cluster head is expected to incorporate half of its neighboring nodes into its cluster.

4.2.3: Energy Consumed by node within Cluster in Ad-hoc networks

The important factor in clustering algorithm is the energy consumption by nodes. A node with higher energy consumption fails faster. When a cluster head depicts its energy, it fails and also shatters the whole cluster and its links and entails lots of maintenance costs. On the other hand, cluster heads have the highest responsibilities among other nodes. Therefore, they need and consume the highest energy in network and are more likely to drain the energy. As a result, when choosing the clusters their remaining energy needs to be concerned. Taking this factor into account, the nodes with the highest residual energy and the least consumed energy among their neighbors are selected as the cluster heads.

The factor is given as follows:

$$\mathbf{E_n=E_{amp}*k*d_2} \tag{4.2}$$

where: $\mathbf{E_n}$: Energy consumed by node in cluster in Ad-hoc network

$\mathbf{E_{amp}}$: Amplifier coefficient

k: A number of transmitted data bits

d: distance between a sensor node and its respective cluster head or between a CH to another cluster head nearer to the BS or between CH and BS.

4.2.4: Packet delivery ratio Calculation for every cluster in Ad-hoc network

In this case, the CH initiates sending the packets between nodes and between clusters. The research used the formula to compute the Packet Delivery Ratio (PDR) which is the ratio between the number of packets transmitted by a traffic source and the number of packets received by a traffic sink. It represents the maximum throughput that the network can achieve in ad-hoc networks. A high packet delivery ratio is desired in an ad-hoc networks and it is computed by the following formula:

$$\text{PDR}=\frac{\sum PRCD}{\sum PSCS}*100 \tag{4.3}$$

Where:

- PRCD: Packets Received by Cluster Destination
- PSCS: Packets Sent by Cluster Source

Graphs show the fraction of data packets that are successfully delivered during simulations time versus the number of nodes is presented in chapter 5.

4.2.5: End-to-End Delay calculation in Ad-hoc networks

For this project as stated in the chapter 2, there is a moving of packets between nodes to nodes and between clusters to clusters. Because of this movement, delays can happen, that is why the researcher took in consideration of the End-to-End delay which is the average time delay for data packets to reach from the source node to the destination node. It includes processing, queuing and propagation delay of the link. The performance is better when packet end-to-end delay is low. The detailed diagrams are shown in chapter 4 of this research thesis.

$$\text{EED}=\frac{1}{N}\sum_{n=1}^{N}(\text{TSn - TRn}) \qquad (4.4)$$

Where

- TRn : Time at which data packets n has been sent
- TSn : Time at which data packets n has been received
- **N** : Total number of data packets received

4.2.6: Throughput Calculation based on cluster based routing

For the cluster based routing, the parameter Throughput has been taken into consideration which is the total packets successfully delivered to individual destinations over total time.

$$\text{Throughput}=\frac{\sum P}{T} \qquad (4.5)$$

Where:

- **P**: Number of packets successfully received at the cluster Destination
- **T** : Unit time

4.3: Proposed System and Algorithm Design

4.3.1: Flowcharts Design

In this project, three parameters were proposed to test but only two have been taken in place which are the cluster density and fault tolerance. The following diagrams indicate the flow of information from the nodes creation to fault tolerance.

4.3.1.1: General System Design Flowchart

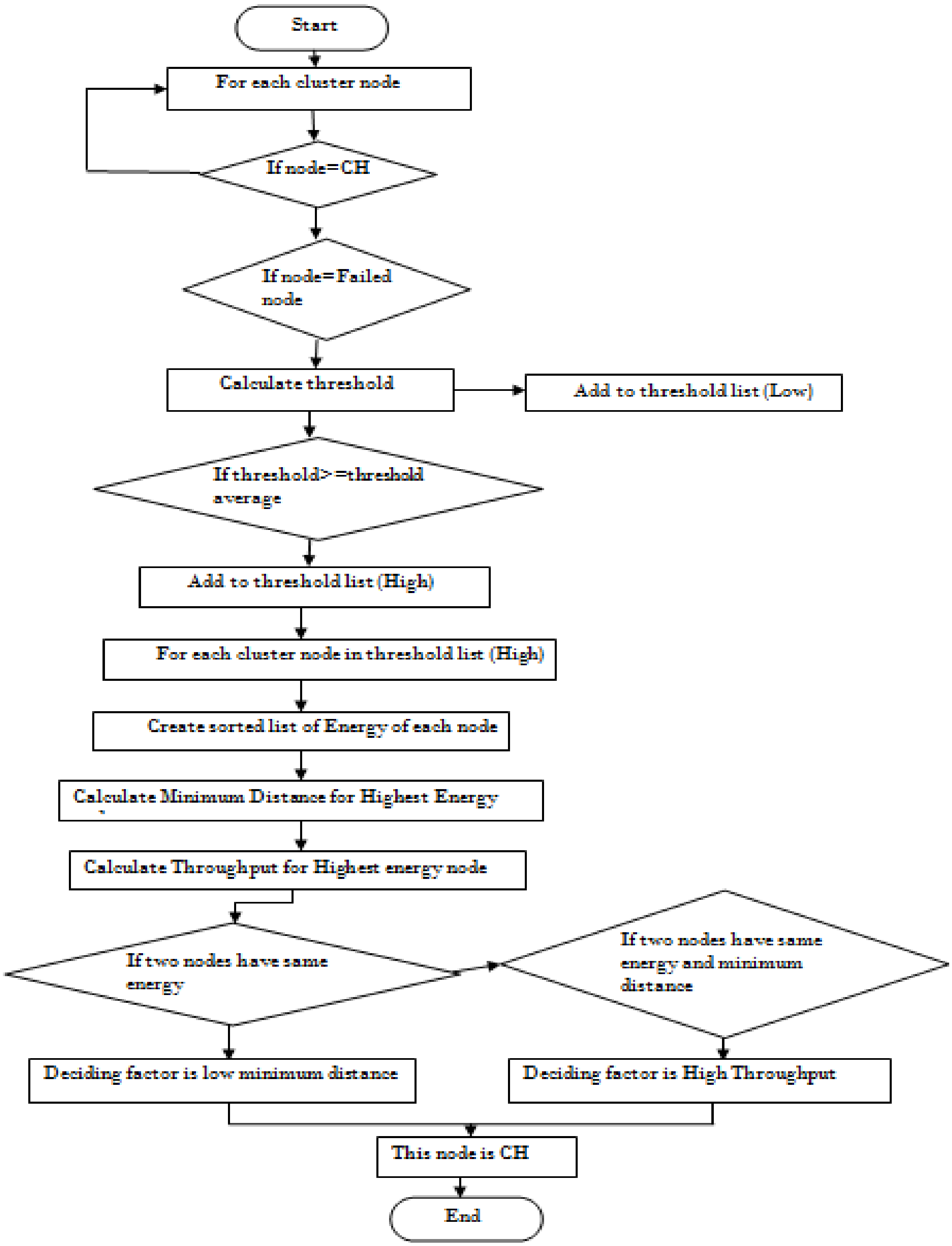

Figure 4.3: System flowchart design

4.3.1.2: Proposed design (Scenario) for Fault-tolerance parameter

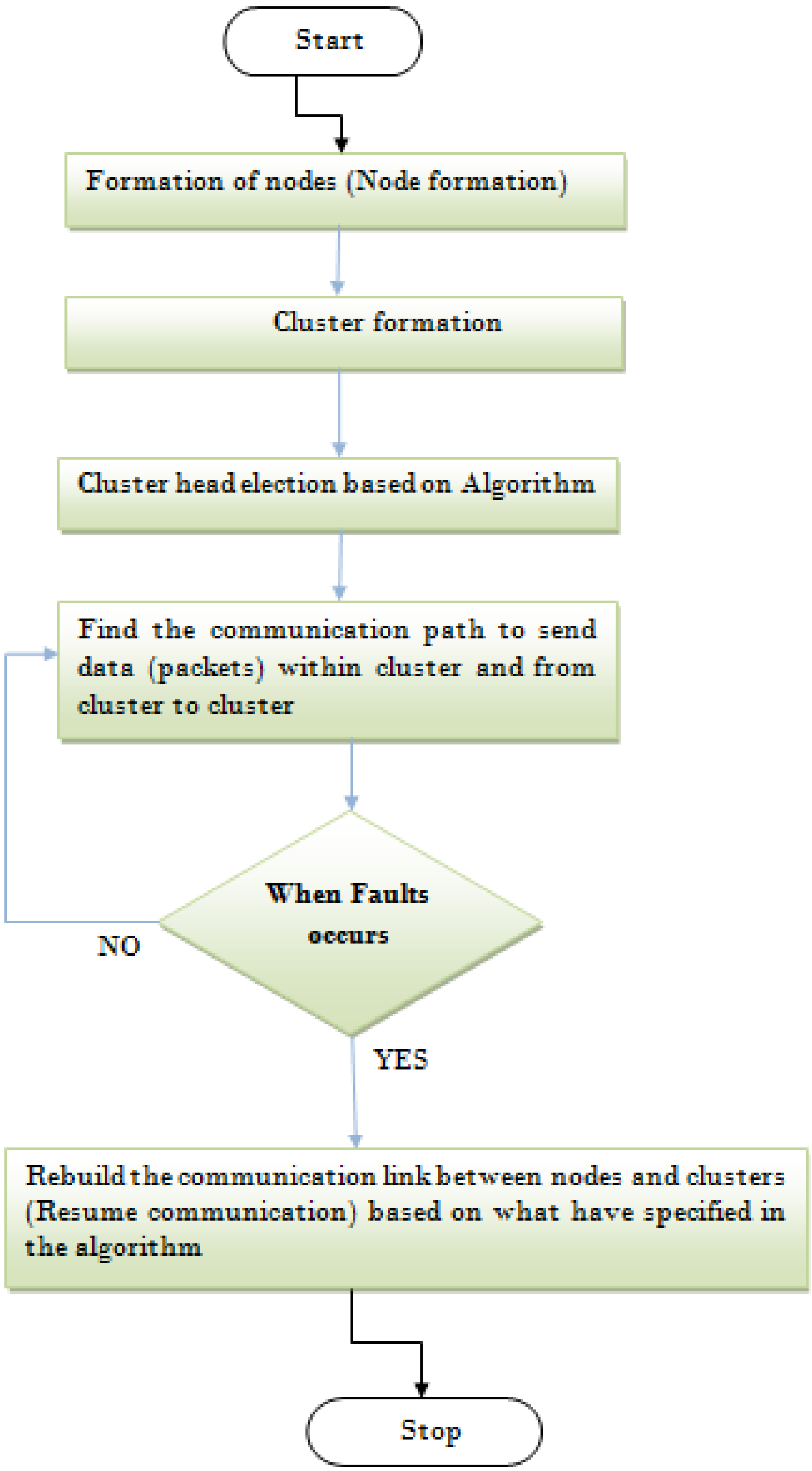

Figure 4.4: Proposed flowchart for fault tolerance

4.3.1.3: Cluster Head selection Flowchart

In the Ad-hoc networks, after forming the clusters, cluster head for each cluster is selected where the average energy of the network is calculated; the nodes having energy more than average energy of network are added to a set of candidate cluster heads. Distance between each candidate cluster head and center is calculated. Candidate cluster head having lowest distance from the center is selected as cluster head for this cluster and remaining nodes from the candidate cluster head set becomes normal nodes. The following figure 4.6 shows the process of electing the CH.

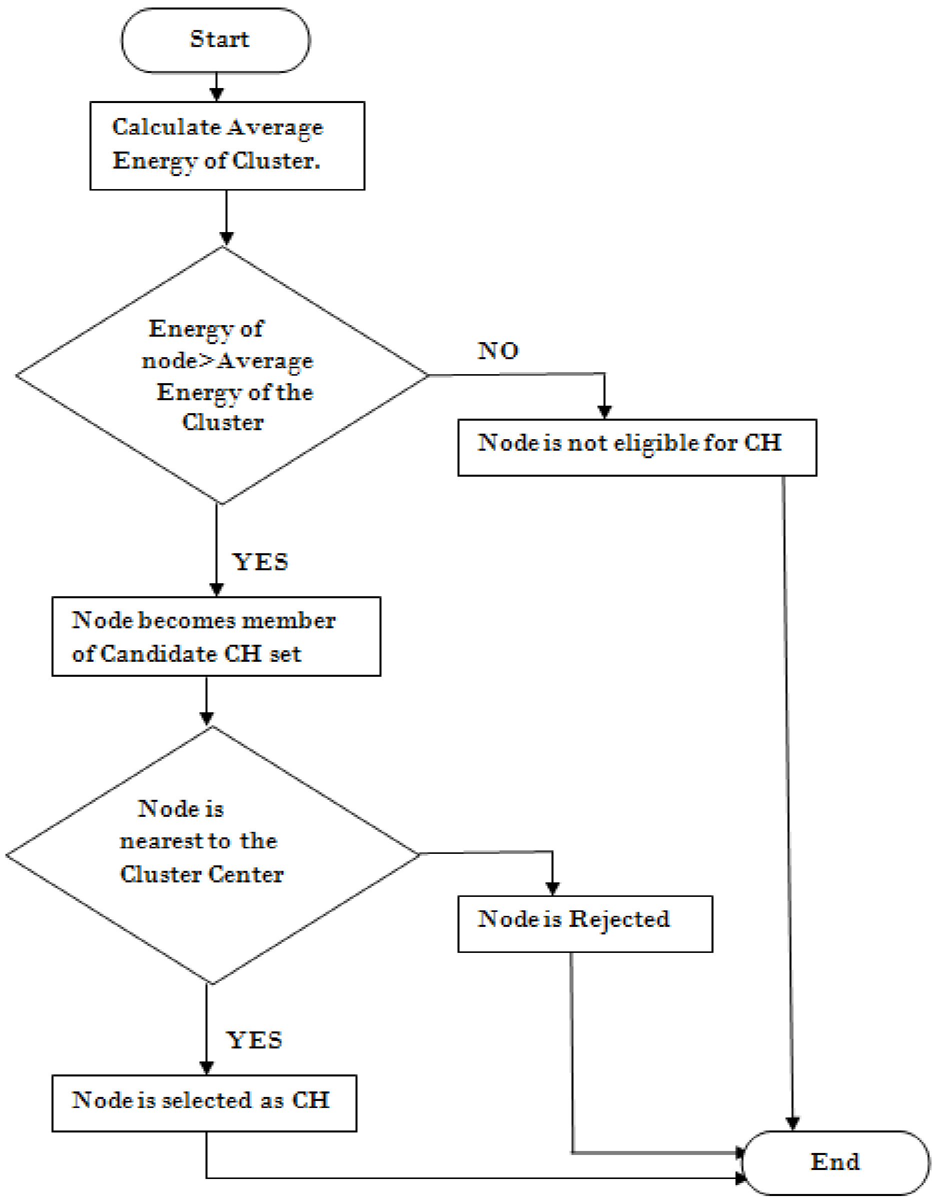

Figure 4.5: CH election flowchart

4.3.2: Algorithms Design of the system

a) Cluster Creation Process

Nodes are grouped together to form a cluster. The Algorithm 1 finds the position of each node and identifies the speed of each node. Cluster size will become smaller, when the average speed of the node is slow. Similarly, whenever the average speed of the nodes is high, the algorithm creates a bigger size cluster.

```
Cluster (N)
{
N-Number of Nodes
Set  the position of N;
Calculate the speed of each node
from N;
If(Average speed of each node is slow)
Form the cluster size smaller;
Else Make the cluster size bigger;
}
```

b) General Algorithm of the system

The Algorithm consists of four stages:

i. *Geographical formation of cluster.*
ii. *Selection of cluster heads in each cluster formed.*
iii. *Data aggregation phase which involves the gathering of collected data by the cluster head from the sensor nodes within its cluster.*
iv. *The initial Energy E_{in} (n) of node is measured*

- ✓ Also, the distance d(n) from each node to the base station or to the corresponding higher level cluster head is measured.
- ✓ Estimation of the energy required by each node for transmission within the cluster not to BS or to higher level CH for two and three cluster formation within a cluster is carried out using the formula: **(E_{amp}*k*d_2).** *(4.6)*
- ✓ The maximum energy after the subsequent transmission round for each node is estimated and selection of CH is done using the formula:

$$\max(E_{in}(n)-E_{amp}*k*d_2) \qquad (4.7)$$

then after the CH selection is carried out, the next cluster head selection will take place after the current round is completed.

Step 1: Network Initialization

- Base Station broadcasts a low cost control messages for header selection to all nodes.
- All nodes send location and energy information to Base Station.

Step 2: Cluster Head- Set Selection

- BS Selects a node with the greater remaining energy to become the first header for Cluster Head Set.
- Header Sends the Advertisement.
- Other Nodes reply to the header with **Ack** (Acknowledgement)
- Three nodes with Maximum Energy are selected as a head set member.

Step 3: Path Chain Formation & Leader Selection

- End Cluster active head sends the TOKEN to Next Cluster
- Leader sends the TOKEN to Base Station
- Base Station broadcasts the 'Chain completion' message.

Step 4: Data Transfer

- Member nodes of each cluster send data to Active Cluster Head.
- Active Cluster Heads collected the data.
- Active Cluster Head send the collected data to the leader through the chain.
- Leader node sends the final gathered data to Base Station.

Step 5: Changing Active Header

- If E of Active Cluster Head < E^{th}, the Head set member with Maximum energy becomes a new header
- If E of the three members is less than E^{th}, go to step 2.

Step 6: Check for the Fault notations in the Cluster

a. Within the Cluster communication
b. Communication with other
c. Checking packet dropping.

- **Fault tolerance (FT)** is the ability of a system to continue its operation after the failure of one of the nodes.
- When Faults happens in the system, choose the backup route or nodes in the system to avoid the system failure and maintain the system functionality (Availability).

1. **Continuous Availability (CA)** means that the user can utilize the service without experiencing any timeouts. It doesn't matter how long it has been since the node failed.
2. **High Availability (HA)** means that a user might experience service timeouts in case if one of the nodes goes down; however, the system will be recovered automatically with minimum downtime.

→ CA cluster is a Continuous Availability cluster.

→ HA cluster is a High Availability cluster.

Step 7: Choosing the backup nodes and,

Step 8: Choosing the back routes. In the first step, the proposed algorithm uses the nodes with the same movement path as the backup nodes for the main nodes.

- Energy dissipated per bit at transmitter- $E_{Tx\text{-}elect}$
- Energy dissipated per bit at receiver- $E_{Rx\text{-}elect}$
- Amplification factor- E_{amp}
- Cost of circuit energy when transmitting or receiving one bit of data- E_{elect}
- Amplification coefficient K- a number of transmitted data bits
- d- Distance between a sensor node and its respective cluster head or between a CH to another cluster nearer to the BS or between CH and BS.

<u>Density-based Algorithm.</u>

Density= number of points (Nodes) within a specific radius (Eps) or area.

- ✓ A point is a core point if it has more than a specific number of points (MinPts) within Eps. These are points that are at the interior of a cluster.

- ✓ A border point has fewer than MinPts within Eps, but in the neighborhood of a core point.
- ✓ A noise point is any point that is not a core-point or a border point.

Step D1: Create a graph whose nodes are the points to be clustered.

Step D2: For each core-point c,create an edge from c to every point p in the □-neighborhood of c

Step D3: Set N to the nodes of the graph;

If N does not contain any core points terminate

Step D4: Pick a core point c in N

Step D5: Let X be the set of nodes that can be reached from c by going forward; create a cluster containing X□{C}

N=N/ (X□{C}) *(4.8)*

Continue with Step D4

1) Expected Distribution condition NNDistSet(C) which is set of nearest neighbor of cluster C has the expected distribution with required confidence level.
2) Maximality Condition**:** Every point that comes into neighboring of C doesn't fulfill condition (1).
3) Connectivity Condition: Each pair (a,b) are connected through cluster structure. Algorithm makes set of candidates using region query if distance set of C has expected distribution then point will remain in cluster. Otherwise insert point in list of unsuccessful candidates. In the same way expand cluster and check condition. Now list of unsuccessful candidate is again checked via condition. If passes then put in cluster otherwise remain in that list.

Secure and Fault Tolerant Dynamic Clustering Algorithm

In step 2 of selecting a CH, the eligible node for the CH should have:

1. Threshold value above average of all the participating nodes.
2. Highest throughput.
3. Minimum distance from the sink node.
4. It should not have failed in the last N transactions.

Input:

Nodes= {N_1, N_2, N_3,.................... N_n}

Clusters= {C_1, C_2, C_3,................................. C_n}

Output:

- ✓ A Cluster Head (CH) in each cluster with Threshold Th above Average Threshold (Th_{avg}),
- ✓ Highest Availability Energy E_{avail},
- ✓ Minimum distance from the SINK MinDistNode and
- ✓ High Throughput (TP).

CH: Cluster Head If for a nodeTh>=(Th_{avg})&&E_{avail} is maximum&&Distance is low&&TP is High, then node=CH.

Steps of Proposed Algorithm:

Phase 1: Determine the eligible nodes by using step1

Phase 2: Determine the Threshold energy of the nodes by using step2.

Phase 3: Determine the Available Energy in the node using step3

Phase 4: Calculate the Maximum Distance and Throughput using step4.

Phase 5: Check the link/node failure

Step FT1: Finding the eligible node. For finding the eligible nodes we focus on the nodes past record of not being a cluster head already and not failed for the last n operations.

For all the nodes in a cluster if node=already (CH) go to next node;

Check the node status in memory table

if nodestatus=fail(trans id) then go next node

else node can be selected

move to step 3

refresh

check for node availability for n sec

if NodeAvailability Continue

else select the next Cluster Head

Step FT2: Calculating the Average Threshold (Th_{avg}) of the network. Threshold value is the value for which the node will survive in the network. Before electing any node as CH, we take into account its threshold value.

for each node in cluster C_1........C_n

Calculate threshold

$$Th(n)=\left[\frac{p}{1-p[r \bmod(\frac{1}{p})]}\right] \quad \text{if n belongs to G} \qquad (4.9)$$

$$Th(n)=0 \quad \text{if n doesn't belong to G} \qquad (4.10)$$

Select all the nodes having the values above the threshold value by calculating the average threshold

$$Th(avg)=\frac{\sum Th(n)}{n} \qquad (4.11)$$

If Th(n)>=Th(avg)

Put in (ListHigh)

else Put in (ListLow)

In the Equation (4.9), p represents percentage (probability) of cluster heads (e.g: p=0.1), r represents the current round of iteration, G is the set of nodes that have not been cluster heads in the last $\frac{1}{p}$ rounds and n is the random number between 0 and 1.

Step FT3: Calculating the Available Energy in the node.

For all the nodes in the ListHigh

Energy=Power*Time

Calculate the available energy of each node

Eavail-Current Enery/Maximum Enery

Create a List Eavailsort in descending order.

Step FT4: Calculating Minimum Distance and Throughput.

For the node with the highest energy, calculate its minimum distance and throughput value. If two nodes have same energy, node with the minimum distance is taken as the CH and if two nodes have same energy and same distance then the node with high throughput is selected as CH.

for the first node in $E_{availsort}$

calculate minimum distance $d=[(x_2-x_1)^2+(y_2-y_1)^2]^{1/2}$, *(4.12)*

Throughput (TP)=Size of the Packet /transmission time *(4.13)*

Node-CH, if 2 nodes have same energy

Node with min dist=CH

if 2 nodes have same energy and same minimum distance

Node with max throughput=CH

Step FT5: Check the link/node failure

```
for the transmission
check the link/node failure
if (node/link failure)
{
Backup node/link to resume communication;
}
else {
System is down}
end
```

4.4: Conclusion

In this chapter, it covers the analysis of the existing system, the requirement of new system and it also describes the proposal of the new system, hence leading to the design of the new system. In this chapter, several formulas packet delivery ratio, end-to-end delay, energy of nodes and throughput in Ad-hoc networks were derived and explained in details. Different system flowcharts were also proposed and designed.

CHAPTER FIVE: RESULTS AND DISCUSSION

5.1: Introduction

In this chapter, I test the performance of the derived models for multiple nodes and cluster in Ad-hoc Networks. In particular, I analyze the Packet loss during the communication within nodes and clusters, the Fault tolerance performance report, the cluster density, the throughput, network life time, fault tolerance Time taken for cluster, the fault tolerant cluster efficiency performance and the cluster time taken. I further analyze the variation of end-to-end delay with number of nodes and cluster. In addition, we also investigate the effect of increasing number of nodes on the system.

5.2: Simulation results

Simulator is designed and implemented in NS-2 in order to investigate the proposed algorithm and test different parameters. As defined in [14], the network lifetime is denoted by the network stability period, which is the time spans from the initialization of a network to the instant when the first node dies. The network throughput is measured by the number of packets delivered to the sink.

5.2.1: Simulation Setup (Parameters)

In my simulation, nodes are randomly deployed in a circular area, and the radius of the circle varies from 2.5cm to 5cm. The number of nodes varies from 33 to 50. The initial energy of normal node is 35 J. The data packet size is 500 bit. In default, the network is divided into three clusters of 33 nodes each with 11 nodes. Table 5.1 summarizes the simulation parameters and their default values.

Table 5.1: Simulation Parameters

Parameter	Value
Radius of the circle *R*	2.3cm~5cm
Number of nodes *N*	35
Packet size *L*	500 bit

Initial energy $E0$	35 J
Simulation Time	2.0ms
Channel Type	Wireless Channel
Network Interface Type	Physical
X Dimension of topography	2000
Y Dimension of topography	2000
Time of simulation end	10.0ms
Bandwidth	10Mbits/sec
Delay	10ms
Scenario Size	200cm x 200cm

5.2.2: Simulation Results and Analysis

Figure 5.1 shows the simulation environment where the nodes are formed and they are grouped into cluster as shown here. Each node is given a number in each cluster to be differentiated with other in the whole Ad-hoc networks. The nodes are numbered from 0 to 34. It also shows the packets dropped from the network.

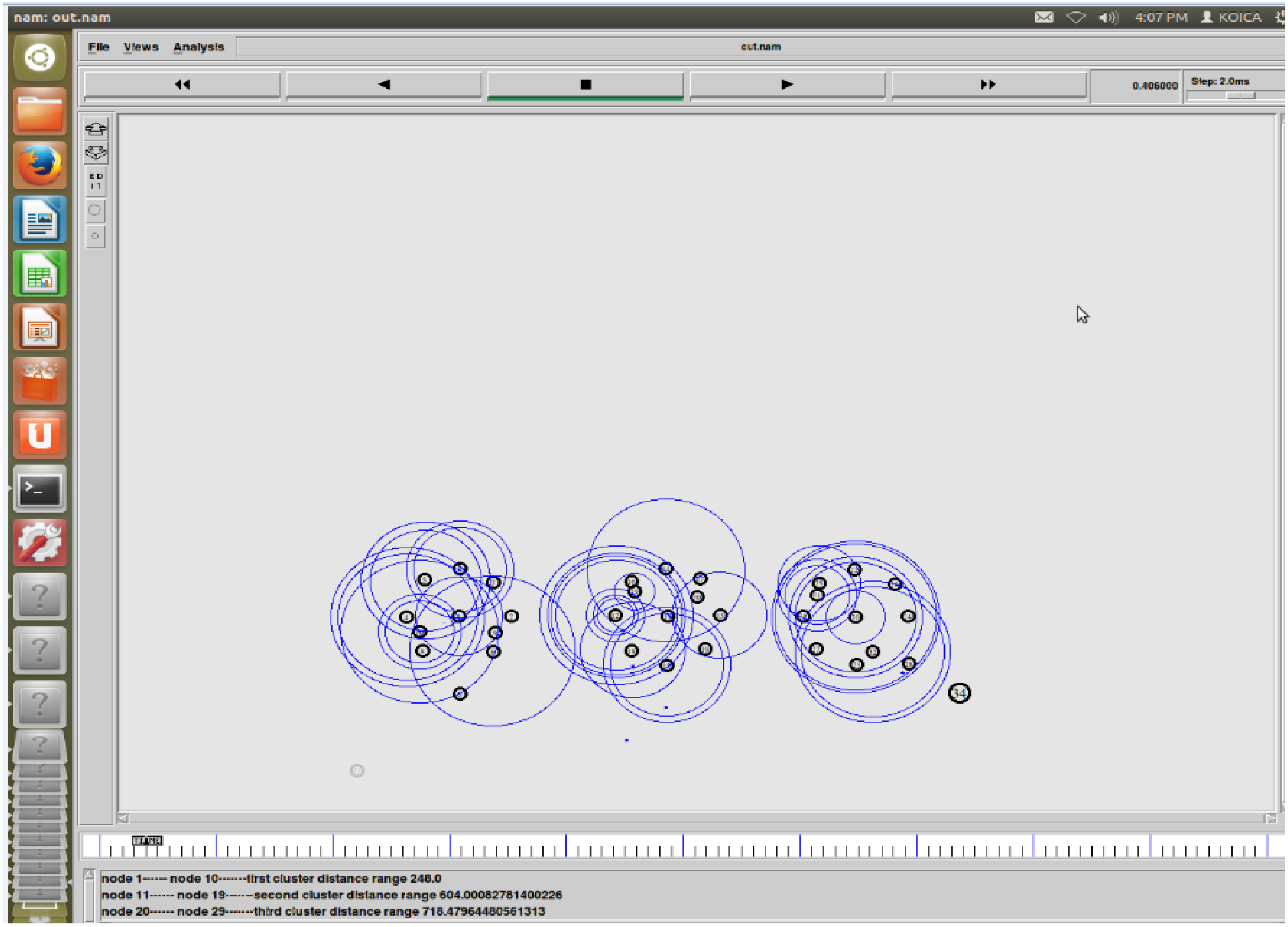

Figure 5.1: Nodes and Cluster Formation plus the dropped packets

Figure 5.2 indicates the CH election in each cluster based on the Energy consumption for each node. Based on the Algorithm in chapter 4, where the Estimation of the energy required by each node for transmission within the cluster not to BS or to higher level CH for two and three cluster formation within a cluster is carried out using the formula: **(E_{amp}*k*d_2).** The maximum energy after the subsequent transmission round for each node is estimated and selection of CH is done using the formula: **max(E_{in}(n)-E_{amp}*k*d_2),** then after the CH is the node with High energy.

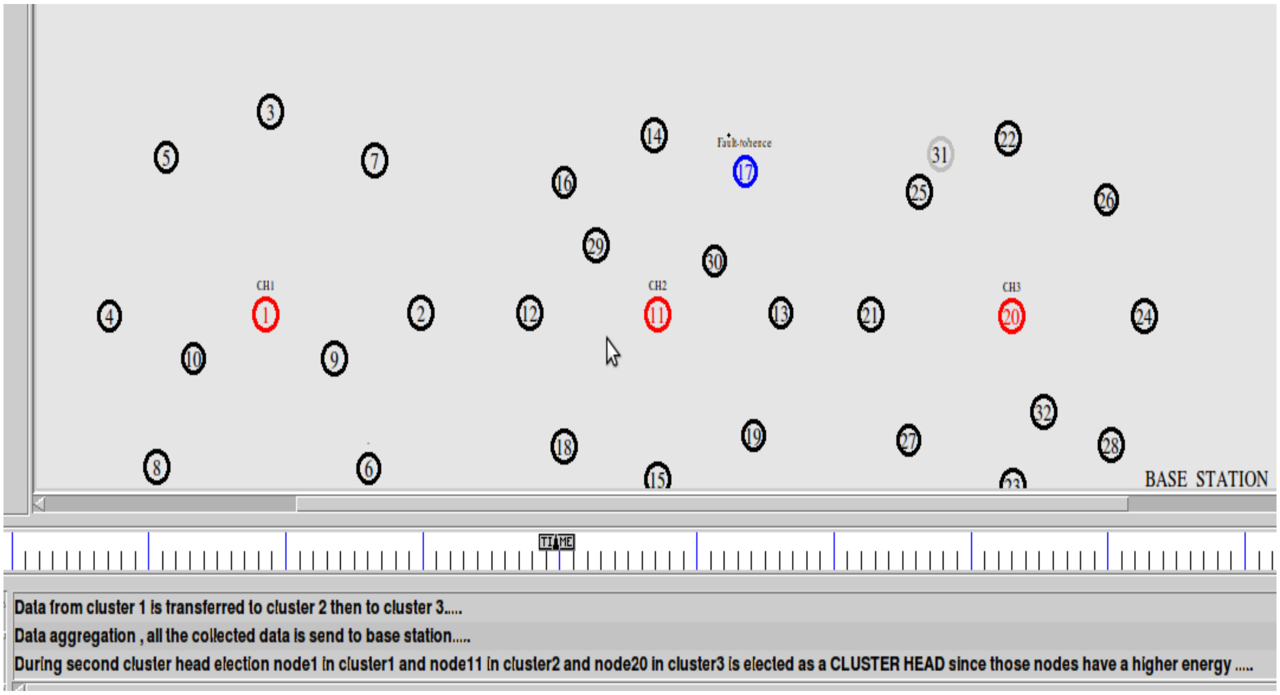

Figure 5.2: CH election scenario

Initially, node 9 in the first cluster, node 30 in the second cluster and node 32 act as cluster head. The first iteration of data transfer is done for all clusters. All CH send the advertisement message to its member nodes. In this figure, in the second iteration, we have 3 clusters where in the first cluster , the node 1, node 11 in the second cluster and node 20 in the third cluster act as Cluster Head since those nodes have a higher energy. After first iteration comes the second iteration and all the procedures are followed as in the first iteration.

In the figure 5.3, there is a process of Fault tolerance (FT) which is the ability of a system to continue its operation after the failure of one of the nodes. When Faults happens in the system, choose the backup route or nodes in the system to avoid the system failure and maintain the system functionality (Availability). As indicated on the figure, at the node 6 in the first cluster, there is a failure but the there is a backup route where the data which was supposed to pass through that node tried to find another route to pass through.

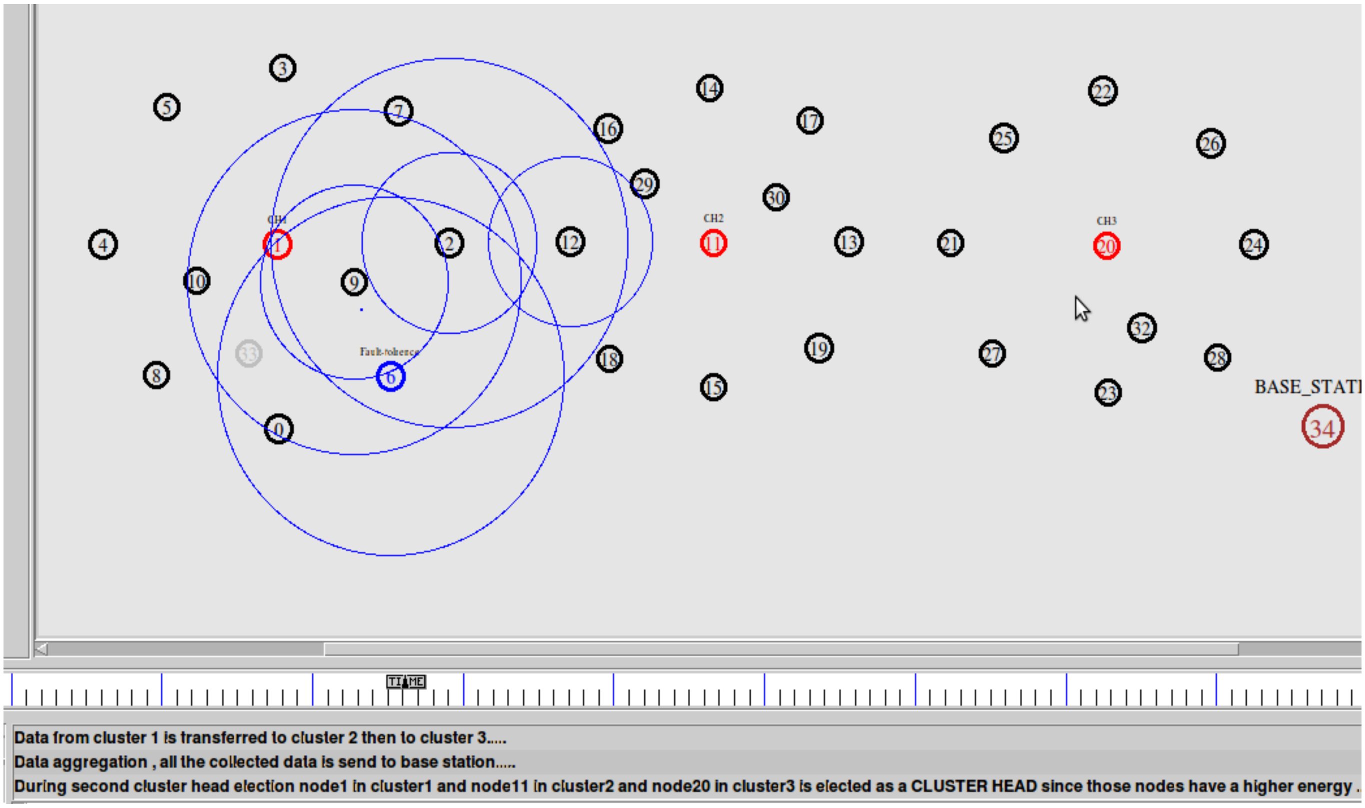

Figure 5.3: Node failure in cluster 1(Node 6)

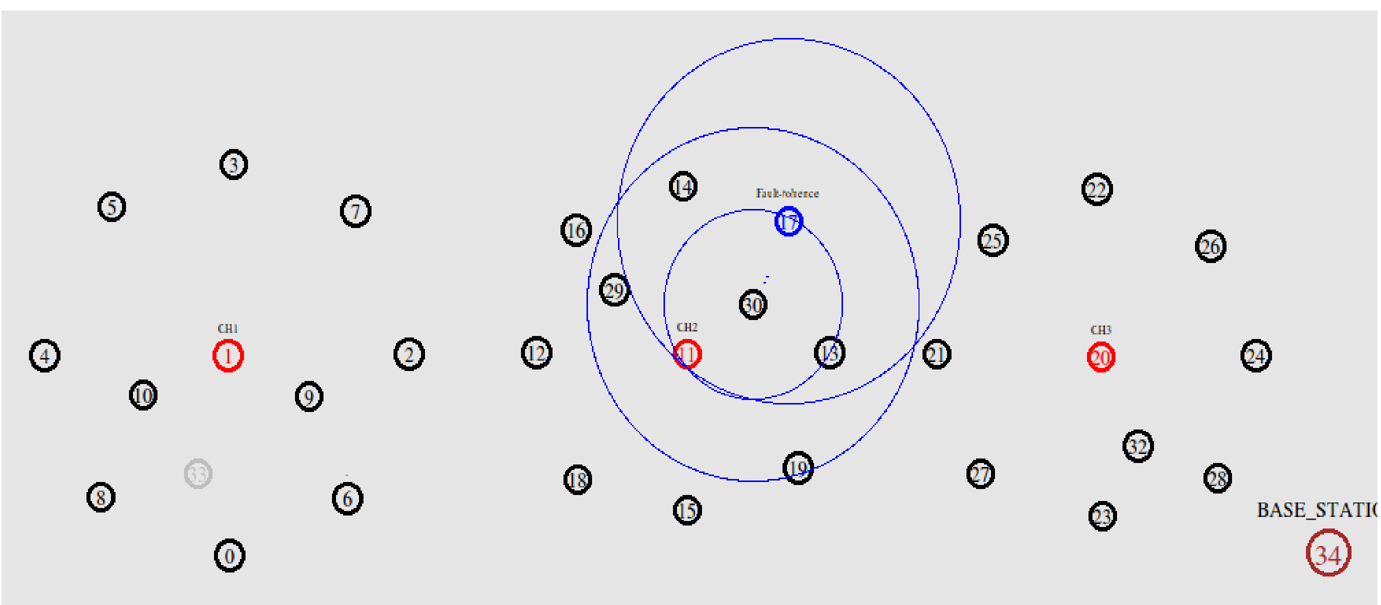

Figure 5.4: Node failure in cluster 2 (Node 17)

As indicated on figure 5.4, the fault tolerance process also happens in cluster 2 at the node 17 where it has failed to send the packets to its neighbors. Due to that failure, the packets tried to search for a backup route as shown where the communication continued to node 11, node 13 and node 19.

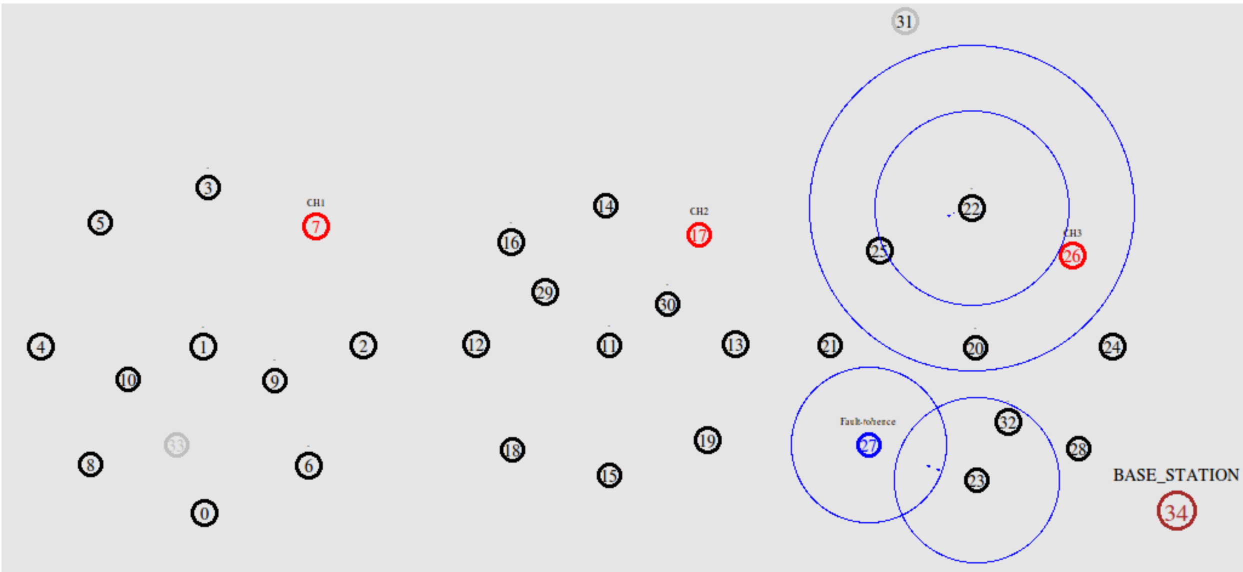

Figure 5.5: Node failure in cluster 3 (Node 27)

In the figure 5.5, the failure also took place at the node 27 in cluster 3, where also there is a backup as shown. The packets which were intended to pass at that node passed at the node 23 and node 32.

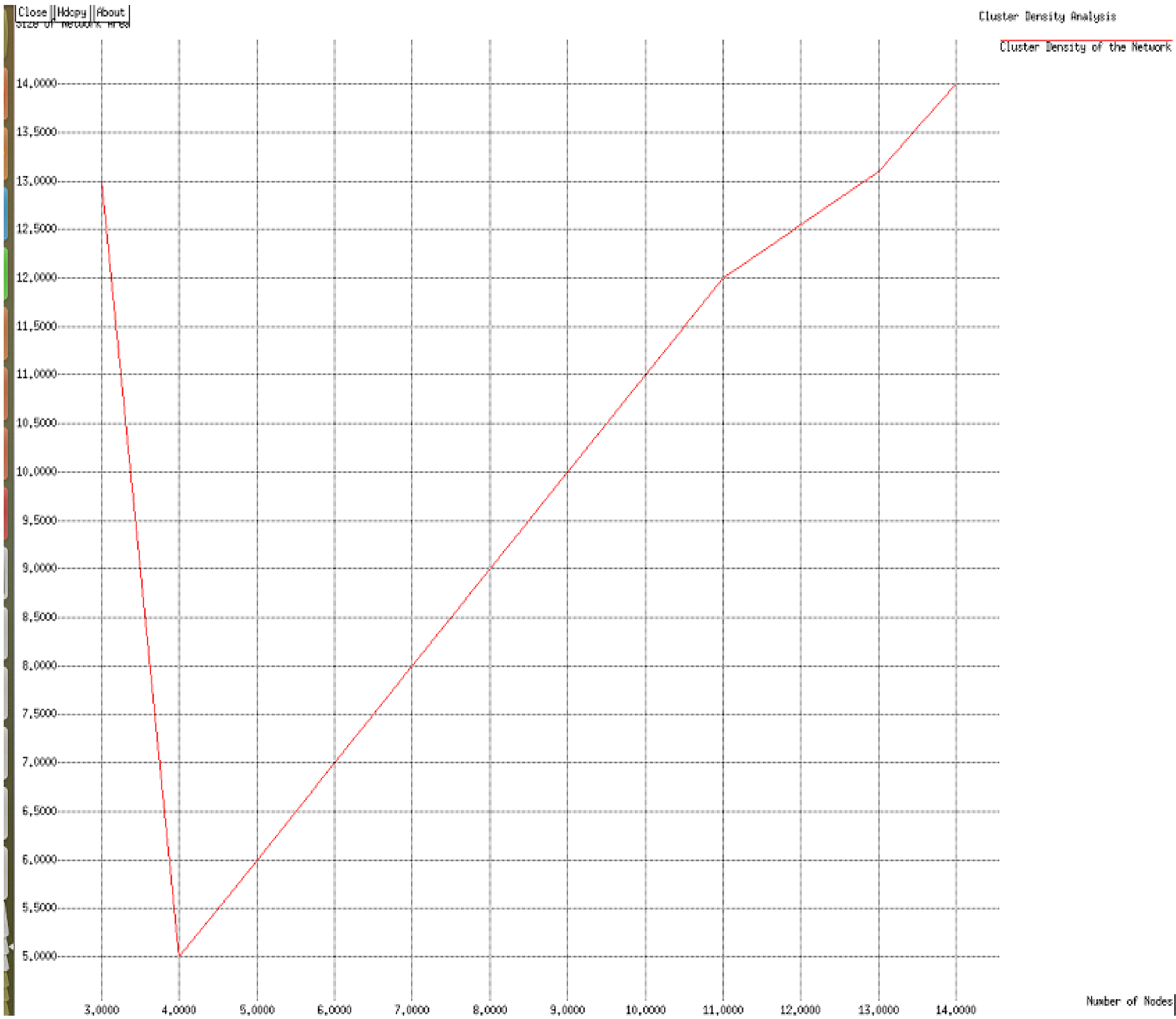

Figure 5.6: Cluster Density Analysis

As explained in the formula (2.1), where the cluster density for an entire network is the number of nodes within a given region of an Ad-hoc network. The figure 5.6 shows the system performance against the parameter Density. The new algorithm indicates that when the size of the area increases while the number of nodes remains constant, the cluster density decreases while it increases when the area becomes big at the same time the number of nodes increases.

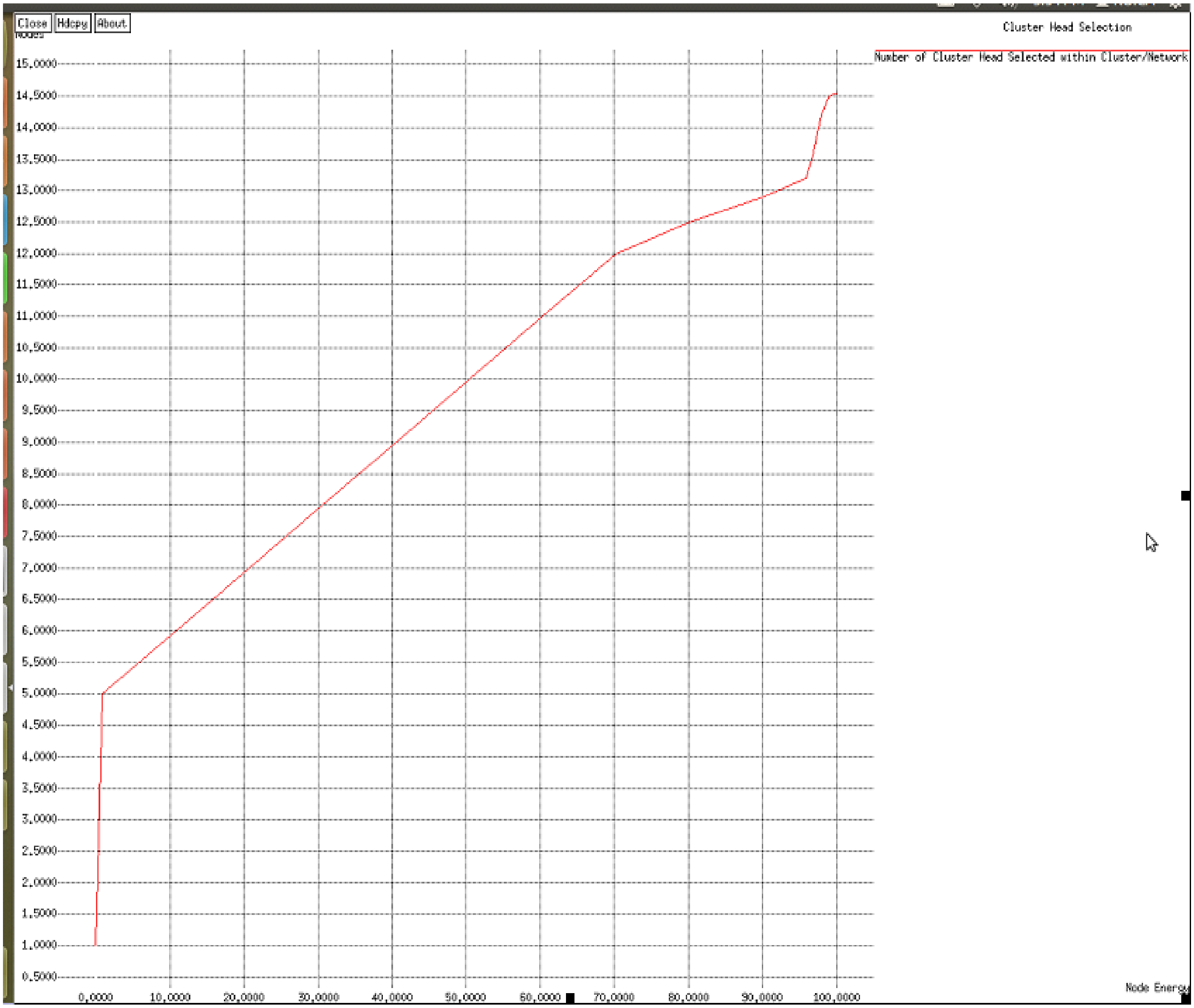

Figure 5.7: CH Selection graph

Figure 5.7 indicates the CH selection graph where the node with high energy is selected as CH as explained in the Figure 5.2. As shown on this figure, the initial node energy was 0.0000 J and there was not CH elected but as long as the Energy of the node increases (many nodes which have the higher energy), many CHs are selected.

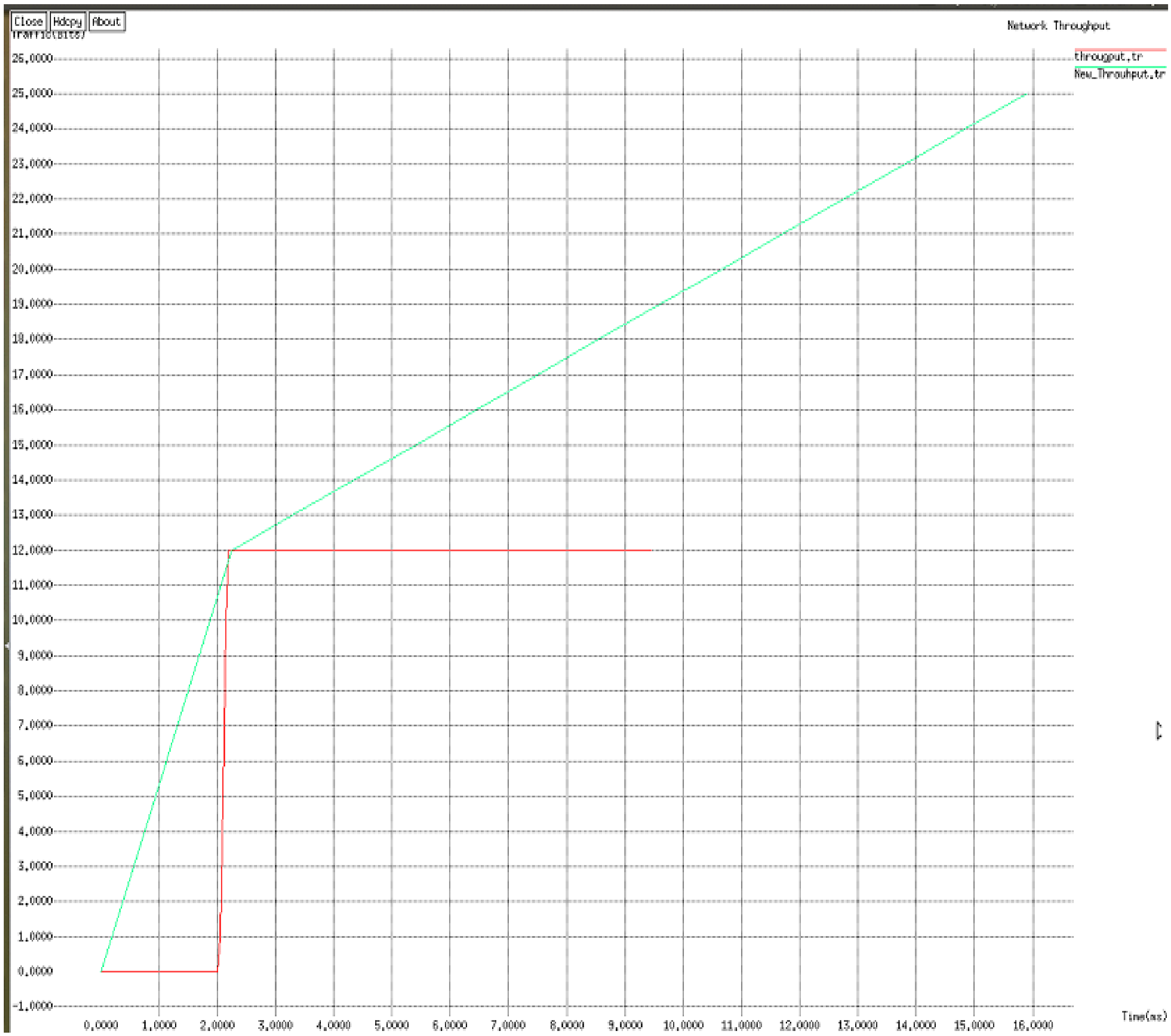

Figure 5.8: Network Throughput

As explained on the formula (4.5), the network throughput is the total packets successfully delivered to individual destinations over total time. Here results are shown in form of lines, where red line shows the Throughput of existing algorithm and green line shows the throughput of the proposed algorithm in Ad-Hoc networks. For example, for 12bits, the throughput of the existing algorithm started to be constant while for the new one it started being increased. It is shown in graph clearly that the proposed algorithm has a good throughput than the existing one in terms of time and traffics.

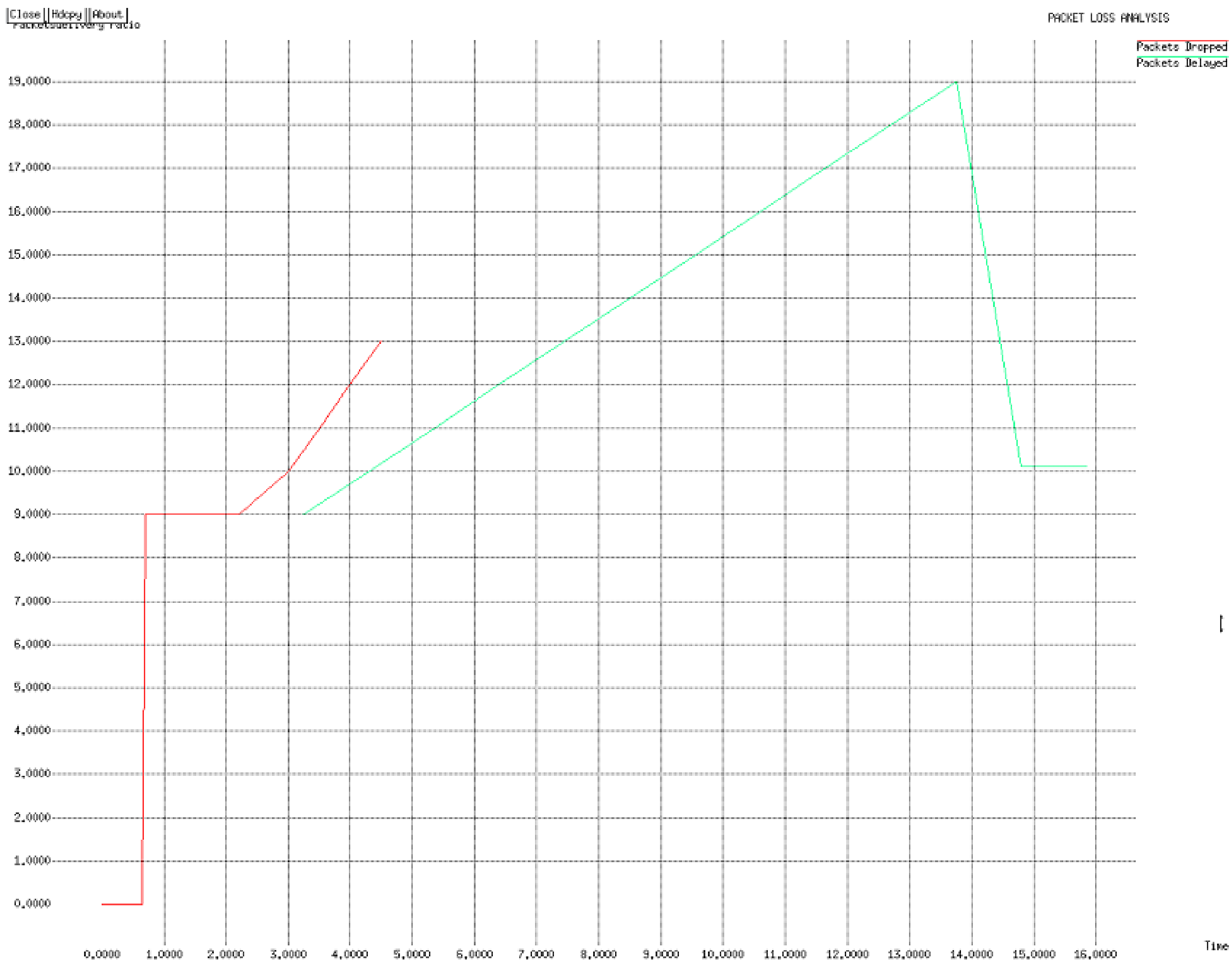

Figure 5.9: Packet Loss Analysis within Network Vs Time

From the Figure 5.9, we can see that when the time increases, many packets are lost as well as they delay to reach the destination. The red line indicates dropped packets and green line shows the delayed packets. The delayed packets are implicitly increased when the time is moving but this delay starts at certain time after the packets transmission. You can see the delay starts at 3.000ms after the starting of the simulation and when the simulation time was 0.000ms, no packet loss (PDR was 0.000).

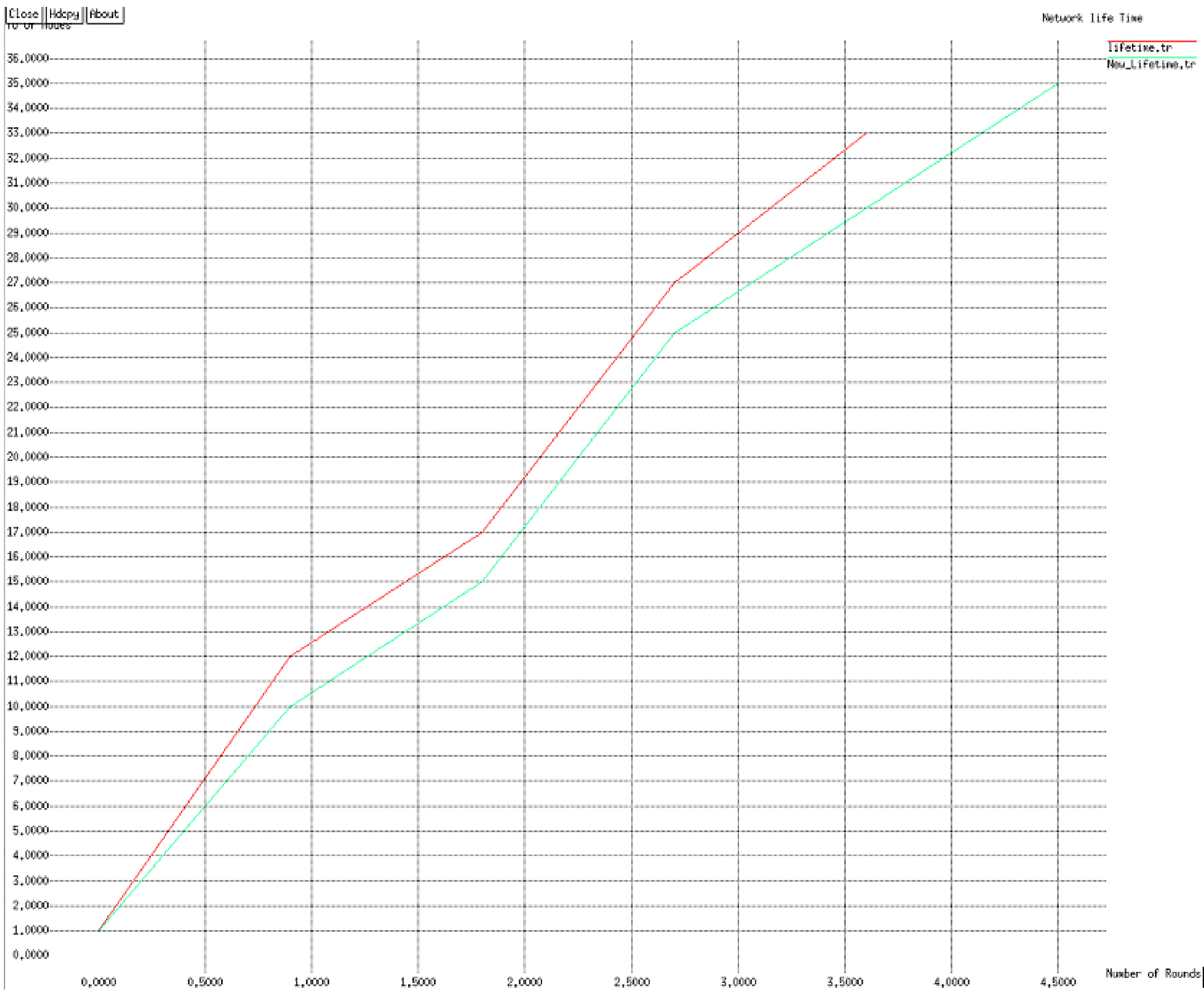

Figure 5.10: Simulation time vs. Network Lifetime

Above figure shows that existing algorithm has less network lifetime as compare to new proposed one. Here red line shows less network lifetime of existing algorithm and green line shows more network lifetime. This concluded that proposed algorithm is better than existing one.

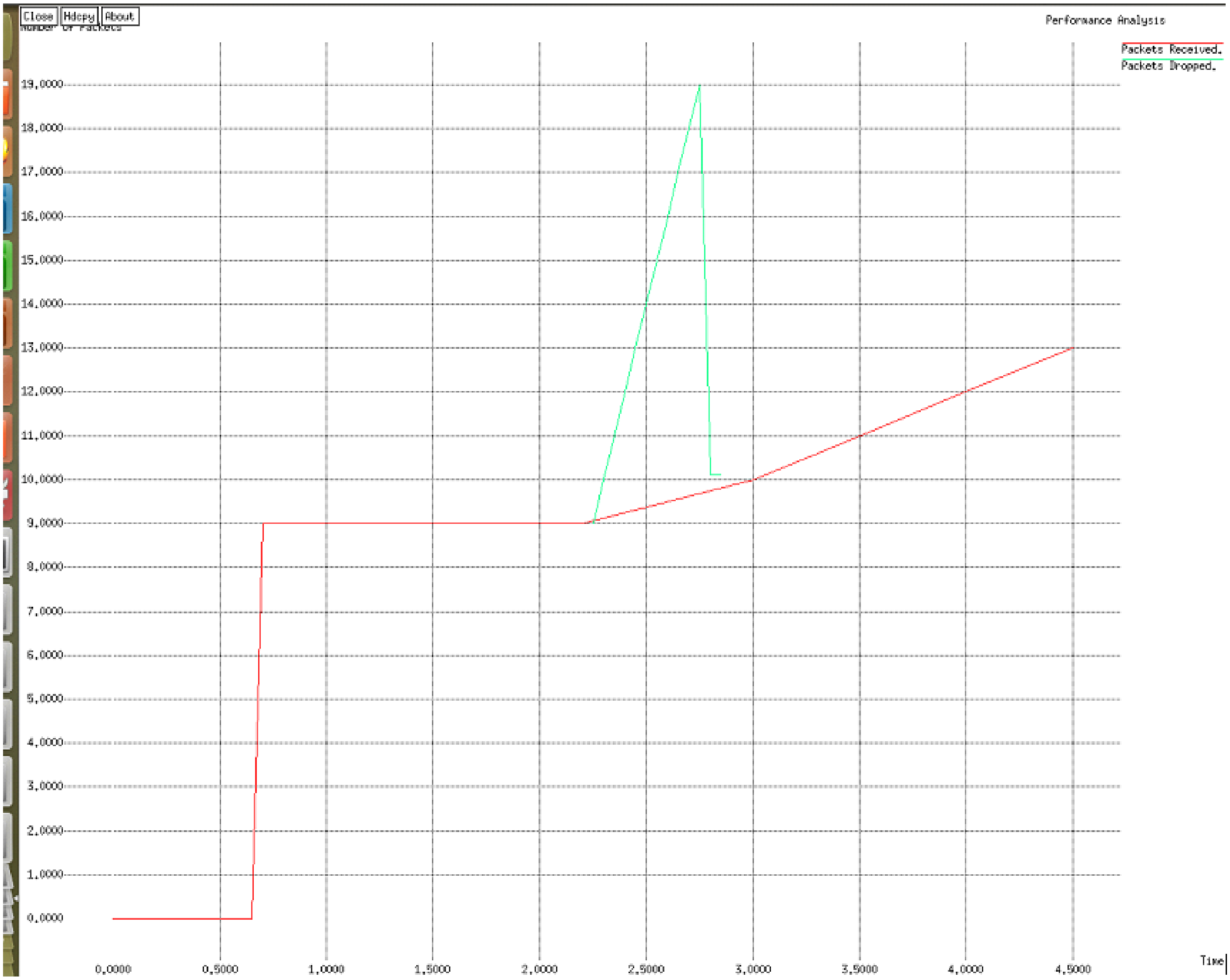

Figure 5.11: Performance Analysis

The above figure shows the network performance during the communication between nodes to nodes and within cluster to cluster in Ad-Hoc networks. The results are shown by red lines which represent the packets received and green line representing dropped packets. As shown, from 0.0000ms to 0.5000ms, there were no received packets but as the time moves, many packets are reaching the destination. Between 2.000ms and 3.000ms there is an increase of the packets dropped. From that, we conclude that many packets are received at the destination while some of them are dropped during the transmission.

Figure 5.12: Fault Tolerance Performance Report

In the figure 5.12, we can see the fault tolerance (failure of node or link) increases with time and with clusters. At time 0.0000ms only one cluster has been formed while there were no failed nodes. The failure has been increased with time where up to 70.000ms, there a high increase of failed nodes as well as the increase of the cluster. We conclude that nodes failure becomes high when many clusters are formed within the network.

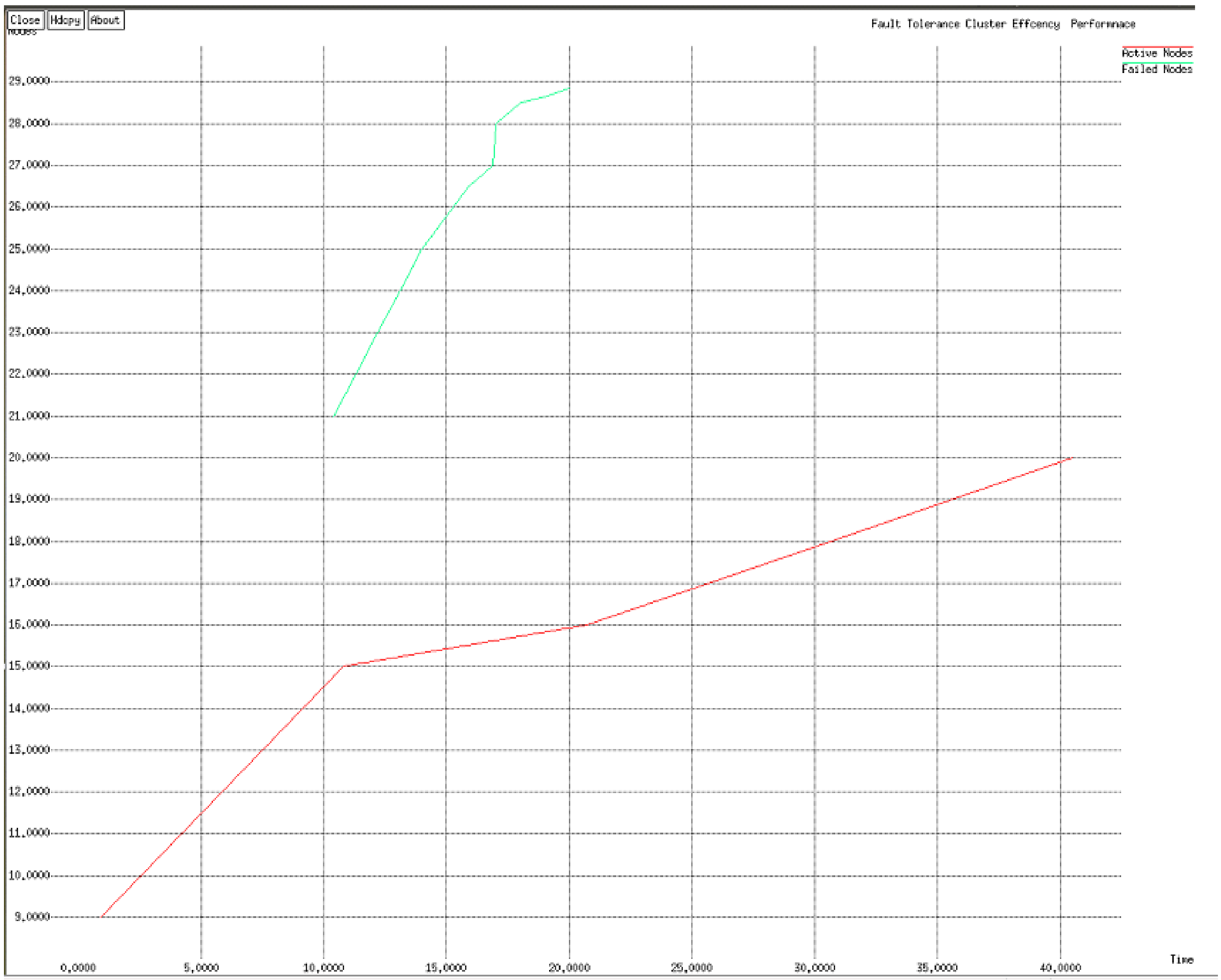

Figure 5.13: Fault Tolerance Cluster Efficiency Performance

Fig 5.13 shows the number of dead (failed) nodes with time. In this algorithm, the numbers of dead nodes are less as compared to the existing as indicated here. The failed nodes are indicated by green line and the active nodes are represented by red line. The results shown above summarized the performance of the proposed algorithm for finding better fault tolerance in Ad-hoc networks. It is indicating that as the time increases, active nodes are higher which means that during the packets transmission, small number of nodes failed to transmit as well as the backup nodes or routes is taking place to resume the communication.

5.3: Conclusion

In this chapter, the detailed information on the simulated results is given. It showed how the proposed algorithm handles the problem of fault tolerance in ad-hoc networks by considering the network which is composed of networks nodes and nodes are grouped into clusters where each cluster has a CH which initiates the packets transmission. It shows that when fault occurs, a back routes or nodes are established to resume the communication. Proposed algorithm in chapter 4 is simulated and the performance metrics was tested such us the throughput, cluster density, packet loss, network lifetime.

CHAPTER SIX: CONCLUSION AND RECOMMENDATIONS

6.1: Conclusion

In this thesis project, the Design of novel high density, fault tolerant Protocol for cluster based routing in Ad-hoc Networks is about to increase the system performance when the faults occurs within nodes or within the transmission link and to test the system performance in case we increase the network nodes in the network area or in the cluster (Cluster Density). In the proposed algorithm, I referred to other algorithms done regarding the fault tolerance in ad-hoc networks and developed the new one with several equations to increase the cluster density. In addition several existing schemes were evaluated and considered but the proposed algorithm works better than the existing one. Several network performance metrics have been tested such the network throughput, the packets loss, the cluster density, the network lifetime to see how the new developed algorithm works better than the existing algorithms that is why I proposed an energy efficient fault tolerant routing method for detecting faults occurred in ad-hoc networks. The goal was to investigate current research work on Fault tolerance and to detect the faults in Ad-hoc networks. Currently I implement the algorithm that detects the faults during data transmission and also recovered it so that communication is done without failure between the nodes. The Energy of each node is considered in order to get energy efficient path to send the data between the nodes. The proposed technique has a number of advantages. For example, it requires only the knowledge of one-hop neighbors to form clusters. Furthermore, the clustering algorithm is robust for topological change caused by node failure, node mobility, CH change and even for node insertion or removal. Simulation results show that the proposed clustering scheme gives better performance in terms of cluster size, variance of cluster size, and number of single node clusters.

6.2: Recommendations

As today, technologies are advancing, in the future work on this project, I recommend to develop a routing algorithm which handles the problem of fault tolerance and density in other network technologies such LTE or LTE-Advances where the cells act instead of having networks nodes as it is in Ad-hoc networks.

REFERENCES

[1] M. Rezaee and M. Yaghmaee, "*Cluster based Routing Protocol for Mobile Ad Hoc Networks*".

[2] C. W. Ahn, R. S. Ramakrishna, and C. G. Kang, "*Efficient Clustering-based Routing Protocol in Mobile Ad-Hoc Networks*".

[3] R. Purtoosi, H. Taheri, A. Mohammadi, F. Foroozan. "*A light-weight contention-based clustering algorithm for wireless ad hoc networks*," Computer and Information Technology (CIT '04). 2004. pp. 627-632.

[4] https://www6.ietf.org/proceedings/44/slides/manet-thesis-99mar.pdf [Visited on 28th November 2016]

[5] C. Chiang, H.-K. Wu, W. Liu, M. Gerla, "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel," Proceedings of SICON 1997.

[6] D.J.Baker and A. Ephremides, "*The architectural organization of a mobile radio network via a distributed algorithm*", IEEE Transactions on Communications COM-29, pp. 1694-1701,11(Nov.1981).

[7] P. Anju and M. Silpa: "*A Distributed Efficient and Accurate Routing Protocol to Increase the Capacity of Hybrid Wireless Networks*",M.Tech Student, Dept. of Computer Science & Engineering, SreeNarayanaGurukulam College of Engineering, Kerala, India1, 2003.

[8] K.L. Scott and N.D. Bambos, "*A fully distributed algorithm for the creation and maintenance of wireless networks of mobile users*", Technical Report, TRUCLA-ENG-95 112, Department of Electrical Engineering, University of California at Los Angeles, 1994.

[9] Michiardi, P. and Molva, R. (2004), 'Mobile Ad Hoc Network', *Wiley-IEEE Press*, ch.12: Ad Hoc Network Security, pp. 329–354.

[10] Upadhayaya, S. and Gandhi, C. (2009), Quality of Service Routing in Mobile Ad Hoc Networks Using Location anf Energy Parameters, *International Journal of Wireless & Mobile Networks (IJWMN),* Vol 1, No 2, November 2009, pp. 138-147.

[11] EssamNatsheh, "*Impact of Varying Offered Data Load on Density-Based Routing in Mobile Ad-hoc Networks",* King Faisal University Al-Hassa 31982 Saudi Arabia,June 2015.

[12] Erik Kuiper," *Node density, connectivity and the percolation threshold*", Linköping University SE-581 83 Linköping, Sweden, November 1, 2010

[13] J.-D. Abdulai, M. Ould-Khaoua, L. Mackenzie, "*Adjusted probabilistic route discovery in* *mobile ad hoc networks*", Computers and Electrical Engineering 35 (1) (2009) 168–182.

[14] Hauspie, M. and Simplot, D. and Carle, J. (2003),'Partition Detection in Mobile Ad-hoc Networks', *In Proceeding of the 2nd Mediterranean Workshop on Ad Hoc Networks (Med-Hoc-Net 2003)*, Mahdia, Tunisia.

[15] Michiardi, P. and Molva, R. (2004), 'Mobile Ad Hoc Network', *Wiley-IEEE Press*, ch.12: Ad Hoc Network Security, pp. 329–354.

[16] J.Niththiyanathan," *Cluster-Based Networks*", Helsinki University of Technology Espoo, Finland, Available at: http://www.netlab.tkk.fi/opetus/s38030/k02/Papers/04-Nithi.pdf

[17] Y. Xue and K. Nahrstedt, "Fault Tolerant Routing in Mobile Ad Hoc Networks", Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), New Orleans, Louisiania, March 2003, pp. 1174-1179.

[18] Y. Xue and K. Nahrstedt, *"Providing Fault-Tolerant Ad Hoc Routing Service in Adversarial Environments",* 2004

[19] B. J. Oommen and S. Misra, "Fault-Tolerant Routing In Adversarial Mobile Ad Hoc Networks: An Efficient Route Estimation Scheme For Non-Stationary Environments", Telecommunication Systems Journal, pp. 159-169, 2010.

[20] S. Misra, S. K. Dhurandher, M. S. Obaidat, K. Verma and P. Gupta, "A Low Overhead Fault-Tolerant Routing Algorithm for Mobile Ad-Hoc Networks Based on Ant Swarm Intelligence" Simulation Modelling Practice and Theory (Elsevier), Vol. 18, No. 5, 2010, pp. 637-649.

[21] Jipeng Zhou and Chao Xia, "A Location-Based Fault-Tolerant Routing Algorithm for Mobile Ad Hoc Networks ", WRI International Conference on Communications and Mobile Computing, Volume 2 ,Page(s) 92 – 96,Jan 2009.

[22] Yang Qin and Kong Ling Pang, " A Fault-Tolerance Cluster Head Based Routing Protocol for Ad Hoc Networks", IEEE Vehicular Technology Conference, Page(s): 2472 – 2476, May 2008.

[23] Sun.Ruozi ,Wang.Yue, Yuan.Jian, Shan.Xiuming and Ren.Yong, "Topology control algorithm using fault-tolerant 1-spanner for wireless ad hoc networks Tsinghua Science and Technology", Volume: 17, Issue: 2, Page(s): 186 -193 , April 2012.

[24] Larry C. Llewellyn, Kenneth M. Hopkinson and Scott R. Graham, " Distributed Fault-Tolerant Quality of Wireless Networks" IEEE Transactions On Mobile Computing, VoL. 10, No. 2, February 2011.

[25] M. Nazeeruddin, G. Parr and B. Scotney, "Fault - Tolerant Dynamic Host Auto-configuration Protocol for Heterogeneous MANET", 14th IST Mobile & Wireless Communication, 2005.

[26] R. Melamed, I. Keidar and Y. Bare, "Octopus: a fault-tolerant and efficient ad-hoc routing protocol ", 24th IEEE Symposium on Reliable Distributed Systems, Page(s) 39 - 49 ,OCT 2005.

[27] M. Khazaei and R. Berangi, "A multi-path routing protocol with fault tolerance in mobile ad hoc networks " , 14th International CSI Computer Conference, Page(s): 77 – 82, Oct. 2009.

[28] S. Chandrasekaran, S. Udhayakumar, U. Mohan Bharathy,and D. Jitendra Kumar Jain, " Trusted Fault Tolerant Model of MANET with Data Recovery ",4th IEEE International Conference on Intelligent Networks and Intelligent Systems(ICINIS), Page(s):21-24, 2011.

[29] Rana E. Ahmed, "A Fault-Tolerant Routing Protocol for Mobile Ad Hoc Networks", Journal of Advances in Information Technology, VOL. 2, no.2, May 2011.

[30] R.S. Shaji, Dr.R.S. Rajesh and B. Ramakrishnan, "A Fault-tolerant Scheme for Routing Path Re-establishment for reliable communication in Heterogeneous Networks", International Journal of Scientific & Engineering Research, Volume 1, Issue 3, ISSN 2229-5518 , December-2010.

[31] R.S. Shaji, Dr.R.S. Rajesh and B. Ramakrishnan, "SFUSP: A Fault-tolerant Routing Scheme for path establishment among Mobile Devices in Pervasive Spaces", Journal of Computing, Volume-2, Issue 11, ISSN 2151 – 9617, Nov.2010.

[32] Weigang Wu, Jiannong Cao and JinYang , "A fault tolerant mutual exclusion algorithm for mobile ad hoc networks", Journal of Pervasive and Mobile Computing , Volume 4, Issue 1,Pages 139-160, Elsevier Science Publishers , February 2008.

[33] Yiyuan Gong and A.S. Fukunaga, "Fault tolerance in distributed genetic algorithms with tree topologies", IEEE Congress on Evolutionary Computation,Page(s): 968 – 975, May 2009.

[34] S. Chandrasekaran, Shanmugam, Udhayakumar, U. Mohan Bharathy and D. Jitendra Kumar Jain, "Trusted Fault Tolerant Model of MANET with Data Recovery", IEEE 4th International Conference on Intelligent Networks and Intelligent Systems, Pages 21-24,201.

[35] S. Corson and J. Macker. Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations. RFC 2501, Jan.1999.

[36] X. Hong, K. Xu, and M. Gerla.Scalable routing protocols for mobile ad hoc networks. Network, IEEE, 16(4):11 {21, Jul./Aug. 2002.

[37] M. Abolhasan, T. Wysocki, and E. Dutkiewicz.A review of routing protocols for mobile ad hoc networks. Ad Hoc Networks, 2(1):1 { 22, 2004.

[38] C. E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance- vector routing (dsdv) for mobile computers. SIGCOMM Comput.Commun. Rev., 24:234{244, October 1994.

[39] E. M. Royer, C. E. Perkins, and S. R. Das. Ad hoc on-demand distance vector (aodv) routing, 2000.

[40] GhaidaaMuttasherAbdulsaheb, NorrozilaSulaiman, OsamahIbrahemKhalaf, "Design and Implementation A new Cluster Based Routing Protocol for Ad-hoc Network", *Baghdad, Iraq*, October,2015.

[41] B. Christian, "On the Minimum Node Degree and Connectivity of a Wireless Multihop Network", Lausanne, Switzerland, June 9-11, 2002, Available at: http://web.cse.ohio-state.edu/~lai/788-Au03/papers/connectivity/k-neigh-mobihoc03.pdf

[42] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks, 1996

[43] M.Gerla and J.T.C.Tsai, “multicluster, mobile,multimedia radio network”,*wireless networks*,1,3,pp.255-265,1995.

[44] A. K. Parekh, “Selecting routers in ad hoc wireless network”, *Proceedings of the SBT/IEEEinternational telecommunication symposium*, August 1994.

[45] D.J.Baker, J.E.Wieselthier and A. Ephremides, “A design concept for reliable mobile radio networks with frequency hoping signaling”, *proceedings of IEEE,* vol. 75, no.1, Jan.1987.

[45] M.Gerla and J.T.C.Tsai, “multicluster, mobile,multimedia radio network”, *wireless networks*,1,3,pp.255-265,1995.

[46] A. K. Parekh, “Selecting routers in ad hoc wireless network”, *Proceedings of the SBT/IEEE international telecommunication symposium,* August 1994.

[47] P. Basu, N. Khan and T.D.C. Little, “A mobility based metric for clustering in mobile Adhocnetworks”, *Proceedings of IEEE ICDCS 2001 workshop on wireless networks and Mobilecomputing,* Phoenix, AZ, 2001.

[48] S.Basagni, “Distributed and mobility-adaptive clustering for multimedia support in multi-Hop wireless networks”, *proceedings of vehicular technology conference, VTC,* vol.2, pp. 889-893, 1999.

[49] S. Basagni,“Distributed clustering for ad hoc networks”, *Proceedings of International Symposium onParallel Architectures, Algorithms and Networks,* pp.310–315, June 1999.

[50] S.Das, M.Chatterjee, D.Turgut, “Wca: A weighted clustering algorithm for mobile ad hoc networks”,*Journal of Cluster computing (special issue on mobile ad hoc networks),* vol.5, no. 2, 4, pp. 193-204,2002.

[51] S.Das, M.Chatterjee, D.Turgut, “An on demand weighted clustering algorithm(WCA) for Adhocnetworks”, *in proceedings of IEEE GLOBECOM 2000,* pp.1697-1701, San Francisco, Nov. 2000.

[52] Wei-dong Yang and G-z Zhang, “A weight-Based Clustering Algorithm for mobile Ad-Hoc Network”, *proceedings of Third International Conference on Wireless and Mobile Communication*(ICWMC’07), 2007.

[54] D.J.Baker and A. Ephremides, “*The architectural organization of a mobile radio network via a distributed algorithm*”, IEEE Transactions on Communications COM-29, pp. 1694-1701,11(Nov.1981).

[55] SuchismitaChinara and Santanu Kumar Rath, “Taca: A Topology Adaptive Clustering Algorithm for Mobile Ad Hoc Networks”, Proceedings of WORLDCOMP International Conference on Wireless Networks (ICWN ’09), July 14th-17th, Las Vegas, USA, pp. 391-397, 2009.

[56] M. Gerla and J. T.-C. Tsai, “Multicluster, Mobile Multimedia Radio Networks,” Wireless Networks 1, pp. 255-265, 1995.

[57] A. phremides, J. E. Wieselthier, and D. J. Baker, “A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling,” Proceedings of the IEEE, Vol. 75, No. 1, January 1987, pp. 56–73.

[58] C-K. Toh, “Associativity-Based Routing for Ad-Hoc Mobile Networks,” International Journal on Wireless Personal Communications, Vol. 4, No. 2, 1997.

Appendix

 # NS-2 codes to Define a 'finish' procedure and run NAM

```
proc finish {} {

global ns tracefile namfile

$ns flush-trace

close $tracefile

 close $namfile

 exec nam out.nam &
```

 # NS-2 codes for displaying X-Graphs

```
exec xgraph througput.tr  New_Throuhput.tr  -geometry 800x400 -t "Network Throughput" -x "Time(ms)" -y "Traffic(Bits)" -bg white &

exec xgraph centralized2.tr  -geometry 800x400 -t "Cluster Head Selection " -x "Node Energy" -y "Nodes" -bg white &

exec xgraph Packets Transmitted within Cluster  centralized3.tr  -geometry 800x400 -t "Cluster networks Performnace " -x "Time (ms)" -y "cluster" -bg white &

exec xgraph  centralized4.tr  -geometry 800x400 -t "Fault Tolerance Cluster Effcency Performnace" -x "Time" -y "Nodes" -bg white &

exec xgraph centralized5.tr  -geometry 800x400 -t "Number of  cluster Network" -x "Tme (ms)" -y "No.of Nodes within Networks" -bg white &

exec xgraph centralized6.tr  -geometry 800x400 -t "Fault Tolerance PERFORMANCE REPORT" -x "Time (ms)" -y "Number of cluster" -bg white &

exec xgraph lifetime.tr New_Lifetime.tr -geometry 800x400 -x "Number of Rounds" -y "To of nodes" -t "Network life Time" -bg white  &

exec xgraph centralized7.tr  -geometry 800x400 -t " Fault Tolerance Time taken for Cluster"   -x "Time" -y "No.of nodes act as Head" -bg white &

exec xgraph pkdropped.tr  -geometry 800x400 -t "PACKET LOSS ANALYSIS" -x "Time" -y " Packetsdelivery ratio" -bg white &

exec xgraph packetsreceived.tr -geometry 800x400 -t "Performance Analysis" -x "Time" -y "Number of Packets" -bg white &
```

exec xgraph density.tr -geometry 800x400 -t "Cluster Density Analysis" -x "Number of Nodes" -y "Size of Network Area" -bg white &

exit 0

}

 # NS-2 Codes for generating Trace files

set f0 [open througput.tr w]

set f1 [open delivryratio.tr w]

set f2 [open packtdelay.tr w]

set f3 [open packtdropped.tr w]

set f4 [open n4-delay1.tr w]

set f5 [open n5-delay2.tr w]

set f6 [open n6-packets_received.tr w]

set f7 [open n7-packets_received.tr w]

set f8 [open n8-packets_received.tr w]

set f9 [open n9-throughput.tr w]

set f10 [open n10-packets_received.tr w]

set f11 [open n11-packets_received.tr w]

set f12 [open n(12)-packets_received.tr w]

set f13 [open n(13)-packets_received.tr w]

set f14 [open n(14)-packets_received.tr w]

set f15 [open n(15)-packets_received.tr w]

set f16 [open n(16)-deliveryratio.tr w]

 # NS-2 Codes for setting nodes location in the Network

set node_(0) [$ns node]

$node_(0) set X_ 299

```
$node_(0) set Y_ 248
$node_(0) set Z_ 0.0
$ns initial_node_pos $node_(0) 35
set node_(1) [$ns node]
$node_(1) set X_ 297
$node_(1) set Y_ 496
$node_(1) set Z_ 0.0
$ns initial_node_pos $node_(1) 35
set node_(2) [$ns node]
$node_(2) set X_ 297
$node_(2) set Y_ 496
$node_(2) set Z_ 0.0
$ns initial_node_pos $node_(2) 35
set node_(3) [$ns node]
$node_(3) set X_ 297
$node_(3) set Y_ 496
$node_(3) set Z_ 0.0
$ns initial_node_pos $node_(3) 35
set node_(4) [$ns node]
$node_(4) set X_ 297
$node_(4) set Y_ 496
$node_(4) set Z_ 0.0
```