M.Sc. Engg. Thesis

Design and Analysis of a Fault-Tolerant Topology Control Algorithm for Wireless Multi-hop Networks

Submitted in partial fulfillment of the requirements for the award of the degree of Master of Science in Computer Science and Engineering

Submitted by Md. Sirat Samyoun
Student Number: 1014052006

Under the Supervision of Dr. A.K.M Ashikur Rahman

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
Dhaka, 1000
Dedicated to my loving parents

Author’s Contact

Md. Sirat Samyoun
Email: siratsamyoun@gmail.com
Candidate's declaration

It is hereby declared that the whole thesis or part of it has not been taken from other works without reference. It is also declared that this thesis or part of it has not been submitted elsewhere for the award of any degree or diploma.

(Signature of the Candidate)

Md. Sirat Samyoun
The thesis titled "Design and Analysis of a Fault-Tolerant Topology Control Algorithm for Wireless Multi-hop Networks", submitted by Md. Sirat Samyoun, Roll No. 1014052006, Session October 2014, to the Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, has been accepted as satisfactory in partial fulfillment of the requirements for the degree of Master of Science in Computer Science and Engineering and approved as to its style and contents. The examination held on October 24, 2017.

Board of Examiners

Dr. A.K.M Ashikur Rahman
Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka.

Chairman
(Supervisor)

Dr. M. Sohel Rahman
Head and Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka.

Member
(Ex-Officio)

Dr. Md. Shohrab Hossain
Associate Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka.

Member

Dr. A. B. M. Alim Al Islam
Associate Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka.

Member

Dr. Rajesh Palit
Associate Professor
Department of Electrical and Computer Engineering
North South University, Dhaka.

Member
(External)
Acknowledgments

First of all I would like to thank my supervisor, Dr. A.K.M Ashikur Rahman, for introducing me to the fascinating field of Topology Control for Wireless Multi-hop Networks. Without his continuous guidance and tremendous support, it would have been impossible to complete the thesis. I am also grateful to him for allowing me to chose my course of actions with freedom, for enlightening me during the moments of frustration, for his patience throughout the entire research work and for his moral support to the extra-curricular activities that I accomplished aside from the thesis. Moreover, the excellent research milieu of the department of Computer Science and Engineering always worked as a constant motivation for me. Specially, I remember all the respected teachers and the staff who earnestly provided me with encouragement and inspiration for achieving this goal. I would also like to thank Bangladesh University of Engineering and Technology for providing me with all necessary research facilities and generous support throughout the thesis. Last but not least, I am grateful to my parents for their patience, interest, and support during my studies.
Abstract

Topology control is a fundamental problem in wireless multi-hop networks where the goal is to determine a set of wireless links satisfying some desirable properties such as planarity, symmetricity, fault tolerance, minimum energy, bounded power stretch factor etc. Among these, fault tolerance which provides the ability of a topology to maintain connectivity even in the presence of failure of one or more node(s) or link(s) is very hard to achieve. Although a number of fault tolerant structures have been proposed recently, they lack any serious mathematical analysis to estimate their performance. Among the topologies derived from the topology control algorithms, the $r$–neighborhood graph is a significant class of planar topologies. Moreover, it is a generalized structure to two other widely used graph structures, such as the Gabriel Graph and the Relative Neighborhood Graph. However, the $r$–neighborhood graph does not ascertain any kind of fault tolerance. So, we fill this notable gap in the literature by augmenting the existing algorithm for constructing $r$–neighborhood graph to provide fault tolerance and flexibility. After designing the algorithm we build mathematical model(s) for various performance metrics of the topology created by the algorithm. We also validate the correctness of the built analytical models through extensive simulation results.
Contents

Acknowledgements ........................................... 3

1 Introduction ............................................... 1
   1.1 What is a Wireless Multi-Hop Network ............... 1
   1.2 Research Issues on Wireless Multi-Hop Networks .... 1
      1.2.1 Throughput maximization .......................... 2
      1.2.2 Reliability and Security .......................... 2
      1.2.3 Quality of Service (QoS) ......................... 2
      1.2.4 Scalability ........................................ 2
      1.2.5 Topology Control ................................. 2
      1.2.6 Fault Tolerance ..................................... 3
      1.2.7 Energy Efficiency ................................. 3
      1.2.8 Internet of Things (IoT) ......................... 3
   1.3 Our Focus ............................................ 3
      1.3.1 What is a Network Topology ....................... 3
      1.3.2 Topology Control(TC) ............................. 4
      1.3.3 What is Fault Tolerance ......................... 4
   1.4 Our Contributions ..................................... 5
   1.5 Organization of the Report ............................ 5

2 Background and Proposition of our research .................... 7
   2.1 State-of-the-art ....................................... 7
      2.1.1 A brief history of Fault Tolerance ............... 7
      2.1.2 Few significant proximity graphs from Topology Control ... 8
      2.1.3 Gabriel Graph (GG) ................................ 8
      2.1.4 Relative Neighborhood graph ..................... 9
      2.1.5 The r–neighborhood graph ..................... 9
2.1.6 Fault-tolerant topology control algorithms .......... 10

2.2 Our Proposition ............................................. 11
   2.2.1 Why $r$-neighborhood graph ......................... 12
   2.2.2 Few definitions ....................................... 12
   2.2.3 Few definitions related to graph theory ............ 12
   2.2.4 Few definitions related to topology control ...... 13
   2.2.5 Few definitions of network and transceiver parameters 14
   2.2.6 Few definitions related to the $r$-neighborhood graph 14
   2.2.7 A description about the $r$-neighborhood graph ...... 14
   2.2.8 The $r$-neighborhood region ......................... 14
   2.2.9 Area of the $r$-neighborhood region ............... 15
   2.2.10 Some performance metrics for the $r$-neighborhood graph ................................................. 16

3 Design of a Fault-Tolerant Topology Control Algorithm 18
   3.1 Algorithm for $r$-neighborhood graph construction .... 18
   3.2 Providing fault tolerance to the $r$-neighborhood topology control algorithm .................................... 19
       3.2.1 $NG_t^2$ topology construction ..................... 20
       3.2.2 Theorems ........................................... 21
       3.2.3 Topology with a desired degree of connectivity ... 23
       3.2.4 $NG_r^k$ topology construction .................... 23
       3.2.5 Sample Topologies after running TC .............. 24
       3.2.6 Validating the parameter $k$ in Sample Topologies 28

4 Analytical Modeling for Performance Metrics 31
   4.1 Performance Metrics ...................................... 31
       4.1.1 Sparseness ......................................... 33
       4.1.2 Average Node Degree ................................ 36
       4.1.3 Topology Size ...................................... 36

5 Simulation and Performance Evaluation 37
   5.1 Performance evaluation .................................... 37
   5.2 Simulation Environment .................................. 37
   5.3 Simulation Results ...................................... 38
   5.4 Effect on Sparseness .................................... 38
       5.4.1 Effect of Node Density on Sparseness ........... 38
5.4.2 Effect of Transmission Range on Sparseness ........ 39
5.4.3 Effect of $r$ on Sparseness .......................... 39
5.4.4 Effect of $k$ on Sparseness .......................... 40
5.5 Effect on Average Node Degree and the Topology Size ... 45
  5.5.1 Effect of Node Density .............................. 45
  5.5.2 Effect of $r$ ........................................ 46
  5.5.3 Effect of Transmission Range ....................... 46
  5.5.4 Effect of $k$ ........................................ 46
5.6 Mathematical analysis of effect of the parameters on Performance Metrics .............................. 49

6 Conclusion and Future Works ............................. 52
  6.1 Conclusion ........................................... 52
  6.2 Future Works ........................................ 53

7 Appendix .................................................. 54
  7.0.1 Graph .............................................. 54
  7.0.2 Graph Model of a Network ......................... 54
  7.0.3 Degree of a Node .................................. 54
  7.0.4 Connected Graph ................................. 54
  7.0.5 Planarity of a Graph ............................... 55
  7.0.6 Gabriel Graph .................................... 55
  7.0.7 Relative Neighborhood Graph ..................... 55
  7.0.8 Minimum Spanning Tree ............................ 55
  7.0.9 Binomial Distribution ............................. 56
  7.0.10 Poisson distribution ............................. 56
# List of Figures

2.1 GG and RNG .................................................. 9
2.2 The r-neighborhood region, $NG_r(u,v)$ ..................... 15

3.1 Proof by case: $NG^1_r$ may not be bi-connected .......... 22
3.2 Initial Topology, ($N=100$, TX Range=175m) ................ 25
3.3 Final Topology ($k=1$, $r=0.0$) ............................. 25
3.4 Final Topology ($k=2$, $r=0.0$) ............................. 26
3.5 Final Topology ($k=3$, $r=0.0$) ............................. 26
3.6 Final Topology ($k=1$, $r=1.0$) ............................. 27
3.7 Final Topology ($k=2$, $r=1.0$) ............................. 27
3.8 Final Topology ($k=3$, $r=1.0$) ............................. 28
3.9 Finding out $k$ in Final Topology ( $k=1$, $N=100$, TX=175m, $r=1.0$) ......................................................... 29
3.10 Finding out $k$ in Final Topology ( $k=2$, $N=100$, TX=175m, $r=1.0$) ......................................................... 30

4.1 Illustrating circular strip at distance $x$ ....................... 32

5.1 Effect of Node density on Sparseness, ($k=1$, $r=0.0$) ...... 40
5.2 Effect of Node density on Sparseness, ($k=2$, $r=0.0$) ...... 41
5.3 Effect of Node density on Sparseness, ($k=1$, $r=1.0$) ...... 41
5.4 Effect of Node density on Sparseness, ($k=2$, $r=1.0$) ...... 42
5.5 Effect of Transmission Radius on Sparseness, ($k=1$, $r=1.0$) ...... 42
5.6 Effect of Transmission Radius on Sparseness, ($k=2$, $r=1.0$) ...... 43
5.7 Effect of $r$ on Sparseness, ($N=500$, TX Range=125m) ...... 43
5.8 Effect of $r$ on Sparseness, ($N=500$, TX Range=225m) ...... 44
5.9 Effect of $k$ on Sparseness (TX Range=175m, $r=0.5$) ...... 44
5.10 Effect of $k$ on Sparseness, ($N=300$, $r=0.5$) .............. 45
5.11 Effect of Node Density on Topology Size (TXR=175m, r=0.5) 47
5.12 Effect of TX Range on Topology Size (N=100, r=0.0, k=1) 47
5.13 Effect of r on Size of the Topology (N=300, TX Range=175m) 48
5.14 Effect of k on Size of the Topology using bar-chart (N=300, TX Range=175m) 48
Chapter 1

Introduction

1.1 What is a Wireless Multi-Hop Network

A Wireless Multi-Hop Network is a wireless network of nodes that uses multiple hops for data transmission. In such networks, routing of data packets is done through one or more intermediate nodes along the path from source to destination. Moreover, transmission over multiple "short" links might require less transmission power and energy than over "long" links. Such networks doesn’t need on a pre-existing infrastructure or any centralized management. Thus multi-hop relaying is a very promising solution for increasing throughput and providing coverage for a large physical area.

Four network paradigms can be classified as wireless multi-hop networks. These paradigms are:

- Mobile Ad-Hoc Networks (MANETs)
- Wireless Sensor Networks (WSNs)
- Wireless Mesh Networks (WMNs)
- Vehicular Ad-Hoc Networks (VANETs)

1.2 Research Issues on Wireless Multi-Hop Networks

Since its introduction, wireless communication have significant advantages over wired communication. Wireless multi-hop networks has drawn heavy
attention due to flexible deployment, cost reduction, mobility and convenience for user service providers. Therefore such networks have reached high interest in the research community but also many practical application examples, scenarios and issues have been developed. Some of the research issues are discussed below-

1.2.1 Throughput maximization

Throughput maximization is one of the key challenges in multi-hop routing. Several factors that degrade the throughput, for example, the presence of the hidden node have drawn huge attention in the research community.

1.2.2 Reliability and Security

Ensuring reliable and secure communication in highly dynamic or hostile environments is one of the key challenges in multi-hop routing. There has been significant amount of research in achieving fast, reliable and secure communication in MANET, VANET etc.

1.2.3 Quality of Service (QoS)

QoS is a vital element of understanding the efficiency of real-world computer networks in terms of consistency of service and satisfaction of customers' application needs. So, implementing QoS-aware multi-hop routing schemes has been receiving increasing attention among the research community.

1.2.4 Scalability

The routing protocols applied to wireless networks sometimes do not have enough scalability as throughput drops significantly as the number of nodes or hops increases. So, effective configuration schemes to ensure scalability must be studied in such networks.

1.2.5 Topology Control

Topology control algorithms have been proposed to construct efficient network topologies with several design goals such as- Planarity, Connectivity,
Symmetricity etc. Moreover, topology control aims to preserve communication energy, reduce interference between nodes and increasing the lifetime of the network.

1.2.6 Fault Tolerance

The significance of implementing fault tolerance networking protocols is about service continuity or maintaining functionality in the event of the faults or failures of one or more of its components. However, implementing these systems can be sometimes costly. The issue of providing fault tolerance to networking protocols has been widely studied in the past decades.

1.2.7 Energy Efficiency

Constructing power-aware multi-hop routing protocols has become a serious concern of the network service providers today. Moreover, the scientific evidence of global warming has caused the urge for building energy efficient networks.

1.2.8 Internet of Things (IoT)

Internet of Things (IoT) is without a doubt one of the greatest technology revolutions today. With the advent of Wi-Fi and large number of IoT devices, there has been a renewed focus on multi-hop networking to facilitate communications between these devices.

1.3 Our Focus

Topology control is one of the fundamental research problems in wireless multi-hop networks. The main aim of topology control in this domain is to save energy, reduce interference between nodes and extend lifetime of the network. Again, fault tolerance is a key property that is often overlooked by the research community. Our research will be focused on the intersection of these two key issues in wireless multi-hop networks.

1.3.1 What is a Network Topology

A network topology means the layout of a network. It refers to how different nodes in a network are connected to each other. In a wireless multi-hop
network, each node can communicate to the other nodes those are within its transmission range. A transmission range is the transmitting radius of a node. Such nodes within each other's transmission range can be connected through virtual links. By weaving all virtual links between every transceiver pairs, the topology of a wireless network is derived.

1.3.2 Topology Control (TC)

Topology control is a technique in which a network topology is altered by removing the unnecessary links such that the composed topology satisfies some desirable properties. Some of the significant properties achieved from topology control are namely- planarity, connectivity, symmetricity, minimal use of energy [1]. The underlying is modeled as a graph structure in this technique. An example of a Topology Control (TC) algorithm is A Minimum Spanning Tree, which is used as a backbone to reduce the cost of broadcast from $O(m)$ to $O(n)$, where $m$ and $n$ are the number of edges and vertices in the graph, respectively.

A significant use of topology control is the reduction of both transmission power and traffic interference. All the nodes in a wireless multi-hop network no longer have to transmit with maximum power. Instead, the nodes collaboratively determine their transmission power by gathering their neighborhood information under a specific topology control algorithm. Thus, topology control can save communication energy, which eventually prolongs network lifetime too.

1.3.3 What is Fault Tolerance

In general, Fault-tolerance is the ability of a system or component to continue normal operation despite the presence of one or more faults within some of its components. For a network topology, fault tolerance means maintaining the network connectivity even in the presence of failure of one or more node(s) or link(s). Achieving fault tolerance is hard and sometimes costly due to it's challenges to achieve. In particular, for hostile environments where faults and failures are common phenomena fault tolerance possess great importance.

Reliability and Availability are the key attributes of a fault-tolerant system. Other attributes include Maintainability, Testability, Performability
etc. There are various techniques for achieving fault tolerance of a system. Some of these techniques are Fault Masking, Redundancy, Reconfiguration etc. These attributes along with the techniques have been explained in [2].

1.4 Our Contributions

Three important contributions of our research work are as follows:

i. We provide an algorithm for constructing a new fault tolerant topology that will maintain network connectivity in case of a breakdown of node(s) or link(s) but as energy efficient as the original topology by maintaining bounded power stretch factor and maximum node degree.

ii. Flexibility in topology control by providing tunable parameter(s) to generate two sets of topologies— (a) planar, and (b) fault tolerant. By tuning the parameter(s), the network designer will have the ability to switch between the two as per desired requirements.

iii. Performance studies on wireless multi-hop networks are mostly based on simulations. Analytical studies are rare. We provide—

   i. The analytical expressions for calculating key performance metrics such as sparseness, average node degree, and topology size that will be used by the network designers prior to the deployment of the real systems without running simulations.

   ii. A comprehensive comparison of the proposed analytical models with the simulation results. The accuracy of the analytical models will thus be determined by observing how closely the simulation results follow the analytical results.

   iii. We also demonstrate how to analytically couple sparseness of such topologies with the radio transceiver parameters.

1.5 Organization of the Report

The rest of the book is organized as following:

1. In the chapter 2, we present state-of-the-art research works deriving fault tolerant topologies in wireless multi-hop networks and deduce the target of our study.
2. In the chapter 3, we augment the existing algorithm for $r$—neighborhood topology construction to provide fault tolerance and flexibility.

3. The analytical models for the performance metrics *Sparseness*, *Average Node Degree*, and *Topology Size* of our proposed topology will be derived in the chapter 4.

4. A comprehensive comparison of the proposed analytical models with the simulation results for different settings of network and transceiver parameters will be presented in the chapter 5. The effects of these parameters on these metrics were also found out mathematically in the same chapter.

5. Chapter 6 concludes our work with the contributions and limitations of our study and possible future works.
Chapter 2

Background and Proposition of our research

In this chapter, we present some of the topology control algorithms for wireless multi-hop networks. None of these works ascertain any kind of fault tolerance. We also discuss state-of-the-art research works deriving fault tolerant topologies in wireless multi-hop networks. However, there is no mathematical analysis or analytical modeling for performance evaluation of these topologies. Thus we deduce the target of our study and also provide an outline of our research plan.

2.1 State-of-the-art

2.1.1 A brief history of Fault Tolerance

The fundamental principles of fault tolerance was laid out by John von Neumann in the 1950s in his work of incorporating redundancy [3] in order to improve the reliability of a system. Subsequently, in 1967, Avizienis worked on the design of fault-tolerant systems [4] by integrating these techniques for error detection, diagnosis and recovery. As day by day computer systems grew, fault tolerance became a basic design concept and a unique field of study among the research community. The taxonomy of the fault tolerance along the key attributes and several techniques of achieving fault tolerance have been explained in [2].

There are several techniques for achieving a fault-tolerant system. Some of them are- Reconfiguration, Fault masking, Redundancy etc. Reconfigu-
ration is technique in which the faulty entity from a system is eliminated in such a way that system to allowed to restore to some operational state [5]. Moreover, introducing redundancy for some of its elements allowing the whole system to operate correctly even after some failures of its elements with a view to ensure high availability has been an approach widely used in research community. For example, the authors in [6], [7] and [8] have used redundancy to achieve fault tolerant network topologies while the authors in [9] have used a reconfiguration algorithm for proving fault tolerance to the network topology.

2.1.2 Few significant proximity graphs from Topology Control

A number of proximity graphs derived from the topology control algorithms have been widely used in various fundamental research problems in wireless networks. A proximity or neighborhood graph is a graph where the existence of an edge between two vertices is decided if the vertices satisfy particular geometric requirements. The main idea of such graph structures is to trade several properties such as sparsity, node degree for creating graphs with better properties in other dimensions such as energy, delay etc. Some of the proximity graphs borrowed from the computational geometry are Gabriel Graph (GG) [10], Relative Neighborhood graph (RNG) [11], Delaney triangulation (DT) [12], Euclidean Minimum Spanning Tree (EMST) etc. These graph structures has been widely used in various protocols in wireless networks. Few of them are discussed below.

2.1.3 Gabriel Graph (GG)

Gabriel Graph, GG(V) is a proximity graph has an edge uv between vertices $u, v \in V$, iff the circular region drawn using $\|uv\|$ as diameter contains there is no node $w \in V$. The shaded region in the figure 2.1a depicts this proximity or neighborhood region between $u, v$. Gabriel graphs are named after K. R. Gabriel, who introduced them with R. R. Sokal in 1969. the Gabriel Graph was used for the delivery guarantees of face and combined greedy-face routing in ad hoc and sensor networks [13].
2.1.4 Relative Neighborhood graph

The Relative Neighborhood Graph, RNG(V) is another proximity graph that has an edge uv between vertices \( u, v \in V \), iff there is no node \( w \in V \) in the intersecting region of the circles centered at \( u \) and \( v \) with a radius of \( ||uv|| \), which is the proximity region for this graph. This proximity or neighborhood region between \( u, v \) is shown in the figure 2.1b. This graph was proposed by Godfried Toussaint in 1980.

The Relative Neighborhood Graph (RNG) was used in the internal node based broadcasting algorithms for wireless one-to-one networks [14].

2.1.5 The \( r-\)neighborhood graph

Recently, the \( r-\)neighborhood graph, a adjustable set of proximity graphs that trade between energy and node degree in a tunable manner has been proposed [16]. It has been shown [16] that both RNG and GG are special instances of the \( r-\)neighborhood graph. The details of the \( r-\)neighborhood graph has been discussed later in this chapter.

Some other examples of such neighborhood graphs are Delanney triangulation (DT) [12], which has been used in applications in path planning in automated driving, the Euclidean Minimum Spanning Tree (EMST), which has been used in approximation algorithm for approximately solving the Euclidean traveling salesman problem etc. Recently, authors in [15] used the neighborhood of nodes in a network topology to find the spread of social contagions in networks.

Moreover, all of the above mentioned proximity graphs are also planar graphs. A topology is said to be planar if there is no link crossing in the
topology. Planar graphs are used in several problems like scheduling and map coloring and various protocols like GPSR routing. Moreover, planar graphs are also very easy to define and can be constructed by simple inductive constructions. However, none of the topology control algorithms generating these significant topologies ascertain any kind of fault tolerance.

2.1.6 Fault-tolerant topology control algorithms

A number of research works exist to provide fault tolerance for wireless multi-hop networks using various techniques. Avresky et al. [9] used reconfiguration for tolerating multiple node and link failures in the network. Their proposed a dynamic network reconfiguration algorithm restores the connectivity between the nodes of the network topology. Several fault-tolerant topology control algorithms have used the redundant vertex disjoint paths to maintain network connectivity in case of node failures. Among them, Shen et al. [17] and Roy et al. [17] proposed fault tolerant topologies can tolerate any one node failure and avoid network partition by preserving bi-connectivity. However, a serious drawback of the these works is that their constructed topologies can not provide any fault tolerance when more than one nodes goes down. Bahramgiri et al. [6], later on, used the idea of preserving the connectivity of a network upon failing of, at most, \( k \) nodes. Li et al. [8] too considered \( k \)-connectivity of wireless network to meet the requirement of fault tolerance. Their proposed Fault-tolerant Global Spanning Subgraph (FGSSk) preserved \( k \)-connectivity.

Authors in [18] too used the idea of introducing redundancy to provide fault tolerance to a network topology by using the existence of multiple internally vertex-disjoint paths between each pair of the nodes. They have defined a network to be \( k \)-vertex connected if \( k \)-vertex disjoint paths exist between each pair of nodes. They have concluded that a \( k \)-vertex connected graph has the property that the failure of any set of \( (k - 1) \) nodes cannot disconnect the network.

However, none of the above works provide any serious performance analysis or any analytical models for any performance metrics of these fault tolerant topologies. With such mathematical modeling to estimate their performance, it would have been easy network designers to choose different network and transceiver parameters for his/her desired network without running any simulations or prior to the deployment of the real systems. Recently
the mathematical modeling of performance metrics such as Sparseness, Average Node degree, and Topology Size of the $r$-neighborhood graphs, a significant proximity graph described above, which doesn't provide any fault tolerance, have been developed [19]. However, fault tolerance remains a significant property for these proximity graphs derived from topology control algorithms in case of environments where faults and failures are common. Moreover, by reducing the number of links in the network, topology control algorithms actually decrease the degree of routing redundancy, and hence the topology thus derived is more susceptible to node failures [8]. So, we find this notable gap of literature to provide fault tolerance to this significant proximity graph derived from topology control algorithm and to estimate the performance metrics for such graphs in wireless multi-hop networks.

### 2.2 Our Proposition

The target of this research work is to fill this notable gap of literature by proposing a fault tolerant topology control algorithm for the $r$-neighborhood graph for wireless multi-hop networks and to develop analytical models for several key performance metrics for our proposed topology. We will augment the existing algorithm for $r$-neighborhood graph to provide fault tolerance. Our system design would use redundancy technique for tolerating the faults. In our study we consider the failures of nodes as faults. Using this technique, the underlying topology will be $k$-vertex connected in order to ensure availability and network connectivity in case of failures of nodes. In our study, we call this as a $k$-connected topology. That means, failures of at most $(k-1)$ nodes in the topology does not disconnect it.

In our research we will consider the values of $k$ from from 1 up to 2 for deriving analytical expressions and running simulations. The simulation results will be used to reveal the accuracy of the analytical models. Moreover we want to provide flexibility to the network designer to choose between fault tolerant and planar topologies for these values of $k$. Finally we want to quantitatively explore the effects of several network and transceiver parameters on the developed analytical expressions for performance metrics.
2.2.1 Why $r$–neighborhood graph

The $r$–neighborhood graph was proposed as an adjustable structure for topology control in wireless ad-hoc networks [16]. The key motivation behind picking the $r$–neighborhood graph to provide fault tolerance is that it is a generalized or an adjustable structure, that is it provides a range of planar topologies as its instances for $r = 0$ to $1$. In particular, for $r = 0$, the $r$–neighborhood graph becomes a Gabriel Graph (GG) and for $r = 1$, it becomes a Relative Neighborhood graph (RNG). This relationship between three graph structures is formally shown below:

$$RNG \subseteq NG_r \subseteq GG$$

Secondly, the $r$–neighborhood is a proximity graph and also a planar graph. As we have previously discussed the significance in proximity graphs and planar graphs in research community, we found the $r$–neighborhood graph as an interesting graph structure to provide fault tolerance and flexibility to the topology control algorithm.

2.2.2 Few definitions

In this section we present few definitions related to the graph theory that would be needed in the upcoming sections. We also provide definitions for few terms related to the topology control, different network and transceiver parameters and the $r$–neighborhood graph that would be necessary in the algorithms and analytical modeling sections.

2.2.3 Few definitions related to graph theory

**Definition 2.2.1. Connected graph.** A graph is connected when there is a path between every pair of vertices.

**Definition 2.2.2. Singly-connected graph.** In graph theory, a singly-connected graph is a connected graph that doesn’t ensure that the graph will remain connected if any vertex were to be removed.

**Definition 2.2.3. Articulation point.** Articulation point is a vertex in an undirected connected graph iff removal of the vertex along with its edges disconnects the graph by dividing the network into two or more components.
Definition 2.2.4. Bi-connected graph. In graph theory, a bi-connected graph is a connected and inseparable graph such that if any single vertex were to be removed, the graph will remain connected. A graph is bi-connected iff it contains no articulation point.

Definition 2.2.5. \( k \)-vertex connectivity. A graph \( G \) is \( k \)-vertex connected if for any two vertices \( u, v \in V(G) \), there are \( k \) pairwise-internally-vertex-disjoint paths between \( u \) and \( v \). Or equivalently, a graph is \( k \)-vertex connected if the removal of any \( k - 1 \) nodes (and all the related links) does not disconnect the network.

Definition 2.2.6. Proximity or Neighborhood graph. A proximity or neighborhood graph is a graph where the existence of an edge between two vertices is decided if the vertices satisfy particular geometric requirements within a region. This region is known as neighborhood region, which can be varied by a parameter.

2.2.4 Few definitions related to topology control

Definition 2.2.7. Sparseness. Sparseness of the topology is the expected fraction number of neighbors removed from a node’s neighborhood list after pruning. Keeping a higher value of sparseness would reduce the expenditures of the network topology.

Definition 2.2.8. Average Node Degree. The Average Node Degree of the topology is the average number of neighbors retained in a node’s neighbor list after pruning. A node with a higher node degree is expected to rapidly draw out their energy.

Definition 2.2.9. Topology Size. The size of the topology is the total number of links obtained in the topology after running the topology control algorithm.

Definition 2.2.10. Power Stretch Factor. For any topology, its power stretch factor is defined as the maximum ratio of the minimum power needed to support any link in this topology to the least necessary.

Definition 2.2.11. Maximum Node Degree. The degree of a node of a graph is the number of edges incident to the vertex, with loops counted twice. The maximum node degree of a graph is the maximum degree of its nodes.
2.2.5 Few definitions of network and transceiver parameters

Definition 2.2.12. Transmission Range. Transmission Range of a node is maximum radius a node can transmit with a certain power.

Definition 2.2.13. Node density. Node density of the topology is number of nodes per square meter of deployment area of the node.

Definition 2.2.14. Neighborhood region parameter. From definition 2.2.6, the neighborhood region of a promixity graph can sometimes be varied by a parameter. This is the parameter $r$ in the $r$–neighborhood graph. This is an important network parameter in such graphs, because the existence of an edge between the vertices depend on the neighborhood region.

2.2.6 Few definitions related to the $r$–neighborhood graph

Definition 2.2.15. $r$–neighborhood cover set. The $r$–neighborhood cover set between the pair $(s,x)$, expressed as $\zeta_r(s,x)$ is the collection of nodes in the $r$–neighborhood region between $(s,x)$. This set is constructed for every node pair of the topology during the algorithm 1.

2.2.7 A description about the $r$–neighborhood graph

The $r$–neighborhood graph, $NG_r = (V,E')$, is a proximity or neighborhood graph derived from any graph $G = (V,E)$. “Proximity” here indicates spatial distance on a two dimensional space. The existence of an edge between two nodes in this graph depends on the proximity region, that is, the $r$–neighborhood region between the node pair located on a two dimensional space.

2.2.8 The $r$–neighborhood region

We formally present the $r$–neighborhood region between a node pair located on a two dimensional space. Given a node pair $(u,v)$ on $R^2$, the $r$–neighborhood region of $(u,v)$, denoted as $NR_r(u,v)$, is defined as,

$$NR_r(u,v) = D(u, \parallel uv \parallel) \cap D(v, \parallel uv \parallel) \cap D(m_{uv}, l_{uv})$$

where, $0 \leq r \leq 1$, $m_{uv}$ is the middle point on $uv$, and

$$l_{uv} = (\parallel uv \parallel /2)(1 + 2r^2)^{\frac{1}{2}}$$

14
To put simply, the \( r \)-neighborhood region between two nodes \((u,v)\) depicted in the figure 2.2 by the shaded region, is basically the intersecting area of three circles:

i. the circle centered at \( u \) with radius \( uv \).

ii. the circle centered at \( v \) with radius \( uv \).

iii. finally, the circle centered at the middle point of \( uv \) with radius.

![Figure 2.2: The \( r \)-neighborhood region, \( NG_r(u,v) \)](image)

### 2.2.9 Area of the \( r \)-neighborhood region

In [19], the authors calculated the area of the \( r \)-neighborhood region on a two dimensional space. For any two points \( u \) and \( v \) separated by distance \( \|uv\| \), the area of \( r \)-neighborhood region is:

\[
A_{\text{NR}_r(u,v)} = \|uv\|^2 \left( \pi + \frac{\alpha^2}{2} \frac{\delta}{\beta} - 2\beta + \frac{\alpha^2}{4} \sin(2\delta) - \sin(2\beta) \right)
\] (2.1)

where, \( 0 \leq r \leq 1 \), \( \alpha = \sqrt{1+2r^2} \), \( \delta = \sin^{-1} \left( \frac{\beta - \alpha^2}{2\alpha} \right) \), and \( \beta = \sin^{-1} \left( \frac{\beta - \alpha^2}{4} \right) \)
2.2.10 Some performance metrics for the $r$–neighborhood graph

Several key performance metrics of the $r$–neighborhood graphs have been analysed using mathematical models. Recently, the authors in [19] derived analytical expressions for the performance metrics such as, Sparseness ($F_e$), Average Node degree ($d_{avg}$), and Topology Size ($S$). These are as follows:

$$F_e(NG_r) = \frac{\gamma^{\mu R^2} + e^{-\gamma^{\mu R^2}} - 1}{\gamma^{\mu R^2}}$$

$$d_{avg}(NG_r) = \frac{\pi}{\gamma} \left(1 - e^{-\gamma^{\mu R^2}}\right)$$

$$S(NG_r) = \frac{\pi n}{2\gamma} \left(1 - e^{-\gamma^{\mu R^2}}\right)$$

where, $R$ is the transmission range of a node, $\mu$ is the node density of the deployment area and $\gamma$ is a parameter dependent on the value of $r$.

Previously two other key performance metrics of the $r$–neighborhood graphs, namely namely the Power stretch factor ($\rho$), and the Maximum node degree ($d_{max}$) were analysed in [16]. Both were defined as functions of $r$ as per the following:

$$\rho(NG_r) \leq 1 + r^\alpha(n - 2)$$

where $n$ is the number of vertices, and,

$$d_{max}(NG_r) = \frac{\pi}{\sin^{-1}(r/2)}$$

Table 2.1 summarizes a complete picture of all the known performance measures.
Table 2.1: Performance metrics of $NG_r$ at a glance

<table>
<thead>
<tr>
<th>Performance metrics</th>
<th>Notation</th>
<th>Analytical expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power stretch factor</td>
<td>$\rho(NG_r)$</td>
<td>$1 + r^\alpha(n - 2)$</td>
</tr>
<tr>
<td>Maximum node degree</td>
<td>$d_{max}(NG_r)$</td>
<td>$\frac{\pi}{\sin^{-1}(r/2)}$</td>
</tr>
<tr>
<td>Average node degree</td>
<td>$d_{avg}(NG_r)$</td>
<td>$\frac{\pi}{4} \left(1 - e^{-\gamma R^2}\right)$</td>
</tr>
<tr>
<td>Sparseness</td>
<td>$F_e(NG_r)$</td>
<td>$\frac{\gamma R^2 + e^{-\gamma R^2}}{\gamma R^2}$</td>
</tr>
<tr>
<td>Topology size</td>
<td>$S(NG_r)$</td>
<td>$\frac{\gamma n}{2\pi} \left(1 - e^{-\gamma R^2}\right)$</td>
</tr>
</tbody>
</table>

0.
Chapter 3

Design of a Fault-Tolerant Topology Control Algorithm

In this section, we augment an existing algorithm proposed in [19] for constructing a \( r \)-neighborhood graph which generates a planar singly-connected topology. We provide an adequate level of fault tolerance such that the network maintains connectivity in case of any breakdown of node(s). Next we add flexibility to the topology control algorithm by providing a *tunable* parameter \( k \), which depicts the number of nodes in the of the \( r \)-neighborhood region between every node pairs of the topology that would provide alternate paths between the pair.

3.1 Algorithm for \( r \)-neighborhood graph construction

First we see an algorithm for constructing the remaining neighbor set of any particular node in the \( r \)-neighborhood graph. We augment this algorithm by updating the \( r \)-neighborhood cover set between any node pair of the graph. Let, \( W \) be the set of all nodes in the deployment area of a given topology \( G \). For each \( s \in W \), we maintain and update the following sets:

i. the initial neighbor set of \( s \), that is, \( N_{init}(s) \). This set typically contains all the nodes within the transmission range of \( s \) following the replies from a broadcast neighbor discovery message from \( s \).

ii. the remaining neighbor set of \( s \) after the elimination of nodes from the
initial neighbor set $N_{init}(s)$ following the topology control algorithm, that is, $N_{rem}(s)$. This set is initially empty. From these two sets of nodes, we also can determine the fraction of nodes eliminated, that is the Sparseness of the topology and the other performance metrics.

iii. for each $x \in N_{init}(s)$, the $r$–neighborhood cover set between the pair $(s, x)$, that is, $\zeta_r(s, x)$. This set is initially empty. This set for every node pair will be a part of augmentation for providing fault tolerance to the topology for the next algorithm.

At first $s$ broadcasts a 'HELLO' message. All nodes within the transmission range of $s$ are expected to receive the message and send back a reply to $s$. Whenever $s$ receives such a reply from a node $v$, it executes the algorithm UpdateSetsFromReply($v$). For each $x \in N_{rem}(s)$ in line 2 of algorithm UpdateSetsFromReply($v$), we check whether the replying node $v$ is located in the $r$–neighborhood region between $S$ and $x$. If $v$ is located in the $r$–neighborhood region of $S$ and $x$ then in line 3 and 6 we update $N_{rem}(s)$ by deleting node $x$ and inserting node $v$. Now, there might be a node in $N_{init}(s)$ which is located in the $r$–neighborhood region of $s$ and $v$. This checking is done in line 8. If such a node $x$ is found, then in line 9 we update the $N_{rem}(s)$ by deleting $v$ from it. The topology thus derived will not provide any kind of fault tolerance because there are no node that could be used as relay to communicate between any two nodes of the topology in case of a node failure between the pair.

We also update the $\zeta_r(s, v)$ by inserting such a node $x$ into it, if $x$ already doesn’t exist in $\zeta_r(s, v)$. Finally in line 13 we insert node $v$ in the $N_{init}(s)$ as the reply from $v$ corresponding to the broadcast message from $s$ indicates that $v$ is within the transmission range of $s$.

### 3.2 Providing fault tolerance to the $r$–neighborhood topology control algorithm

By constructing the remaining neighbor set of all the nodes in the deployment area according to the algorithm 1, we construct the $r$–neighborhood topology for the given topology $G = (V, E)$, which does not provide any fault tolerance. We denote this topology by $NG_r^1$, where $NG_r^1 = (V, E_1)$. The superscript 1 in $NG_r^1$ indicates that the topology will be singly-connected,
Algorithm 1 \text{UpdateSetsFromReply}(v)

\begin{verbatim}
1: for each \(x\) in \(N_{\text{rem}}(s)\) do
2:   if \(\text{Loc}(v) \in NR_r(s, x)\) then
3:     \(N_{\text{rem}}(s) = N_{\text{rem}}(s) \setminus \{x\}\)
4:   end if
5: end for
6: \(N_{\text{rem}}(s) = N_{\text{rem}}(s) \cup \{v\}\)
7: for each \(x\) in \(N_{\text{init}}(s)\) do
8:   if \(\text{Loc}(x) \in NR_r(s, v)\) then
9:     \(N_{\text{rem}}(s) = N_{\text{rem}}(s) \setminus \{v\}\)
10:   if \(x \notin \zeta_r(s, v)\) then
11:     \(\zeta_r(s, v) = \zeta_r(s, v) \cup \{x\}\)
12: end if
13: end if
14: end for
15: \(N_{\text{init}} = N_{\text{init}} \cup \{v\}\)
\end{verbatim}

which we will prove in a later section.

Now provide an adequate level of fault tolerance to the topology such that the network maintains connectivity in case of any breakdown of node(s).

We propose a new topology \(NG_r^2\) for the given topology \(G = (V, E)\), where \(NG_r^2 = (V, E_2)\). The superscript \(2\) in \(NG_r^2\) indicates that the topology will be bi-connected. Now we will propose an algorithm for generating the topology \(NG_r^2\) from the topology \(NG_r^1\), which will be our desired augmentation. After proposing the algorithm, we will prove that \(NG_r^2\) ensures bi-connectivity and will provide fault tolerance when at most 1 node will go down.

3.2.1 \(NG_r^2\) topology construction

We consider the topology \(NG_r^1\) derived after running the previous algorithms. As discussed earlier, we now have the remaining neighbors list \(N_{\text{rem}}(v)\) for every node \(v\) of the topology \(NG_r^1\). We also have the \(r\)–neighborhood cover set, \(\zeta_r(u, v)\) between every pair of nodes \((u, v)\) of the topology \(NG_r^1\).

Now we run the \(NG_r^2\) topology construction algorithm.

For each node \(v\) in \(N_{\text{rem}}(s)\), the \(NG_r^2\) topology construction algorithm will check the number of nodes falling in the \(r\)–neighborhood cover set \(\zeta_r(s, v)\) between \(s\) and \(v\). If the cover set is empty or contains exactly 1 node, then \(v\) is included as a neighbor of \(s\) in \(NG_r^2\). Otherwise, \(v\) is not included as a neighbor of \(s\) because it indicates that there are at least 1
node that could be used as relay to communicate between \( s \) and \( v \). This at least 1 node will provide an additional or back-up vertex-disjoint path that ensures the topology will remain connected in case of a node failure between \( s \) and \( v \).

**Algorithm 2 NG\(_r^2\) TOPOLOGY CONSTRUCTION**

1: for each \( v \) in \( N_{rem}(s) \) do  
2: if \( \zeta_r(s,v) \) is empty or contains 1 node then  
3: \( N_{rem}(s) = N_{rem}(s) \cup \{v\} \)  
4: end if  
5: end for

### 3.2.2 Theorems

In this section we complete our proposed augmentation for providing fault tolerance to \( r \)-neighborhood topology construction algorithm by proving that \( NG_r^1 \) topology doesn’t ensure any bi-connectivity, but \( NG_r^2 \) topology ensures bi-connectivity. As we discussed earlier, a \( k \)-vertex connected topology has the property that the failure of any set of less than \( k \) nodes (edges) cannot disconnect the network. So, \( NG_r^2 \) topology will provide fault tolerance when at most 1 node or link will go down, but \( NG_r^1 \) does not ascertain any fault tolerance.

**Lemma 3.2.1.** \( NG_r^2 \) topology has all the edges of \( NG_r^1 \) topology.

*Proof.* Each edge \((u, v) \in E(G)\) is preserved in \( NG_r^1 \) iff the \( r \)-neighborhood cover set between \( u \) and \( v \) is empty, for \( NG_r^2 \), \((u, v) \in E(G)\) is preserved for following cases-

Case 1: if the \( r \)-neighborhood cover set between \( u \) and \( v \) is empty.

Case 2: if the \( r \)-neighborhood cover set between \( u \) and \( v \) contains at most 1 node.

So, \( NG_r^2 \) would contain all the edges of \( NG_r^1 \) by case 1. \( \square \)

**Theorem 3.2.2.** \( NG_r^1 \) topology may or may not be bi-connected.

*Proof.* Let, \( G = (V, E) \) be a bi-connected graph where \( V = \{u, v, w\} \) and \( E = \{(u, v), (v, w), (u, w)\} \) as shown in figure 3.1a. Let us assume that node \( w \) is in the \( r \)-neighborhood region between pair \((u,v)\). In such a case, the \( r \)-neighborhood cover set between pair \((u,v)\) has only one node, \( w \),
that is, \( \zeta_r(u,v) = w \). By definition, edge \((u,v) \in E(G)\) will not be preserved in \(NG^1_r\), as shown in figure. Now, \( w \) will be an articulation point in \(NG^1_r\), because the removal of \( w \) in figure 3.1b will clearly disconnect the graph because edge \((u,v)\) is no longer preserved. So, bi-connectivity is not ensured. In other words, \(NG^1_r\) only ensures singly-connectivity.

\[ \square \]

**Theorem 3.2.3.** \(NG^2_r\) topology ensures bi-connectivity.

*Proof.* To prove by contradiction, we assume a bi-connected graph \( G = (V,E) \) and \( u \in V(G) \) and \( v \in V(G) \) be two nodes. Let, there exists two paths \(<u,x,v>\) and \(<u,y,v>\) where \( x \) and \( y \) are two nodes in the \( r \)-neighborhood cover set between \( u \) and \( v \).

So, according to \(NG^2_r\) topology construction algorithm, \((u,v) \in E(G)\) is not preserved in \(NG^2_r\). Now we have to prove that \( G - (u,v) \) is still bi-connected. Let \( G' = G - (u,v) \) is not bi-connected and there must be an articulation point \( w \). So, \( G' - w \) is not connected. There might be three kinds of cases in choosing \( w \):

Case 1: If \( w \not\in x,y \) and \( w \not\in u,v \) then \( G' - w \) has \(<u,x,v>\) and \(<u,y,v>\) paths.

Case 2: If \( w \in x,y \), then \( G' - w \) has at least one of the paths from \(<u,x,v>\) and \(<u,y,v>\). Such as, if \( w = x \), then \(<u,y,v>\) still exists. Alternately, if \( w = y \), then \(<u,x,v>\) still exists.

Case 3: If \( w \in u,v \) then all vertices in \( G' - w \) are in same component. So, \( u \) and \( v \) are always connected. Therefore, there is no such articulation point \( w \), which contradicts our assumption.

So, \( G' = G - (u,v) \) is bi-connected and \(NG^2_r\) ensures bi-connectivity. \[ \square \]
3.2.3 Topology with a desired degree of connectivity

In this part, we incorporate flexibility to the topology control algorithm by defining a topology with a desired degree of vertex connectivity. We define this flexible topology for the given topology $G = (V, E)$ as $NG_r^k = (V, E_k)$, where $k$ is the degree of vertex connectivity which indicates that the network will still remain connected despite the failure of any set of less than $k$ nodes. The network designer can choose any planar or fault tolerant topology as per desired requirements by tuning the parameter $k$. Now we will propose an algorithm for generating the topology $NG_r^k$ from the topology $NG_r^1$.

We shall also prove that this flexible topology will be power efficient by maintaining a bound of constant factor $k$ in Power Stretch Factor compared to that of the original topology.

3.2.4 $NG_r^k$ topology construction

For each node $v$ in $N_{r_{em}}(s)$, this algorithm will check the number of nodes falling in the $r$-neighborhood cover set $\zeta_r(s,v)$ between $s$ and $v$. If the cover set contains less than $k$ node(s), then $v$ is included as a neighbor of $s$ in $NG_r^k$. Otherwise, $v$ is not included as a neighbor of $s$ in $NG_r^k$ because it indicates that there are at least $(k - 1)$ node(s) that could be used as relay to communicate between $s$ and $v$. This at least $(k - 1)$ node(s) will provide an additional or back-up vertex-disjoint path that ensures the topology will remain connected in case of $(k - 1)$ node(s) failure between $s$ and $v$.

Thus, when $k = 1$, the algorithm will generate the planar, singly-connected topology $NG_r^1$. $NG_r^k$ in a way represents a set of planar, singly-connected topologies for different values of $r$. For $k \geq 2$, it will generate a range of topologies which will provides fault tolerance when at most $k - 1$ node(s) will go down.

**Algorithm 3 $NG_r^k$ TOPOLOGY CONSTRUCTION**

1: for each $v$ in $N_{r_{em}}(s)$ do
2:     if $|\zeta_r(s,v)| < k$ then
3:         $N_{r_{em}}(s) = N_{r_{em}}(s) \cup \{v\}$
4:     end if
5: end for

**Theorem 3.2.4.** The power stretch factor of $NG_r^k$ is bounded by a
factor of \( k \) compared to that of \( NG_r^1 \).

Proof. Let \( \rho_{NG_r^1}(u, v) \) be the least energy consumed by all paths connecting nodes \( u \) and \( v \) in \( G \). The path connecting \( u, v \) and consuming the energy \( \rho_{NG_r^k}(u, v) \) is called the least energy path in \( NG_r^k \) for \( u \) and \( v \). The power stretch factor of a topology \( NG_r^k \) with respect to \( NG_r^1 \) is then defined as:

\[
\rho_{NG_r^k|NG_r^1} = \max_{(u, v) \in V} \frac{\rho_{NG_r^k}(u, v)}{\rho_{NG_r^1}(u, v)}
\]

As the power needed to support a link between \( u, v \) is \( ||uv||^\beta \) [20], where \( ||uv|| \) is the distance between \( u \) and \( v \), \( \beta \) is a constant between dependent on the wireless transmission environment. Now, by definition, \( v \) will be in the neighbor list of \( u \), or alternatively a link \( uv \) exists between \( u \) and \( v \) in \( NG_r^k \), if there are less than \( k \) nodes in the \( r \)–neighborhood cover set of \( (u, v) \). So, there will be up to \( k \) number of links between \( u \) and \( v \) in this topology compared to that of \( NG_r^1 \). Now the power needed to support \( k \) links between \( u \) and \( v \) will be \( k||uv||^\beta \). So, it is obvious that the power stretch factor of \( NG_r^k \) is bounded by a factor of \( k \) compared to that of \( NG_r^1 \).

\[
\square
\]

### 3.2.5 Sample Topologies after running TC

For simulation environment, we deployed 100 nodes with a transmission range of 175m in an area of 625 \( \times \) 625m. Figure 3.2 shows the initial topology scenerio which shows the links between all neighbor nodes which are in the transmission range of each other. After running the above algorithm, we get a topology with a desired degree of connectivity, that is \( NG_r^k \). Figures from 3.3 to 3.6 shows the topologies generated in the scenarios where \( k = 1 \) was considered for three different values of \( r \). All these topologies are planar and doesn’t provide any alternate paths between all node pairs.

Figures from 3.4 to 3.7 shows the topologies generated in the scenarios where \( k = 2 \) was considered for three different values of \( r \). That is all these topologies provide a desired degree of connectivity by maintaining alternate paths between all node pairs.
Figure 3.2: Initial Topology, (N=100, TX Range=175m)

Figure 3.3: Final Topology (k=1, r=0.0)
Figure 3.4: Final Topology (k=2, r=0.0)

Figure 3.5: Final Topology (k=3, r=0.0)
Figure 3.6: Final Topology (k=1, r=1.0)

Figure 3.7: Final Topology (k=2, r=1.0)
3.2.6 Validating the parameter $k$ in Sample Topologies

Now, we find out the vertex connectivity of the sample $NC_k^r$ topologies generated above using a software tool and check if it matches with the value of $k$. The software tool we use is Wolfram Mathematica [21], which is basically a mathematical symbolic computation program widely used in many scientific, engineering, mathematical, and computing fields. It provides the tools for visualizing and analyzing directed and undirected graphs.

We considered the topologies in figures 3.3 and 3.4 where the values of $k$ were 1 and 2 respectively. As input parameters for a test, we provided the list of edges of the topology. Every edge was represented by its two nodes numbers separated by an arrow sign whereas every node number was distinctly numbered starting from 1 up to the number of nodes. We ran the VertexConnectivity tool of the Wolfram Mathematica, which corresponds to the degree of vertex connectivity parameter $k$ in our study, as defined in 2.2.5. We found that results from the tool as 1 and 2 respectively for the given topologies which were the same as their corresponding values of $k$. We provide the screen-shots of the program tool while finding out the vertex connectivity as figures 3.9 and 3.10 respectively.
Figure 3.9: Finding out $k$ in Final Topology ( $k=1, N=100$, $TX=175m$, $r=1.0$)
Figure 3.10: Finding out k in Final Topology (k=2, N=100, TX=175m, 
r=1.0)
Chapter 4

Analytical Modeling for Performance Metrics

In this section, we develop mathematical expressions for determining Sparseness, Average Node Degree, and Topology Size for our proposed topology in the previous section NG_r^2.

4.1 Performance Metrics

We start with the metric Sparseness. The number of links removed from the initial topology determines the Sparseness of a r−neighborhood graph. Here we focus on per node sparseness which is the average fraction of links removed from a node’s neighborhood. When a link is removed, the node at the other end of the link also gets removed from a node’s neighbor set. Thus, mathematically, (per node) sparseness is defined as follows:

\[
\text{Sparseness} = \frac{\text{Average number of nodes removed}}{\text{Number of nodes in a node’s TX area}} \quad (4.1)
\]

Let us consider a multi-hop wireless network with n nodes uniformly distributed over a rectangular region with area A. The average node density is \( \mu = \frac{n}{A} \). The maximum transmission radius of each node is R, which is assumed to be the same for all nodes. Let us observe an arbitrary node s within the deployment area. As the node distribution is assumed to be
uniform, it is easy to determine the number of nodes present in a node’s transmission (TX) area if the node density is known apriori. To see how, let us observe an arbitrary node \( s \) within the deployment area. The average number of nodes located in the communication region of node \( s \) is:

\[
N_R = \text{Node density} \times \text{Transmission area} = \mu \times \pi R^2
\]

![Figure 4.1: Illustrating circular strip at distance \( x \)](image)

Let \( P_N(x) \) be the probability that there exists a neighbor \( t \) at distance \( x \) from \( s \). Clearly, \( P_N(x) = 0 \) for \( x > R \).

For \( x \leq R \), at first we consider a small area strip defined by \( dx \) at the perimeter of the circle with radius \( x \) and centered at \( s \) as shown in figure 4.1. Also we consider a small angle \( d\theta \) measured from an arbitrary but fixed axis. The length of the arc \( l = x d\theta \) and the area of the small region \( dA \) within this small strip can be approximated as \( dA = l dx = x dx d\theta \). Therefore, the area of the entire small strip denoted by \( A_{\text{strip}} \) becomes,

\[
A_{\text{strip}} = \int_0^{2\pi} dA = \int_0^{2\pi} l dx = \int_0^{2\pi} x dx d\theta = 2\pi x dx
\]

Thus \( P_N(x) \) becomes:

\[
P_N(x) = \text{Area of the strip} \times \text{Node density} = A_{\text{strip}} \times \mu = 2\pi x dx \times \mu = 2\pi \mu x dx
\]  

(4.2)

Let \( P_C(x) \) be the probability that there exists at least 2 nodes in \( \zeta_r(s, t) \), the \( r \)-neighborhood cover set between node pair \((s, t)\). The distance \( x \) between \( s \) and \( t \) plays an important role in determining the value of \( P_C(x) \).
For large $x$, the size of the $r$–neighborhood region is also large and the probability of a node’s existence within this $r$–neighborhood region also becomes large.

The probability $P_E(x)$ of eliminating any node $t$ from the neighbor set of $s$ is the probability that there exists a neighbor $t$ at distance $x$ from $s$, and there are 2 nodes in the $r$–neighborhood cover set between $(s,t)$. So $P_E(x)$ is:

$$P_E(x) = P_N(x) \times P_C(x) = 2\pi \mu x dx \times P_C(x)$$

The expected number of neighbors eliminated by $s$ from its neighbor set is found by integrating $P_E(x)$ over the transmission radius $R$ within which $s$ possibly can communicate:

$$T_e = \int_0^R 2\pi \mu x \times P_C(x) dx$$  \hspace{1cm} (4.3)

If we divide the expected number of eliminated neighbors by the expected number of nodes in a node’s communication area, we get the expected fraction of neighbors that are eliminated from a node’s neighbor set, which we define as Sparseness, $F_{\text{elim}}$.

$$F_{\text{elim}} = \frac{T_e}{N_R} = \frac{T_e}{\pi \mu R^2}$$  \hspace{1cm} (4.4)

### 4.1.1 Sparseness

To derive the exact expression for Sparseness, we need to determine the value of $P_C(x)$, which is the probability that there exists at least 2 nodes in the $r$–neighborhood cover set $\zeta_r(s,t)$ between node pair $(s,t)$.

The probability that a node is placed in the $r$–neighborhood region cover set $\zeta_r(s,t)$ between node pair $(s,t)$ is,

$$P_T = \frac{\text{Pruning Area}}{\text{Total Area}}$$

From the equation 2.1, the pruning area of the $r$–neighborhood region is,

$$A_{NR_r(s,t)} = \gamma ||st||^2$$
Therefore, $P_T$ becomes,

$$P_T = \frac{\gamma x^2}{A} \quad (4.5)$$

Using binomial distribution, the probability $P_m(\zeta_r(s,t))$ that exactly $m$ nodes are in $\zeta_r(s,t)$ is:

$$P_m(\zeta_r(s,t)) = \binom{n-2}{m} \frac{P_T^m}{m!} \times (1 - P_T)^{n-2-m}$$

Here, $n-2$ is used rather than $n$ because we exclude $s$ and $t$. For large values of $n$ and small values of $P_T$, the binomial distribution can be approximated using Poisson distribution [22] with mean $nP_T$:

$$P_m(\zeta_r(s,t)) = \frac{(nP_T)^m \times e^{-nP_T}}{m!}$$

So, the probability that there exists at least 2 nodes in $\zeta_r(s,t)$ is,

$$P_C(x) = \sum_{j=2}^{n} P_m(\zeta_r(s,t)) = \sum_{j=2}^{n} \frac{(nP_T)^m \times e^{-nP_T}}{m!}$$

$$= e^{-nP_T} \left( \sum_{j=2}^{\infty} \frac{(nP_T)^m}{m!} \right)$$

$$= e^{-nP_T} \left( \sum_{j=0}^{\infty} \frac{(nP_T)^m}{m!} - 1 - nP_T \right)$$

Now, the Maclaurin series expansion for exponential function is,

$$e^x = \sum_{k=0}^{\infty} \frac{(x)^k}{k!}$$

Using this series, the value of $P_C(x)$ becomes,

$$P_C(x) = e^{-nP_T} \left( e^{nP_T} - 1 - nP_T \right)$$

$$= 1 - e^{-nP_T} - nP_T e^{-nP_T} \quad (4.6)$$

Now, we find out the value of $nP_T$ using equation 4.5,

$$nP_T = \frac{n\gamma x^2}{A} = \gamma x^2 \frac{\mu}{A} \quad \text{[Since, } \mu = \frac{n}{A}]$$

34
Substituting the value of $nP_T$ in the equation 4.6,

$$P_C(x) = 1 - e^{-\gamma \mu x^2} - \gamma \mu x^2 e^{-\gamma \mu x^2}$$

By replacing $P_C(x)$ in the equation 4.3, we get:

$$T_e = \int_0^R 2\pi \mu x dx - \int_0^R 2\pi \mu x e^{-\gamma \mu x^2} dx - \int_0^R 2\pi \gamma \mu^2 x^3 e^{-\gamma \mu x^2} dx$$

$$= \pi \mu R^2 + I_1 + I_2 \tag{4.7}$$

To evaluate $I_1$, let, $\gamma \mu x^2 = z$. Therefore, $2xdx = \frac{dz}{\gamma \mu}$. So,

$$I_1 = -\frac{\pi}{\gamma} \int_0^{\gamma \mu R^2} e^{-z} dz$$

$$= \frac{\pi}{\gamma} \left[ e^{-z} \right]_0^{\gamma \mu R^2}$$

$$= \frac{\pi}{\gamma} \left( e^{-\gamma \mu R^2} - 1 \right)$$

To evaluate $I_2$, let, $-\gamma \mu x^2 = t$. Therefore, $-2\pi \mu x dx = dt$. So,

$$I_2 = -\frac{\pi}{\gamma} \int_0^{-\gamma \mu R^2} te^t dt$$

We now use the following commonly used equation for integration by parts:

$$\int xe^x dx = xe^x - e^x + Constant$$

Using this formula, we get,

$$I_2 = -\frac{\pi}{\gamma} \left[ te^t - e^t \right]_0^{-\gamma \mu R^2}$$

$$= \frac{\pi}{\gamma} \left( e^{-\gamma \mu R^2} + \gamma \mu R^2 e^{-\gamma \mu R^2} - 1 \right)$$

Using the values of $I_1$ and $I_2$ in the equation 4.7, we get,

$$T_e = \pi \mu R^2 + \frac{\pi}{\gamma} \left( \gamma \mu R^2 e^{-\gamma \mu R^2} - 2e^{-\gamma \mu R^2} - 2 \right)$$

Therefore, from 4.4, the expected fraction of neighbors that are elimi-
nated from a node’s set is:

\[
S\text{parsity} = F_{\text{elim}} = \frac{\pi \mu R^2}{\gamma} \left( \frac{\gamma \mu R^2 e^{-\gamma \mu R^2} - 2 e^{-\gamma \mu R^2} - 2}{\gamma \mu R^2} \right)
\]

\[
= \gamma \mu R^2 + \gamma \mu R^2 e^{-\gamma \mu R^2} - 2 e^{-\gamma \mu R^2} - 2
\]

\[
= (1 + e^{-\gamma \mu R^2}) \left( 1 - \frac{2}{\gamma \mu R^2} \right)
\]

(4.8)

### 4.1.2 Average Node Degree

The **Average Node Degree** \((d_{\text{avg}})\) of the topology is the expected number of neighbors retained after pruning. Therefore, if we subtract \(T_\varepsilon\) from the expected number of neighbors within \(s\)'s communication range then we get \(d_{\text{avg}}\):

\[
d_{\text{avg}} = \pi \mu R^2 - T_\varepsilon
\]

\[
= \frac{\pi}{\gamma} \left( \gamma \mu R^2 e^{-\gamma \mu R^2} - 2 e^{-\gamma \mu R^2} - 2 \right)
\]

(4.9)

### 4.1.3 Topology Size

Finally, if we multiply \(d_{\text{avg}}\) by the total number of nodes \(n\), we obtain twice the number of links retained after running the topology control algorithm (an edge contributes to exactly two node’s degree counts). Thus, the **Topology Size** of the topology \(NG_r^k\) becomes,

\[
S(NG_r^k) = \frac{n \times d_{\text{avg}}}{2}
\]

Substituting value of **Average Node Degree** from 4.9 we get:

\[
S(NG_r^k) = \frac{\pi n}{2\gamma} \left( \gamma \mu R^2 e^{-\gamma \mu R^2} - 2 e^{-\gamma \mu R^2} - 2 \right)
\]

(4.10)
Chapter 5

Simulation and Performance Evaluation

5.1 Performance evaluation

In the previous chapters, we augmented the algorithm for $r$-neighborhood graph construction to provide fault tolerance and flexibility. We also derived the mathematical models for estimating few key performance metrics of the topology such as Sparseness, Average Node degree, and Topology Size. In this chapter, we will form a simulation environment and run simulations for these performance metrics. We will also derive results from both analytical expressions for the same settings. From these results, we will be able to present a comprehensive comparison between these two and an insight on how different network and transceiver parameters effect these performance metrics.

5.2 Simulation Environment

To evaluate the performance, we simulate randomly deployed networks of 100, 300, 500 nodes uniformly distributed over a 625 m $\times$ 625 m square region. We vary the transmission range of a node between 125m to 225m. We also consider 3 different values of $r = 0, 0.50, 1$. We ran simulations for different scenarios for 2 different values of the tunable parameter $k$. For the simulations, firstly we randomly generated set of a scenarios as per the simulation environment described above, each of which represents an initial
topology. Next we wrote a Java code for generating the final topology as per the topology control algorithm described in the algorithm chapter. That is, after running the Java code on an initial topology, we found the remaining neighbor set of all the nodes of the topology after the elimination of nodes from the initial neighbor following topology control. From these initial set and the remaining set of neighbors, we found out the values of sparseness, average node degree, and topology size of the topology.

5.3 Simulation Results

In this section we show the experimental results for different performance metrics such as sparseness, average node degree, and topology size of the topology. We also present a comprehensive comparison between the results from analytical models and simulations. Let us start with Sparseness for example.

5.4 Effect on Sparseness

For each of the above scenarios, firstly we compute the Sparseness, that is the fraction of eliminated nodes after running the topology control algorithm described in the previous section to obtain simulation results. Secondly, we calculate the values of Sparseness from the analytical expression derived in the previous section under the same settings of parameters. These results we call as analytical results. Now we compare the two sets of results side by side using graphs. The accuracy of our analytical models can be seen from the two graphs as they are very close to each other.

5.4.1 Effect of Node Density on Sparseness

We focus on how node density affects the Sparseness of the topology. Node density were varied by varying number of nodes between 100 and 500 while keeping the deployment region constant at 625m × 625m square area. These provide a change in the node density \( \mu \) between 0.000256 to 0.00128. Figures 5.1 to 5.4 shows the result for three different values of \( r \) and \( k \). For all the combinations of two different settings of \( r \) and \( k \), we plot three curves under three transmission ranges 225m, 175m and 125m, respectively in each plot. A higher fraction of neighbors is eliminated in more dense networks for all
transmission ranges. With larger node densities, it is highly probable that at least one node exists in the r-neighborhood region of a link, and the link gets pruned by the algorithm.

All figures show that for all scenarios, the results of analytical expressions are very close to the simulation results; the difference is very small, maximal being around 4.5 percent. Thus, the analytical models are effectively able to capture the generic pattern of the simulation results. The small inaccuracy arises from the nodes located close to the boundaries of the deployment region, for which the communication area is restricted, and thus they have fewer neighbors. With larger transmission ranges the effect also becomes larger. We ignored this “boundary effects” to simplify the analytical models.

5.4.2 Effect of Transmission Range on Sparseness

To see the effect of transmission range on Sparseness, we measure the Sparseness of a topology for different values of transmission range. The transmission range is varied between 125m to 225m with an increment of 25m at each step. A higher value of transmission range means more number of nodes are expected to be in the neighborhood lists initially. So, after running the topology control algorithm, more number of nodes are expected to be eliminated from a node’s neighborhood list. Alternately, the number of links removed will be few as less links appear to be in a smaller value of transmission range of a node. Figures 5.5 and 5.6 shows the result for two different values of k. In each figure, we have plotted three curves for three different values of number of nodes, N=100,300 and 500. Measurements from both simulation experiments and analytical expressions are plotted in the same graph for a fair comparison. When transmission range is increased, Sparseness exponentially increases at the beginning and linearly increases at the end. Also, as expected, the Sparseness increases when we increase N values.

5.4.3 Effect of r on Sparseness

The parameter r here defines the neighborhood or the proximity region. As the value of r will increase, the size of the r-neighborhood region will increase. So, more nodes are expected to fall in the neighborhood region of a node pair. Hence as we run topology control algorithm, more nodes
are likely to be eliminated from the neighbors list of the nodes. So, the Sparseness increases when we increase the values of \( r \). We can see our experimental results demonstrates the expected behavior. The fraction of eliminated neighbors is much higher in both figure 5.7 and figure 5.8 as the \( r \) value increases from 0 to 1. We also notice as the TX value raises from figure 5.7 to figure 5.8, and the change is slow.

### 5.4.4 Effect of \( k \) on Sparseness

The tunable parameter \( k \) here stands for the number of nodes in the of the \( r \)-neighborhood cover set between every node pair that would provide alternate paths between the pair. Thus it will ensure the network will still remain connected despite the failure of any set of less than \( k \) nodes. So an increase of the values of \( k \) would ensure less nodes getting eliminated from a node’s neighbors list after running the algorithm. So, the Sparseness would decrease when we increase the values of \( k \). The figures 5.9 and 5.10 here depict the same scenario.

![Deployment Area= 625m X 625m](image)

Figure 5.1: Effect of Node density on Sparseness, \((k=1, r=0.0)\)
Figure 5.2: Effect of Node density on Sparseness, (k=2, r=0.0)

Figure 5.3: Effect of Node density on Sparseness, (k=1, r=1.0)
Figure 5.4: Effect of Node density on Sparseness, (k=2, r=1.0)

Figure 5.5: Effect of Transmission Radius on Sparseness, (k=1, r=1.0)
Figure 5.6: Effect of Transmission Radius on Sparseness, (k=2, r=1.0)

Figure 5.7: Effect of r on Sparseness, (N=500, TX Range=125m)
Figure 5.8: Effect of $r$ on Sparseness, ($N=500$, TX Range=225m)

Figure 5.9: Effect of $k$ on Sparseness (TX Range=175m, $r=0.5$)
5.5 Effect on Average Node Degree and the Topology Size

Now we evaluate Average Node Degree, and Topology Size for different settings of Node density $\mu$, Transmission Range $R$, Neighborhood region parameter $r$ and degree of vertex connectivity parameter $k$. Moreover, as the Topology Size can be derived by multiplying Average Node Degree with a constant $n/2$ as shown previously, so the effect of these parameters on these metrics will be the similar. We choose to show plots for Topology Size and explain the effects.

5.5.1 Effect of Node Density

With the increase of node density, degrees of the nodes before and after running the algorithm will be more and hence the size of the topology will tend to increase. Figure 5.11 depicts the same scenario, where the Topology Size calculated from it’s analytical expression were plotted against corresponding node densities for a topology with TX Range= 175m and $r=0.5$
in three different plots representing three different values of $k$.

5.5.2 Effect of $r$

Next we focus on how Average Node Degree, and Topology Size are affected with different values of $r$. Figure 5.13 shows the effect where for three different values of $k$, the Topology Size calculated from its analytical expression for a topology with $N=300$ nodes and TX Range $= 175m$ were plotted against corresponding values of $r$. With the increase of $r$, the Average Node Degree, and Topology Size decays exponentially indicating that the $r-$neighborhood graph becomes sparser when we increase $r$. Also $r-$neighborhood graphs with smaller values of $k$ are sparser for all values of $r$.

5.5.3 Effect of Transmission Range

As the Transmission Range of a node will increase, there will be more nodes in a node's neighborhood region. Hence degrees of the nodes before and after running the algorithm will be more and the topology will be larger in size as well. From figure 5.12, we see a similar effect. For different plots, the Topology Size plotted along the X-axis will span through a huge range making it tough for the viewer to visualize the change in the given paper size. So, we choose to show only one plot to understand the change. The topology has $k=1$, $N=100$ nodes and $r = 0$. As expected, the plot shows that the Topology Size increases exponentially with the increase of TX Range.

5.5.4 Effect of $k$

As the increase of the value of $k$ ensures less nodes getting eliminated from a node's neighbors list after running the algorithm, so it is expected that the Average Node Degree, and Topology Size will increase with the increase of values of $k$. Plotting with the help of bar-chart, we find similar effects too in figure 5.14 for three different values of $r$. Here, $N=100$ nodes and TX Range $=175m$, the topology enlarges in size with the increasing values of $k$.
Deployment Area= 625m X 625m

Figure 5.11: Effect of Node Density on Topology Size (TXR=175m, r=0.5)

Deployment Area= 625m X 625m

Figure 5.12: Effect of TX Range on Topology Size (N=100, r=0.0, k=1)
Figure 5.13: Effect of $r$ on Size of the Topology ($N=300$, TX Range=175m)

Figure 5.14: Effect of $k$ on Size of the Topology using bar-chart ($N=300$, TX Range=175m)
5.6 Mathematical analysis of effect of the parameters on Performance Metrics

In this section we mathematically analyze the effect of transceiver parameter $R$ (the transmission range) and two other network parameters $n$ (the number of nodes), $r$ (the parameter for defining $r$-neighborhood graph). In particular, we try to verify the effect of these network and transceiver parameters on these performance metrics on our simulation and analytical plots.

At first let us start from equation 4.4 where the Sparseness is mathematically defined:

$$Sparseness = \left(1 + e^{-\gamma \mu R^2}\right) \left(1 - \frac{2}{\gamma \mu R^2}\right)$$

To find the effect of three parameters $R$, $n$, and $r$ on $F_{elim}$, we rewrite the equation as,

$$F(\phi) = \left(1 + \frac{1}{e^\phi}\right) \left(1 - \frac{2}{\phi}\right)$$  \hspace{1cm} (5.1)

where $\phi = \gamma \mu R^2$. As all three parameters ($R$, $n$, and $r$) are greater than zero, finding out the effect of $\phi$ on $F(\phi)$ will suffice.

Without loss of generality, let us assume that $F(\phi_1) > F(\phi_2)$, for any two positive real numbers $\phi_1$ and $\phi_2$.

So, we have,

$$\left(1 + \frac{1}{e^{\phi_1}}\right) \left(1 - \frac{2}{\phi_1}\right) > \left(1 + \frac{1}{e^{\phi_2}}\right) \left(1 - \frac{2}{\phi_2}\right)$$

Taking logarithms on both sides,

$$\ln \left(1 + \frac{1}{e^{\phi_1}}\right) + \ln \left(1 - \frac{2}{\phi_1}\right) > \ln \left(1 + \frac{1}{e^{\phi_2}}\right) + \ln \left(1 - \frac{2}{\phi_2}\right)$$

Now, $\phi$ is a multiple of $\gamma$, $\mu$ and $R^2$ which is supposed to be a fairly large value for our scenario. We have varied the transmission range $R$ from 125m to 225m. $\gamma$ is also a positive value, which depends on value of $r$. The value of $r$ was varied between 0 and 1, which provides the values of $\gamma$ between 0.785 and 1.228. Node density $\mu$ were varied by varying number of nodes.
between 100 and 500 while keeping the deployment region constant at 625m \( \times \) 625m square area. These provide a change in the node density \( \mu \) between 0.000256 to 0.00128.

So, the maximum value of \( \phi \) in our scenarios,

\[
\phi_{\text{max}} = 1.228 \times 0.00128 \times 225^2 = 79.57
\]
\[
N_{\text{ow}}, e^{-79.57} = 4.9 \times 10^{-35}
\]

The minimum value of \( \phi \) in our scenarios would be,

\[
\phi_{\text{min}} = 1.228 \times 0.000256 \times 125^2 = 4.912
\]
\[
N_{\text{ow}}, e^{-4.912} = 0.0067
\]

So, \( \frac{1}{\phi} \) will be a very small value close to 0 in our case. As the purpose of this mathematical analysis is to verify the effect of the network and transceiver parameters on these performance metrics in our simulation and analysis plots, we can consider \( \frac{1}{\phi} \) as equal to 0 in both sides, that leaves us with,

\[
\ln \left(1 - \frac{2}{\phi_1}\right) > \ln \left(1 - \frac{2}{\phi_2}\right)
\]

or,

\[
\left(1 - \frac{2}{\phi_1}\right) > \left(1 - \frac{2}{\phi_2}\right)
\]

(5.2)

Subtracting 1 from both the sides and then dividing both the sides by -2,

\[
\frac{1}{\phi_1} < \frac{1}{\phi_2}
\]

So, \( \phi_1 > \phi_2 \)

(5.3)

We see that the \( F_{\text{elim}} \) is increasing as we increase the value of the parameters.

So, we can conclude that, the \textit{Sparseness} increases as we increase the TX Range, Number of Nodes or \( r \), which are similar to our observations from plots from simulation and analytical models. Similar observations can be hold for the \textit{Average Node Degree}, and \textit{Topology Size}. So, we see that effect of
the network and transceiver parameters on these performance metrics follows the same behavior as those obtained from the plots from simulation and analytical models. These provides a validation for our developed analytical models and conducted simulation tests. Moreover, the above quantitative study provides an insight for the network designers about how to choose different network and transceiver parameters for his/her desired network without running any simulations or prior to the deployment of the real systems.
Chapter 6

Conclusion and Future Works

6.1 Conclusion

Topology control in wireless multi-hop networks is a open research problem. The $r-$neighborhood graphs constitute an important class of proximity graphs derived from topology control algorithms. Moreover, as topology control algorithms reduces the number of links in the network, the topology thus derived is more susceptible to node failures. Thus providing fault tolerance to the topology in case of failure of node(s) is very significant for such proximity graphs.

In our study, we designed an algorithm for constructing a new fault tolerant topology for $r-$neighborhood graphs that will maintain network connectivity in case of a failure of node(s). Then we provided flexibility in topology control by providing tunable parameter(s) to generate two sets of topologies— (a) planar, and (b) fault tolerant. By tuning the parameter(s), the network designer will have the ability to switch between the two as per desired requirements. We developed analytical models for few key performance metrics such as Sparseness, Average Node Degree, and Topology Size. The accuracy of the analytical models were validated through extensive simulations. We also found out mathematically that the effects of different network and transceiver parameters on these metrics follows the same behavior as those obtained from the simulation and analysis plots. Thus, we provided the network designer an insight to use these parameters
without any simulation or prior to real system deployment.

6.2 Future Works

A limitation of our paper is that we considered the values of degree of vertex connectivity $k$ from 1 to 2 in analytical modeling and simulations. A future scope of our research can be working with the greater values of $k$. Moreover, we considered the redundancy technique for guaranteeing availability to provide fault tolerance. Like the prior works which used this approach, we ensured alternate paths between nodes to maintaining network connectivity in case of nodes failures. However, other techniques such as Fault masking, Reconfiguration could also be considered in future. Moreover, extending these topology control algorithms to provide fault tolerance would mean an additional increase in the complexity regarding the activating or deactivating of the nodes, the network coverage characteristics, nodes deployment etc, which we did not consider in our scope. We keep that as a future challenge to our works.

Moreover, we would like to work with other significant graphs generated from topology control algorithms such as Delta Graphs, Yao Graph, Delaunay triangulation etc to provide fault tolerance and flexibility. We also plan to find analytical models for the performance metrics of such topologies.
Chapter 7

Appendix

7.0.1 Graph

A graph $G$ is a triple consisting of a vertex set $V(G)$, an edge set $E(G)$, and a relation that associates with each edge two vertices (not necessarily distinct) called its endpoints.

7.0.2 Graph Model of a Network

By the term node, in this paper we mean the network stations (example can be a personal computer, laptop computer, cell phone i.e. a device with computational and networking capability), which can perform some network activity. A network topology can mathematically be modeled as a graph. The set of all the networking capable devices or the stations be the vertices set, and the set of all the interconnections between the stations be the edge set. In an ad hoc wireless network, all the network stations x physically in the radio range of a network station a will cause the corresponding topology graph to have undirected edge $(a, x)$.

7.0.3 Degree of a Node

The degree of a Node of a graph is the number of edges incident to the vertex, with loops counted twice.

7.0.4 Connected Graph

A graph is connected when there is a path between every pair of vertices. In a connected graph, there are no unreachable vertices. A graph that is not
connected is disconnected.

7.0.5 Planarity of a Graph

A planar graph is a graph that can be embedded in the plane; that is, it can be drawn on the plane in such a way that its edges intersect only at their endpoints. In other words, it can be drawn in such a way that no edges cross each other.

7.0.6 Gabriel Graph

The Gabriel graph of a set $S$ of points in the Euclidean plane expresses one notion of proximity or nearness of those points. Formally, it is the graph with vertex set $S$ in which any points $P$ and $Q$ in $S$ are adjacent precisely if they are distinct and the closed disc of which line segment $PQ$ is a diameter contains no other elements of $S$. Gabriel graphs naturally generalize to higher dimensions, with the empty disks replaced by empty closed balls. Gabriel graphs are named after K. R. Gabriel, who introduced them in a paper with R. R. Sokal in 1969.

7.0.7 Relative Neighborhood Graph

The relative neighborhood graph (RNG) is an undirected graph defined on a set of points in the Euclidean plane by connecting two points $p$ and $q$ by an edge whenever there does not exist a third point $r$ that is closer to both $p$ and $q$ than they are to each other. This graph was proposed by Godfried Toussaint in 1980 as a way of defining a structure from a set of points that would match human perceptions of the shape of the set.

7.0.8 Minimum Spanning Tree

Given a connected, undirected graph, a spanning tree of that graph is a subgraph that is a tree and connects all the vertices together. A single graph can have many different spanning trees. We can also assign a weight to each edge, which is a number representing how unfavorable it is, and use this to assign a weight to a spanning tree by computing the sum of the weights of the edges in that spanning tree. A minimum spanning tree (MST) or minimum weight spanning tree is then a spanning tree with weight less than or equal to the weight of every other spanning tree.
7.0.9 Binomial Distribution

In probability theory and statistics, the binomial distribution with parameters \( n \) and \( p \) is the discrete probability distribution of the number of successes in a sequence of \( n \) independent experiments, each asking a yes-no question, and each with its own boolean-valued outcome: a random variable containing single bit of information: success/yes/true/one (with probability \( p \)) or failure/no/false/zero (with probability \( q = 1 - p \)).

7.0.10 Poisson Distribution

Poisson distribution is the probability distribution of the number of successes in a sequence of \( n \) independent yes/no experiments with success probabilities \( p_1, p_2, \ldots, p_n \). The ordinary binomial distribution is a special case of the Poisson binomial distribution, when all success probabilities are the same, that is \( p_1 = p_2 = \cdots = p_n \). An important property of the Poisson distribution is that, it can be used to approximate a binomial distribution when the binomial parameter \( n \) is large and \( p \) is small.
References


[17] Z. Shen, Y. Chang, C. Cui, and X. Zhang, “A fault-tolerant and minimum-energy path-preserving topology control algorithm for wire-
less multi-hop networks,” in International Conference on Computational and Information Science (CIS), Xi’an, China, December 15-19, 2005, pp. 864–869.


