The Effect of Topology Control for Wireless Multi-Hop Networks

by

Pragasen Mudali
(B.Sc Hons. Computer Science)

Submitted to the Faculty of Science and Agriculture, in fulfillment of the requirements for the degree of Master of Science in Computer Science in the Department of Computer Science at the University of Zululand

Supervisor: Professor M.O. Adigun

2007

DECLARATION

This dissertation represents the author's own work and has not been submitted in any form to another University for degree purposes. All sources used in the dissertation have been duly acknowledged.

Signature

DEDICATION

To the Mudali family

ACKNOWLEDGEMENTS

I would like to begin by thanking my family for their unwavering love and support during the past six years. They have supported me through the inconveniences of early classes, late classes, Saturday classes and countless other events that have had to take precedence over their schedules and activities. I am forever indebted by their sacrifices for me.

Many thanks also go to my friend and colleague Thulani Nyandeni whose drive and passion got me through the tough times. His work as my unofficial supervisor contributed immensely to the successful and early completion of this work.

My gratitude extends to my supervisor Prof. M.O. Adigun for his expert guidance and for allowing me the necessary freedom to complete this work. I would also like to thank him for initiating the activities that eventually led to my stay at the Meraka Institute. Many thanks must also go to Dr. Justice who provided the motivation for the early completion of this work.

I am forever thankful for the opportunity to work with the Wireless Africa project currently being undertaken at the Meraka Institute, as this was my first experience of corporate culture. Many thanks to Kobus Roux, David Johnson, Karel Mathee, Lawrence Mboweni and Sonja Hall for making my integration into Wireless Africa's activities as seamless as possible. Particular thanks go to Karel Mathee for granting me the opportunity to participate in the installation of a wireless mesh node in Mamelodi, and to Lawrence Mboweni, with whom I shared an office during my stay, for patiently answering my questions and introducing me to the world of Linux. Lawrence's knowledge and insight has contributed immeasurably to the successful completion of this work. Thanks are also reserved for Prof. Malherbe for his contribution to my proposal during my stay at the Institute.

My thanks also go to my fellow research colleagues that contributed to the completion of this work, particularly Sihle Sibiya for his help with Java.

Special thanks go to Telkom, whose sponsorship of me via their Centre of Excellence program enabled me to meet my financial obligations, thus allowing me to focus my efforts solely on the completion of this project.

Finally, I would like to thank GOD for ensuring that I had the strength, fortitude and the necessary forms of support for the entire duration of this work.

TABLE OF CONTENTS

1 INTRODUCTION AND BACKGROUND 1
1.1 Preamble....................1
1.2 Statement of the Problem....................6
1.3 Rationale for the Research....................7
1.4 Research Questions....................8
1.5 Research Goals and Objectives....................8
1.5.1 Goal....................8
1.5.2 The Objectives....................8
1.6 Methodology....................9
1.6.1 Research Method One....................9
1.6.2 Research Method Two....................10
1.6.3 Research Method Three....................10
1.6.4 The Chosen Method....................11
1.7 Organization of the Dissertation....................12
2 LITERATURE REVIEW 13
2.1 Overview....................13
2.2 The Optimal Node Degree (Number of Neighbours)....................14
2.2.1 Overview....................14
2.2.2 Framework for Analysing Work Related to the Optimal Node Degree (Number of Neighbours)....................14
2.3 Review of Work Related to Determining the Optimal Number of Neighbours in a Wireless Multi-Hop Network....................16
2.4 Summary of Work Related to the Optimal Number of Neighbours....................26
2.5 Topology Control for Wireless Multi-hop Networks....................27
2.5.1 Overview....................27
2.5.2 Framework for Analysing Work Related to Topology Control....................28
2.6 Review of Topology Control Algorithms....................30
2.7 Summary of Work Related to Topology Control....................50
2.8 Summary of the Required Design Criteria for a Topology Control Algorithm for Low Cost, Resource-Constrained Wireless Multi-hop Network Nodes .51
3 PERFORMANCE ANALYSIS OF THE OPTIMAL NUMBER OF NEIGHBOURS IN A STATIC WIRELESS MULTI-HOP NETWORK 53
3.1 Overview....................53
3.2 Simulation Methodology....................53
3.3 Simulation Environment....................55
3.3.1 Physical and Data Link Layer Model....................56
3.3.2 Medium Access Control....................56
3.3.3 Packet Buffering Model....................57
3.3.4 Data Traffic Model....................57
3.3.5 Multi-Hop Network Routing Protocol....................58
3.3.6 Criteria for Choosing Network Topologies....................58
3.4 Simulation and Experiments Results....................59
3.4.1 Experiment 1: Packet Delivery Ratio (PDR)....................60
3.4.2 Experiment 2: Routing Overhead....................63
3.4.3 Experiment 3: End-to-End Delay....................65
3.5 Simulator and Experimental Limitations....................68
3.6 Summary....................69
4 TOKEN-BASED TOPOLOGY CONTROL (TbTC) 71

4.1 Overview 71
4.2 Design Criteria and Assumptions 72
4.2.1 Distributed Algorithm 73
4.2.2 Minimal Computation 73
4.2.3 Node degree information requirements 74
4.2.4 Heterogeneous Transmission Radius 74
4.2.5 Maintenance of network connectivity 75
4.2.6 Synchronicity between neighbours 75
4.3 Proposed Algorithm 75
4.3.1 The Transmit Power Selection and Adjustment Component 76
4.3.2 The Network Connectivity Component 78
4.3.3 The Next Node Selection Component 78
4.4 The Packet Structure of the Token Employed 82
4.5 Performance Analysis of Token-based Topology Control Algorithm 85
4.5.1 Simulation Methodology 85
4.5.2 Simulation Results 87
4.5.2.1 Experiment 1: Packet Delivery Ratio 88
4.5.2.2 Experiment 2: Routing Overhead 90
4.5.2.3 Experiment 3: End-to-End Delay 93
4.5.2.4 Experiment 4: Power Consumption 95
4.5.2.5 Experiment 5: Network Traversal (Hop Count) 96
4.5.3 Limitations of the Simulator and the Experiment 99
4.6 Token-based Topology Control Algorithm Limitations 100
5 CONCLUSION AND FUTURE WORK 103
5.1 Conclusion 103
5.2 Future Work 105
6 BIBLIOGRAPHY 106
7 APPENDIX A – NS-2 Simulation Script for "Perfect" Scenario 113
8 APPENDIX B – NS-2 Simulation Script for Disconnected Scenario 116
9 APPENDIX C – NS-2 Simulation Script for TbTC 119
10 APPENDIX D – Linksys WRT54G Specifications 122

LIST OF FIGURES

Fig. 1.1 – Wireless multi-hop network example 2
Fig. 1.2 – Interdependence of Network Elements 3
Fig. 1.3 – Layout of a typical South African community 4
Fig. 1.4 – Network Topologies proposed in (Cardell-Oliver 2003) 5
Fig. 3.1 – Packet Delivery Ratio (PDR) per network topology 61
Fig. 3.2 – Packet Delivery Ratio versus the average node degree 62
Fig. 3.3 – Routing Overhead per network topology 63
Fig. 3.4 – Routing Overhead versus the average node degree 64
Fig. 3.5 – End-to-End Delay per network topology 67
Fig. 3.6 – End-to-End Delay versus the average node degree 67
Fig. 4.1 – Components of the Token-based Topology Control Algorithm 76
Fig. 4.2 – Token-based Topology Control algorithm 77
Fig. 4.3 – Flowchart depicting the execution of TbTC 79
Fig. 4.4 – Sequence diagram depicting the execution of TbTC 81
Fig. 4.5 – Internet Protocol (IP) Packet Structure 82
Fig. 4.6 – IP Type of Service specification for the token 83
Fig. 4.7 – IP Flags specification for the token 84
Fig. 4.8 – PDRs before and after TbTC 90
Fig. 4.9 – Routing Overhead before and after TbTC 92
Fig. 4.10 – End-to-End Delay before and after TbTC 94
Fig. 4.11 – Average Path Length before and after TbTC 94
Fig. 4.12 – Average Node degree before and after TbTC 95
Fig. 4.13 – Power Consumption before and after TbTC 96
Fig. 4.14 – Hop Count of the token employed in TbTC 98
Fig. 4.15 – Network topology prior to network partitioning 102
Fig. 4.16 – Network topology after partial network partitioning 102

LIST OF TABLES

Table -2-1 – Characteristics of (Kleinrock and Silvester, 1978) 17
Table -2-2 – Characteristics of (Hajek, 1983)....18
Table -2-3 – Characteristics of (Takagi and Kleinrock, 1984) 20
Table -2-4 – Characteristics of (Hou and Li, 1986)....21
Table -2-5 – Characteristics of (Philips, et al, 1989) 22
Table -2-6 – Characteristics of (Xue and Kumar, 2004)....23
Table -2-7 – Characteristics of (Ferrari and Tonguz, 2004) 24
Table -2-8 – Characteristics of (Wan and Yi, 2004)....26
Table -2-9 – Characteristics of centralised NTC (Hu, 1993)....32
Table -2-10 – Characteristics of distributed NTC (Hu, 1993) 33
Table -2-11 – Characteristics of CONNECT (Ramanathan and Rosales-Hain, 2000)34
Table -2-12 – Characteristics of BICONNECT (Ramanathan and Rosales-Hain, 2000) 35
Table -2-13 – Characteristics of CBTC (Li, et al, 2001) 36
Table -2-14 – Characteristics of CBTC-3D (Bahramgiri, et al, 2002) 38
Table -2-15 – Characteristics of CPC (Park and Sivakumar, 2002) 39
Table -2-16 – Characteristics of IPC (Park and Sivakumar, 2002)....40
Table -2-17 – Characteristics of (Gerharz, et al, 2003)....42
Table -2-18 – Characteristics of XTC (Watenhofer and Zollinger, 2004)....43
Table -2-19 – Characteristics of LMST (Li, Hou and Sha, 2005) 44
Table -2-20 – Characteristics of LILNF (Wu and Liao, 2006)....46
Table -2-21 – Characteristics of LILST (Wu and Liao, 2006) 47
Table -2-22 – Characteristics of $FGSS_k$ (Li and Hou, 2006)....48
Table -2-23 – Characteristics of $FLSS_k$ (Li and Hou, 2006)....49
Table -3-1 – Average node degree of each network topology....59
Table -3-2 – Application Layer packets per network topology per simulation run....60
Table -3-3 – Results for the PDR per network topology per simulation scenario 61
Table -3-4 – Results for the Routing Overhead per network topology per simulation scenario 63
Table -3-5 – Results for the End-to-End Delay per network topology per simulation scenario 66
Table -3-6 – Average Path Length per network topology per simulation scenario 66
Table -3-7 – Performance Summary....69
Table -4-1 – Characteristics of Token-based Topology Control 72
Table -4-2 – Application Layer packets per simulation run 88
Table -4-3 – PDRs without and with TbTC....89
Table -4-4 – Routing overhead before and after TbTC....91
Table -4-5 – End-to-End Delay before and after TbTC....93
Table 4-6 – Power Consumption before and after TbTC....96
Table 4-7 – Hop Count of a 30 node network....97

ABSTRACT

Wireless multi-hop networks are not restricted to rural development efforts. They have found uses in the military, industry, as well as in urban areas. The focus of this study is on stationary wireless multi-hop networks whose primary purpose is the provisioning of Internet access using low cost, resource-constrained network nodes.

Topology control algorithms have not yet catered for low cost, resource-constrained network nodes resulting in a need for algorithms that do cater for these types of wireless multi-hop network nodes. An algorithm entitled "Token-based Topology Control (TbTC)" was proposed. TbTC comprises three components, namely: *transmit power and selection*, *network connectivity* and *next node selection*. TbTC differs significantly in its treatment of the synchronisation required for a topology control algorithm to work effectively by employing a token to control the execution of the algorithm. The use of the token also ensures that all the network nodes eventually execute the topology control algorithm through a process called *neighbour control* embedded within the *next node selection* component.

The proposed topology control algorithm, TbTC was simulated using ns-2 and the performances of a 30-node network before and after the algorithm was utilised, were compared. The Packet Delivery Ratio, Delay, Routing Protocol Overhead and Power Consumption were used as the simulation parameters. The *neighbour control* process was found to significantly reduce the number of hops taken by the token to visit each network node at least once. It was found that this process shortened the token traversal by 37.5%.

Based on the results of its simulation, TbTC proves the positive benefits that can be accrued to the use of tokens in topology control as well as highlighting the negative benefits of the creation of uni-directional links in wireless multi-hop networks that utilise the IEEE 802.11 standard.

CHAPTER ONE

INTRODUCTION AND BACKGROUND

1.1 Preamble

Efforts to bridge the digital divide in rural communities has led to the design and deployment of wireless multi-hop networks (see Fig. 1.1). A common scenario sees the provision of wireless links between nearby houses. These links are created by the mounting of antennas (connected to routers) onto houses. The routers are then considered to be nodes on the resulting wireless multi-hop network. These wireless multi-hop networks may span an entire community where the primary purpose is usually the sharing of an outside connection to the Internet (via a pre-designated Internet gateway) and may or may not contain multiple distinct routes between any source-destination pair.

Wireless multi-hop networks are attractive alternatives for connecting rural communities (Allen, et al, 2005), due to their potentially low cost, simplicity, potential for robustness, low power requirements, and the ability to dynamically add subsequent nodes (ensuring scalability), amongst others (Motorola Inc, 2005).

Critical elements for a wireless multi-hop network include the continuous participation of network nodes, the use of a routing protocol, and the network topology or layout. The inter-dependence of these elements is shown in Fig. 1.2.

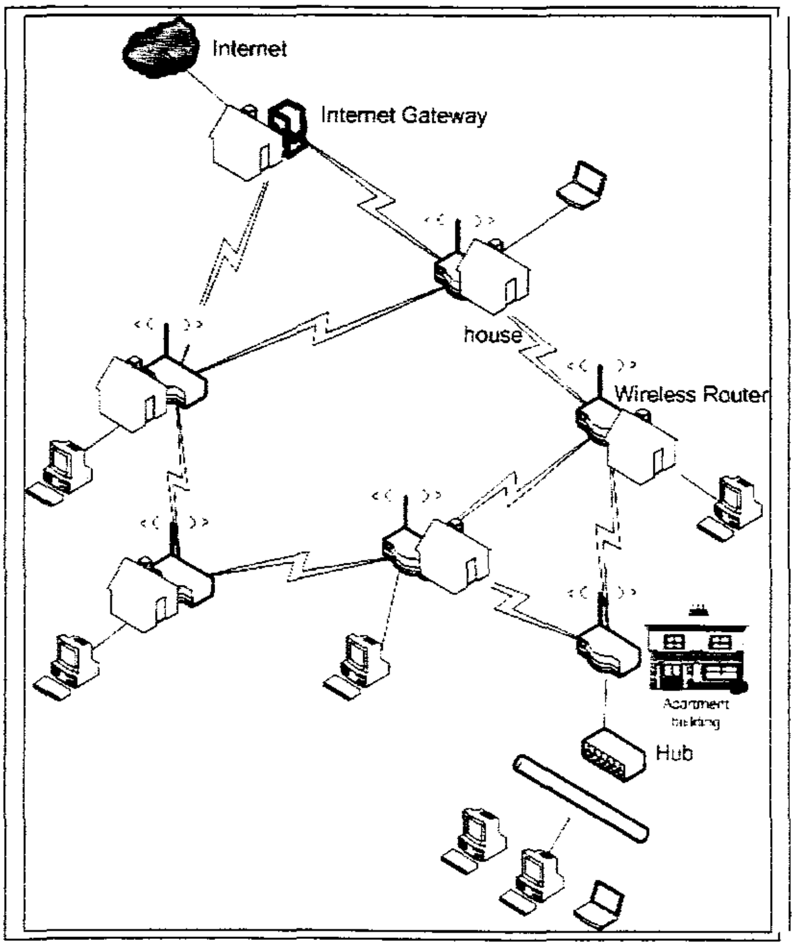


Fig. 1.1 – Wireless multi-hop network example

Unfortunately, there are circumstances under which a network node may cease to participate in the wireless mesh network. Possible causes include environmental factors such as atmospheric interference, electromagnetic interference, attenuation, multipath interference, refraction and reflection (Stallings, 2005).

Defective equipment and human intervention (Lee, et al, 2004) are possible alternate causes for wireless multi-hop network nodes to cease their participation in the network. In other words, network node disconnectivity is experienced.

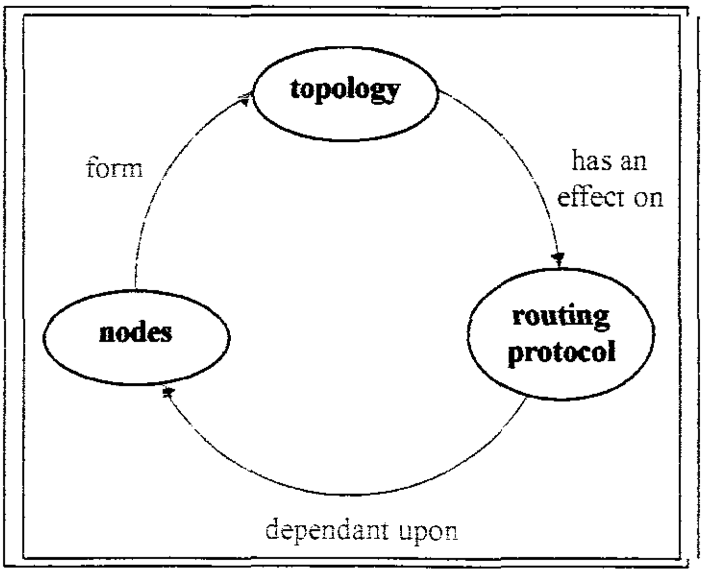


Fig. 1.2 – Interdependence of Network Elements

Topologies in wireless multi-hop networks are formed by the connections that are made between the nodes that comprise the network, and are dependent upon individual nodes' transmission characteristics (Naghian, 2004). An efficient wireless multi-hop network topology is not too dense or too sparse. In other words, the average node degree (number of neighbours) for every node in the network should fall within some pre-defined bound. If the topology is too sparse (the average node degree is too low), there is a danger of network partitioning and high end-to-end delay; if the topology is too dense (the average node degree is too high), the limited spatial reuse reduces network capacity (Ramanathan and Rosales-Hain, 2000).

Network node disconnectivity affects the network topology by reducing the probability that a path exists between every possible source destination pair. The network topology affects the aggregate throughput of the network (Jangeun and Sichitiu, 2003), and is usually a reliable indicator of the robustness of a wireless multi-hop network, but robustness can only be assured if network nodes continuously participate in the network.

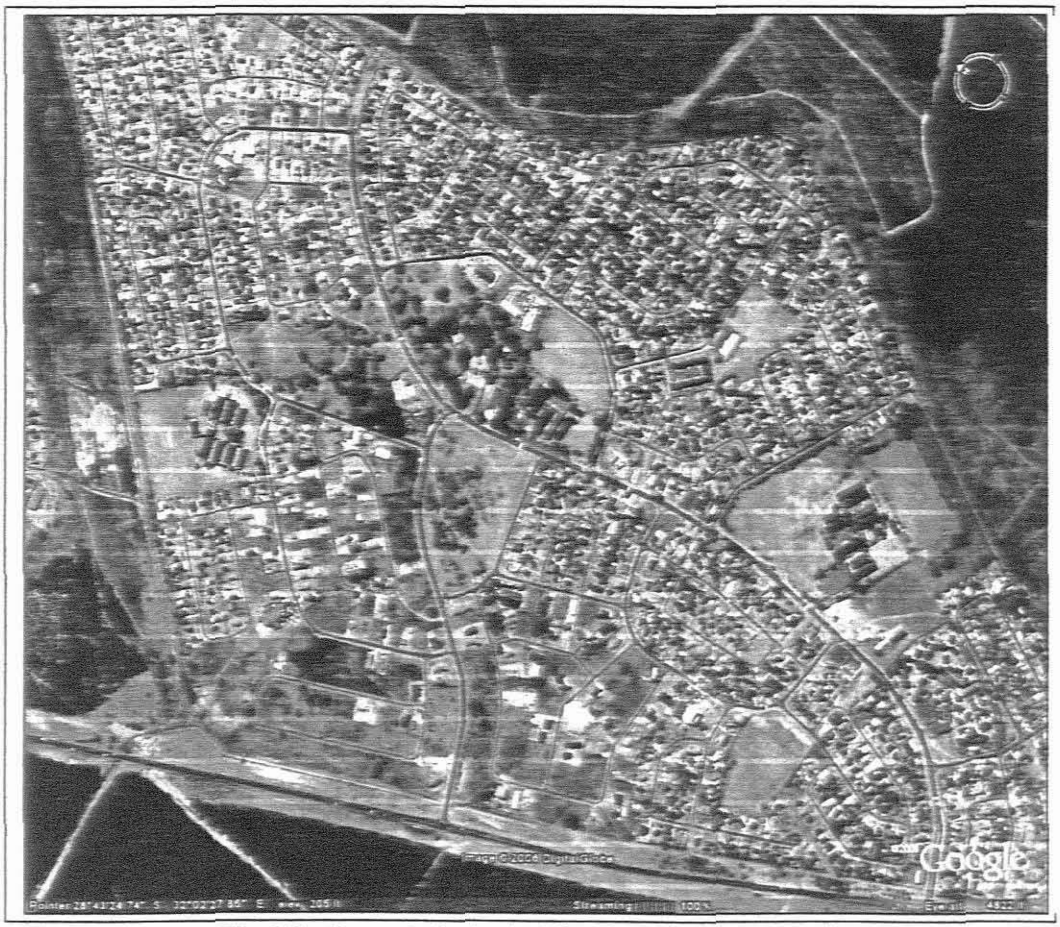


Image © 2006 DigitalGlobe
Pointer 28°43'24.74" S 32°02'27.85" E elev 205 ft
Streaming 100%
Eye alt
Google

Fig. 1.3 – Layout of a typical South African community

Desirable properties for wireless multi-hop network topologies are given in (Hu, 1993), and the need for regular and uniform topological structures is stated. Regular wireless multi-hop network topologies have the added advantage of being able to leverage the design and layout of houses or buildings in a typical community, as depicted in Fig. 1.3.

Cardell-Oliver (2003) conducted a study in which the network topologies shown in Fig. 1.4 were used. These topologies heed the appeal for regular and uniform topologies in (Hu, 1993). Although the work of (Cardell-Oliver, 2003) focuses on wireless sensor networks, there is sufficient commonality between wireless sensor

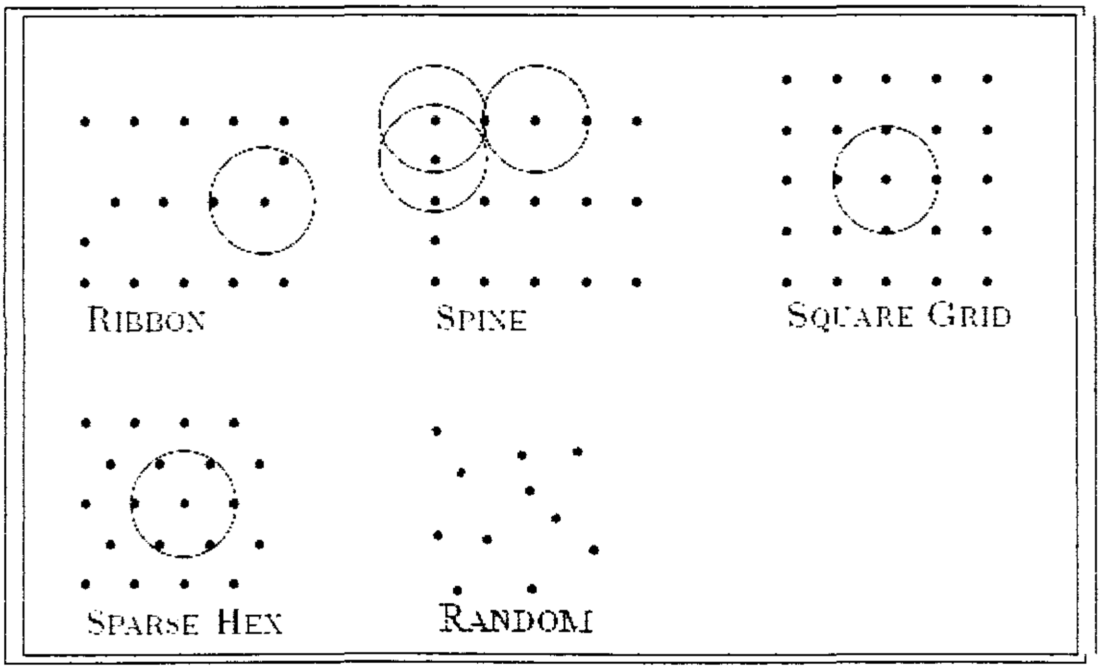


Fig. 1.4 – Network Topologies proposed in (Cardell-Oliver 2003)

networks and wireless multi-hop networks in general for the topologies shown in Fig. 1.4 to be applied in this research.

An important additional criterion for the selection of the network topologies shown in Fig. 1.4 is the variation in the average node degree (number of neighbours) for each network topology. As stated earlier the average node degree has an important influence on the performance of the network and the optimal average node degree is usually bounded. For the purposes of this study, the ability of the network topologies to deliver data packets to the intended destinations in "perfect" conditions (where nodes possess 100% reliability) and when confronted with network node disconnections will be investigated. This investigation will aid in providing the necessary bounds for the proposed topology control algorithm.

For the purposes of this study we focus on stationary, community-based wireless multi-hop networks that are moderately sized and aim to obtain the optimal average node degree based on the results obtained from the network topologies shown in Fig. 1.4. The results obtained will be used to develop a topology control algorithm that can be implemented on resource-constrained wireless multi-hop network nodes (such as the Linksys WRT54G) in order to achieve maximum network performance.

1.2 Statement of the Problem

A network topology is a critical element for the successful operation of the network. For a wireless multi-hop network, the network topology assumes added importance because it affects the average throughput, the amount of interference experienced (Akella, et al, 2005), (Jain, et al, 2005), the efficiency of the routing protocol and the robustness of the network.

The topology control algorithms proposed thus far are unsuitable for low-cost community-based wireless multi-hop networks due to their computational complexity (which increases the cost of the nodes), high communication overhead and information requirements that require capabilities currently not available for low cost, resource-constrained wireless multi-hop network nodes.

These topology control algorithms can also lead to network instability (possibly resulting in a partitioned network) whilst converging in deployed scenarios (Srivastava, et al, 2004). An additional weakness of the majority of proposed topology control algorithms lies in the lack of data originating from either recognised network simulators or real-life wireless multi-hop network test-beds.

In this work an investigation has been conducted into the optimal average number of neighbours and the results of this study have been used to develop a topology control algorithm specifically for low cost, resource-constrained nodes. A practical topology control algorithm eliminates complex computations, reduces the communication overhead to its absolute minimum, does not require position information that can only be obtained via the Global Positioning System, and never creates a partitioned network however short it may be. As a result a practical topology control algorithm should reduce interference and contention for the transmission medium whilst lowering the total energy consumption.

1.3 Rationale for the Research

The continual bridging of the digital divide throughout the world is being accomplished through the use of various technologies, one of them being wireless multi-hop networks. These networks are eminently suitable because of their specific characteristics (highlighted in Section 1.1). Despite the social importance of these networks, the emphasis is usually on providing cost-effective solutions. These networks are created through the use of low cost, resource-constrained nodes and the efficient operation of these networks can be aided by topology control algorithms to ensure that the optimal numbers of neighbours are within transmission range.

Topology control algorithms are designed to reduce interference and increase the network capacity by maximising the spatial reuse of the transmission medium. Current topology control schemes are not suitable for use on low-cost, resource-constrained nodes due to the characteristics highlighted in the previous section. This study aims to develop a practical topology control scheme for low cost, resource-constrained wireless multi-hop network nodes.

1.4 Research Questions

Pertinent to the study two research questions are posed:

1. Which network topologies can be utilised in static community-based wireless multi-hop networks and are they robust enough to handle node disconnectivity?
2. Can a topology control algorithm be tailor-made for use on low cost, resource-constrained wireless multi-hop network nodes?

1.5 Research Goals and Objectives

1.5.1 Goal

The purpose of this project is to develop a wireless multi-hop network topology control algorithm for low cost, resource-constrained network nodes.

1.5.2 The Objectives

The following research objectives were derived from the main research goal:

i. To identify and investigate possible wireless multi-hop network topologies with varying average node degrees under "perfect" conditions;

ii. To investigate these wireless multi-hop network topologies when confronted with random disconnections amongst critical nodes;

iii. To develop a wireless multi-hop network topology control algorithm for low cost, resource-constrained network nodes using the identified design criteria. This topology control algorithm will utilise the bounds obtained (on completion of Objectives 1 and 2) for the optimal average node degree, and

iv. To compare the performances of a network before and after the topology control algorithm has been applied to it.

1.6 Methodology

The results of this project should be applicable to wireless multi-hop networks that employ low cost, resource-constrained network nodes. These networks include research test-beds as well as those wireless multi-hop networks that are deployed in communities across South Africa. The alternative research methods that were available are listed below and the most suitable method was chosen.

1.6.1 Research Method One

This method constitutes the construction of a stationary wireless multi-hop network test-bed with which we can study the performance of the network topologies identified. A real-world implementation of the topology control algorithm would be required for installation on the network nodes.

The disadvantages of this approach are three-fold. Firstly, this method is time consuming because the test-bed has to be correctly set up. Secondly, the time required to develop a real-world implementation of the topology control algorithm may exceed the time limits for this study. Thirdly, use of the test-bed often does not provide enough detail of the real-world phenomena and usage characteristics that are encountered in community-based wireless multi-hop networks.

1.6.2 Research Method Two

An alternative approach would be to identify an existing community-based wireless multi-hop network. This type of network would provide more detailed information on real-world phenomena as well as the usage characteristics of the participants in the network.

This type of network may also introduce another layer of complexity if the deployed network does not employ the Ad hoc On-Demand Distance Vector (AODV) routing protocol. In this case the firmware of all the nodes in the network would need to be modified to run the AODV routing protocol.

Other critical disadvantages associated with this research method are the inability to easily alter the topology of the wireless multi-hop network as well as having to update all the nodes in the network to accommodate the topology control algorithm. The risk of exceeding the time limit imposed on this study is a mitigating factor.

1.6.3 Research Method Three

The final method constitutes the determination of the "typical" wireless multi-hop network node. Information such as the transmission power, reception threshold, channel bandwidth, channel delay, channel error probability, and the area covered by the network could be obtained from those responsible for the deployment of these wireless multi-hop networks in communities. This data could then be used as input into a simulation model.

This method has advantages of being easily modified to employ the required routing protocol as well as the topology control algorithm. Physical interaction with actual network nodes is also avoided.

Unfortunately, the results are dependent on the accuracy of the model parameters used and therefore may not accurately portray or consider some real-world phenomena.

1.6.4 The Chosen Method

Both Method One and Method Two provided more detail in terms of the real-world phenomena and usage characteristics encountered in the deployment and operation of community-based wireless multi-hop networks.

However, neither of the two methods fell within the time constraints imposed upon the project, therefore Method Three was chosen. This method employed the ns-2 (http://www.isi.edu/nanam/ns) network simulation tool. Ns-2 is an open-source, standards-based discrete event simulator targeted at networking research. This tool provided extensive support for IEEE[1] 802.11[2] wireless networks

[1] Institute of Electrical and Electronic Engineers

[2] IEEE Working Group responsible for Wireless LAN standards

1.7 Organization of the Dissertation

The remainder of this dissertation is organised as follows: Chapter Two consists of the review of work relating to the optimal number of neighbours and topology control. The chapter begins by presenting the theoretical framework that is used to analyse the related work and ends with the design criteria necessary for a topology control algorithm. In Chapter Three the optimal number of neighbours in a 30-node wireless multi-hop network is determined. This result is added to the list of design criteria determined in Chapter Two. Chapter Four details the Token-based Topology Control (TbTC) algorithm which is designed using the design criteria from Chapters Two and Three. This chapter ends with the results obtained from TbTC's simulation as well as some of its limitations. Finally, Chapter Five presents the conclusion and possible future work.

CHAPTER TWO

LITERATURE REVIEW

2.1 Overview

Topology control for wireless multi-hop networks is an attempt to automate the process of obtaining and maintaining the optimal node degree (number of neighbours). The need to control the number of neighbours that a node possesses arose from the need to minimise interference, maximise the network capacity and throughput and to improve the power consumption of the nodes in the network.

Several studies have been conducted into obtaining the optimal node degree (also known as "magic numbers") in a wireless multi-hop network (Kleinrock and Silvester, 1978), (Takagi and Kleinrock, 1984), (Wan and Yi, 2004), (Xue and Kumar, 2004) (Philips, et al, 1989) (Xue, Kumar, 2006) (Hou and Li, 1986), yet none of these "magic numbers" have been compared in simulated networks (Mudali et al, 2007). Some of these studies propose magic numbers that are lower bounds whilst others propose upper bounds. Some studies propose magic numbers that are independent of the total number of nodes in the network whilst others establish a relationship between the "magic numbers" and the total number of nodes in the network.

Topology control aims to automate the construction of a network topology that is based upon each node in the network possessing the optimal number of neighbours. Several researchers have developed topology control schemes and many of these topology control schemes are based on the application of Graph Theory[1] to create and maintain an optimal network topology.

This chapter presents a critical analysis of both "magic number" research and existing topology control schemes. Both analyses are based on specifically designed frameworks that will be introduced.

2.2 The Optimal Node Degree (Number of Neighbours)

2.2.1 Overview

Research into the optimal number of neighbours was initiated to aid in the planning of optimal wireless multi-hop network topologies. An optimal network topology is dependent on every node in the network having an optimal number of neighbours. Nodes having too few neighbours results in reduced route redundancy, which reduces the robustness of the network whilst too many neighbours results in increased interference and contention for the transmission medium. The next section introduces the framework that is used to analyse the accumulated body of literature that has dealt with this aspect of wireless multi-hop networks.

2.2.2 Framework for Analysing Work Related to the Optimal Node Degree (Number of Neighbours)

[1] The term "node degree" is derived from the field of Graph Theory

The review of related work was conducted with the aid of the following characteristics identified within the literature. The aim of this framework is to help categorise and critically analyse the published body of research results in this field.

i. **Dependency on Network Size**

The optimal number of neighbours is either a fixed constant that is independent of the total number of nodes in the network, or it varies dependent on the total number of nodes in the network.

ii. **Is the Optimal Number of Neighbours Lower-bound and/or Upper-bound?**

The optimal number of neighbours' value can be either a prescribed minimum or a prescribed maximum. The lower-bound is utilised for specifying connectivity whilst the upper-bound is utilised for minimising interference and maximising network throughput.

iii. **Primary Objective for finding the Optimal Number of Neighbours**

Maximising throughput, reducing interference, guaranteeing connectivity, minimising path length, etc.

iv. **Specification of the Medium Access Control (MAC) scheme used**

A primary purpose of determining the optimal number of neighbours is to maximise a network nodes' access to the transmission medium, therefore, the specification of the optimum number of neighbours for a network should take into account the type of MAC scheme that is used to control access to the transmission medium.

v. **Proof of Concept by Simulation and/or Mathematical Modelling and/or Implementation**

Some values for the optimal number of neighbours for a wireless multi-hop network are determined solely by either simulation or mathematical modelling whilst other works perform the modelling and subsequently utilise the simulation to verify the model.

2.3 Review of Work Related to Determining the Optimal Number of Neighbours in a Wireless Multi-Hop Network

This review is presented in chronological order in order to portray the evolution that has taken place in this field of research.

i. **Optimum Transmission Radii for Packet Radio Networks or Why Six is a Magic Number (Kleinrock and Silvester, 1978)**

Table 2-1 summarises the salient characteristics of the optimal number of neighbours obtained by (Kleinrock and Silvester, 1978). This is a seminal contribution in the field when it comes to determining the optimal number of neighbours in Packet Radio Networks (now commonly known as Wireless Ad-hoc or Wireless Multi-Hop Networks). As the title suggests, the authors prescribe that the optimal number of neighbours is six in terms of maximising the network capacity and throughput. It should be noted that the result obtained for the optimal number of neighbours is independent of the total number of nodes in the network.

Table -2-1 – Characteristics of (Kleinrock and Silvester, 1978)

Characteristics	**Value**
Dependency on Network Size (total number of nodes)	independent
Lower-bound and/or Upper-bound	Lower-bound
Primary Objective	Maximising the one-hop progress of a packet in the desired direction
Specification of the Medium Access Control (MAC) scheme	Yes – ALOHA
Proof of Concept	Mathematical modelling

Kleinrock and Silvester's (Kleinrock and Silvester, 1978) work is not directly applicable to present day community-based wireless multi-hop networks due to the advances in wireless local area network communications technology. Their work is based on the use of the Slotted ALOHA medium access control scheme whilst present day wireless multi-hop networks that are based on the IEEE 802.11 standard typically use the Distributed Coordination Function (DCF) in order to gain access to the transmission medium.

Kleinrock and Silvester (Kleinrock and Silvester, 1978) considered the heavy traffic case in which every node always has data to transmit and will transmit at every opportunity. This is not a realistic assumption for a community-based wireless multi-hop network, because for their assumption to be valid, it would require that the network usage remains stable whereas it has been shown that network traffic is highly dependent on the time of day and thus impacts on the total network usage.

Table -2-2 – Characteristics of (Hajek, 1983)

Characteristics	**Value**
Dependency on Network Size (total number of nodes)	independent
Lower-bound and/or Upper-bound	indeterminate
Primary Objective	Maximising transmission efficiency i.e. The expected progress versus the area covered by the transmission
Specification of the Medium Access Control (MAC) scheme	No
Proof of Concept	Mathematical modelling

ii. Adaptive Transmission Strategies and Routing in Mobile Radio Networks (Hajek, 1983)

This work builds upon the work done by (Kleinrock and Silvester, 1978), in which they suggested that the optimal transmission range was one that allowed for communication with approximately six neighbours. Thereafter the transmission range remained fixed or constant.

Hajek (Hajek, 1983) suggests that the transmission range be dynamic and adjusted at the beginning of every transmission. The routing strategy used was to adjust the node's transmission range in order to reach a neighbour node in the direction of the intended destination node. This strategy does have one obvious limitation when applied to low cost, resource constrained nodes that we are considering. This routing strategy requires that the sending node know the direction of the intended destination node, which is currently not possible with the network nodes that are typically used in community-based wireless multi-hop networks.

However, the mathematical modelling done in this work showed that the adaptive transmission range strategy resulted in each node having an optimal number of neighbours of approximately 3 despite disregarding the MAC scheme. Some additional characteristics of this work are listed in Table 2-2.

iii. Optimum Transmission Ranges for Randomly Distributed Packet Radio Terminals (Takagi and Kleinrock, 1984)

This work differs from most other work in the field by dealing with the optimal transmission power required to maximise the expected one-hop progress that is made in delivering data to its destination. Most other work focuses on the optimal transmission range and the subsequent number of neighbours needed to maximise throughput. The optimal transmission radii for nodes that are randomly distributed are calculated for both the ALOHA and Carrier Sense Medium Access (CSMA) protocols. It should be noted that the ALOHA protocol is a deterministic, time-division based protocol whereas CSMA is based on the ability to detect when the transmission medium is not being utilised.

The researchers found that the MAC scheme employed affected the value obtained for the optimal number of neighbours. The optimal number of neighbours for networks that used the ALOHA medium access control scheme was determined to be approximately eight. It should be noted that this result differs from the results

Table -2-3 – Characteristics of (Takagi and Kleinrock, 1984)

Characteristics	**Value**
Dependency on Network Size (total number of nodes)	independent
Lower-bound and/or Upper-bound	indeterminate
Primary Objective	optimal transmission power required to maximise the expected one-hop progress
Specification of the Medium Access Control (MAC) scheme	Yes – both ALOHA and CSMA
Proof of Concept	Mathematical modelling

obtained from (Kleinrock and Silvester, 1978). This difference is attributed to an inconsistency where the number of neighbouring nodes around the receiver was confused with the number of neighbouring nodes around the transmitter. The optimal number of neighbours for networks that used the CSMA MAC was estimated to be approximately 5. The results obtained for the optimal number of neighbours for both the ALOHA and CSMA MAC protocols is independent of the total number of nodes in the network. A summary of this work can be found in Table 2-3.

iv. Transmission Range Control in Multihop Packet Radio Networks (Hou and Li, 1986)

In this work, the question of the optimal number of neighbours was investigated from a hitherto unique position. The authors considered the impact of three different routing strategies on the transmission range adjustment and ultimately on the optimal number of neighbours.

Table -2-4 – Characteristics of (Hou and Li, 1986)

Characteristics	**Value**
Dependency on Network Size (total number of nodes)	dependent
Lower-bound and/or Upper-bound	Both lower-bound and upper-bound
Primary Objective	Impact of different routing strategies on the transmission range adjustment
Specification of the Medium Access Control (MAC) scheme	no
Proof of Concept	Mathematical modelling and simulation

The three transmission strategies considered were:

- Most Forward with Fixed Radius (MFR)

  This routing strategy forwards packets to neighbours with the greatest forward progress, regardless of the position of the receiving neighbour;

- Nearest with Forward Progress (NFP)

  Packets are forwarded to the nearest neighbour that will result in forward progress. The transmission power will be adjusted so that it is just strong enough to reach the receiving neighbour, and

- Most Forward with Variable Progress (MVR)

  Same as MFR with the exception that the transmission power is adjusted so that the distance between the sender and receiver is successfully traversed.

Table -2-5 – Characteristics of (Philips, et al, 1989)

Characteristics	**Value**
Dependency on Network Size (total number of nodes)	independent
Lower-bound and/or Upper-bound	Both lower-bound and upper-bound
Primary Objective	Determining the effect of the area covered by the network on the optimal number of neighbours
Specification of the Medium Access Control (MAC) scheme	Yes, slotted ALOHA
Proof of Concept	Mathematical modelling

The mathematical modelling and the subsequent simulation of 100 randomly placed nodes showed that the optimal number of neighbours required for the maximum progress and throughput is approximately 8 for NFP and 6 for both MFR and MVR. A summary of this work can be found in Table 2-4.

v. Connectivity Properties of a Packet Radio Network Model (Philips, et al, 1989)

This work establishes a relationship between the optimal number of neighbours and the area of the plane in which the nodes are distributed. Both lower and upper bounds for the optimal number of neighbours are presented: 2.195 < optimal number of neighbours < 10.526, dependent on the use of the slotted ALOHA MAC scheme.

The bounds presented in this work are also dependent on the process followed when distributing nodes across the plane being considered and this could affect the bounds

Table -2-6 – Characteristics of (Xue and Kumar, 2004)

Characteristics	**Value**
Dependency on Network Size (total number of nodes)	dependent
Lower-bound and/or Upper-bound	Both lower-bound and upper-bound
Primary Objective	Establishing a relationship between the total number of nodes in the network and the optimal number of neighbours.
Specification of the Medium Access Control (MAC) scheme	no
Proof of Concept	Mathematical modelling and simulation

obtained. The plane considered in this work is assumed to be a square. This may not be a realistic assumption in real-world deployments of wireless multi-hop networks. A summary of the characteristics of this work is presented in Table 2-5.

vi. The Number of Neighbors Needed for Connectivity of Wireless Networks (Xue and Kumar, 2004)

This work developed heuristics that can be applied to any wireless multi-hop network, independent of the total number of the nodes in the network. The heuristics provided help to arrive at the lower and upper bounds for the connectivity of the network. The optimal number of neighbours lies somewhere within the bounds proposed.

One significant proposal emanating from this research is that when each node is connected to less than 0.0074log(n) nearest neighbours, then the network tends to be

Table -2-7 – Characteristics of (Ferrari and Tonguz, 2004)

Characteristics	**Value**
Dependency on Network Size (total number of nodes)	independent
Lower-bound and/or Upper-bound	lower-bound
Primary Objective	To guarantee network connectivity
Specification of the Medium Access Control (MAC) scheme	no
Proof of Concept	Mathematical modelling

partitioned (disconnected) with probability one as *n* increases (where *n* is the total number of nodes in the network). This amounts to the lower bound on connectivity. The proposed upper bound states that if each node is connected to more than 5.1774log(*n*) nearest neighbours then the network is connected with probability one as *n* increases. As stated earlier, 0.074log(*n*) < optimal number of neighbours < 5.1774log(*n*).

A shortcoming of this work is the disregard for the MAC protocol to be utilised by the nodes in the network. As proven by (Takagi and Kleinrock 1984) the type of MAC protocol used does impact on the number of neighbours that a node should possess within its transmission range. A summary of the characteristics of this work can be found in Table 2-6.

vii. Minimum Number of Neighbors for Fully Connected Uniform Ad Hoc Wireless Networks (Ferrari and Tonguz, 2004)

This paper determines the minimum number of neighbours for uniform (each node has on average the same number of neighbours) wireless multi-hop networks. Existing work is extended by establishing relationships between the minimum number of neighbours and the transmission power and node spatial density in a two-dimensional plane. Critical values for the required transmission power and node spatial density are established and the failure to meet the critical values means that the full connectivity of the network cannot be guaranteed.

The authors determined that the "magic number" guaranteeing a fully connected uniform network is on average equal to π (3.14), provided that the critical values for the transmission power and node density are met. A summary of the characteristics of this work can be found in Table 2-7.

viii. Asymptotic Critical Transmission Radius and Critical Neighbor Number for *k*-Connectivity in Wireless Ad Hoc Networks (Wan and Yi, 2004)

This work improved upon the upper bound for the optimal number of neighbours that was established by (Xue and Kumar, 2004). The authors suggested that if each node in the network was able to connect to a maximum of 2.718log(n) nearest neighbours then the resultant network topology would be almost surely connected.

A major disadvantage of this work is the implicit requirement that each node in the network knows the total number of nodes in the network at any point in time. This requirement is a drawback especially when the case of community-driven wireless multi-hop networks is considered. These types of networks are typically expected to

Table -2-8 – Characteristics of (Wan and Yi, 2004)

Characteristics	**Value**
Dependency on Network Size (total number of nodes)	dependent
Lower-bound and/or Upper-bound	upper-bound
Primary Objective	To obtain the maximum number of neighbours required to ensure an almost surely connected network topology
Specification of the Medium Access Control (MAC) scheme	no
Proof of Concept	Mathematical modelling

grow unaided over time and therefore the knowledge of the total number of networks is not a practical consideration. Additional characteristics of this work are shown in Table 2-8.

2.4 Summary of Work Related to the Optimal Number of Neighbours

The initial aim of research in this field of research was obtaining the "magic number" that would help to achieve the maximum throughput and the minimum delay whilst minimising the interference caused. These initial "magic numbers" took an individualistic approach by assuming that once all the nodes in the network had the prescribed number of neighbours, that the resulting network topology would be connected. It was subsequently shown that this assumption was invalid (Wan and Yi, 2004), (Xue and Kumar, 2004), (Xue and Kumar, 2006).

This situation led to the "magic number" being dependent on the total number of nodes in an attempt to ensure that the resulting network topologies were connected. Once the connectivity could be assured, both the upper and lower bounds on the proposed "magic numbers" were tightened in order to achieve greater accuracy.

Despite the progress made in this field of research, the number of neighbours could only be controlled prior to the network's deployment, during the initial planning and site determination phase. The lack of an automated method of always ensuring that a wireless multi-hop network node had the optimal number of neighbours led to the design of topology control algorithms.

2.5 Topology Control for Wireless Multi-hop Networks

2.5.1 Overview

Topology control can be viewed as a product of the prior research work conducted to determine the optimal number of neighbours in a wireless multi-hop network. This recent field of study ultimately aims to automate the process of creating a network topology wherein each node in the network is connected to the optimal number of neighbours, resulting in reduced interference and power consumption, whilst guaranteeing that the network remains connected. Topology control is often a compromise between the node's transmission range (which is proportional to the transmission power), the number of neighbours and the average number of distinct paths between every source and destination pair. As a result several complex topology control algorithms have been proposed.

The next section details the framework used to analyse the literature that has accumulated in the creation of topology control algorithms.

2.5.2 Framework for Analysing Work Related to Topology Control

The review of related work was conducted with the aid of the following characteristics identified within the literature. The aim of this framework is to help categorise and critically analyse the published work in this field.

i. Node characteristics

Node characteristics refer to the hardware and firmware platforms utilised by the nodes. Networks comprised of these nodes can be classified as either *homogeneous* or *heterogeneous*. Homogeneous networks contain nodes with the same characteristics, such as antenna type, transmission range, routing protocol, and so on. On the other hand, heterogeneous networks contain nodes with varying characteristics most often the transmission range.

ii. Information Requirements

All topology control algorithms require information in order to perform their functions. There is an inherent trade-off between the quality of the information required and the cost of performing the algorithm. Higher quality information results in higher control message overheads, which adversely affect the performance of the network. The information that is required by topology control algorithms usually come in one of three types, namely:

a. Location information

b. Direction information

c. Neighbour information

Location-based topology control algorithms rely on the network node's ability to determine its location usually through the use of a GPS-based system. Direction-based algorithms assume that network nodes do not know their positions but possess the ability to estimate the relative direction of each neighbour. The most common information requirement for topology control algorithms is neighbour-based information. Neighbours are distinguished by some form of identification, usually an Internet Protocol address and are usually ordered by link quality. The works reviewed are based on a requirement for neighbour-based information.

iii. Architecture

The architectural structure refers to the functioning of the topology control scheme. Some schemes use a single (usually central) node to determine the best connectivity for each node in the network. Other schemes are distributed in nature whereby each node determines its own best connectivity based on the information gathered. Connectivity relates to the number of neighbours that a node is able to connect to, or within transmission range.

iv. Link Characteristics

This refers to the types of links that are eventually created by a topology control algorithm. Links can be either uni-directional or bi-directional in nature.

v. **Connectivity Flexibility**

This refers to a topology control algorithm's ability to vary the connectivity of the network that is created once the algorithm is applied. Some algorithms possess the ability to create a network topology with varying degrees of connectedness (*k*-connectedness) whilst other algorithms do not offer such flexibility.

vi. **Suitability for low-cost community-based wireless multi-hop networks**

This work focuses on community-based wireless mesh networks. As such, the issue of cost must be considered when evaluating topology control algorithms. The costs considered stem from the computational cost of the topology control algorithm, which in turn determines the cost of the hardware.

2.6 Review of Topology Control Algorithms

The review is presented in chronological order to portray the evolution of the topology control algorithms proposed to-date.

i. **Topology Control for Multihop Packet Radio Networks (Hu, 1993)**

This work represents one of the earliest topology control algorithms to be formulated. Both a centralised and distributed versions of the Novel Topology Control (NTC) algorithm are described.

NTC adopted a two-step approach in which a network with good connectivity is created via the use of the Delaunay Triangulation (DT) algorithm and the resulting network topology's capacity is optimised. The DT algorithm is implemented at a central node and was used to maximise the minimum angle of all triangles in the original network topology. The result of this process was the reduction in the average number of neighbours for the nodes in the network.

The distributed version of the NTC algorithm differs from the centralised version by requiring every node in the network to implement the DT algorithm independently. This version eliminates the transferring of data to a central site and the subsequent wait until the results of the processing can be returned, but it does have its disadvantages as well.

The disadvantages of the centralised version are discussed first. No mention is made of the central node selection criteria and the subsequent process of alerting the other nodes in the network of the identity of the selected node. The use of the centralised NTC algorithm also introduces unnecessary latency into the topology control process. All the nodes in the network must submit their local neighbourhood information to the central node and must wait until the results of the processing can be returned to them before adjusting their transmission powers.

The processed information that will be returned to the network nodes will not reach all the nodes in the network simultaneously due to the multi-hop nature of the networks being considered. This situation may inadvertently compromise the return of the processed information. The reason for this is that a path that existed before the application of the centralised NTC algorithm may no longer exist after one or more intermediate nodes in the original path act on the information obtained from the

Table -2-9 – Characteristics of centralised NTC (Hu, 1993)

Algorithm Characteristics	**Value**
Node characteristics	heterogeneous
Information Requirements	Neighbour-based
Architecture	centralised
Link characteristics	Symmetric (bidirectional)
Connectivity flexibility	*k*-connectivity, flexible
Suitability for low-cost community-based networks	Not suitable due tc the use of a central node for computation, the need for global knowledge and unnecessarily high latency.

central node. At best, a suboptimal path to the destination will have to be chosen. At worst, the processed information will not reach the intended destination.

Another concern is the scalability of the centralised NTC algorithm. As the number of nodes increases, the average path length from the outer regions of the network to the central node will increase thereby increasing the latency and the possibility that routes from the central node to the outer regions would have changed as intermediate nodes apply the changes to their transmission power. New routes would have to be found, increasing the communication overhead of this algorithm. This issue and the ones identified above impact severely on the practicality of the centralised NTC algorithm. Additional characteristics of the centralised NTC algorithm are given in Table 2-9.

The distributed NTC algorithm has its fair share of problems as well. This version assumes that all the nodes in the network possess the ability to compute the DT

Table -2-10 – Characteristics of distributed NTC (Hu, 1993)

Algorithm Characteristics	**Value**
Node characteristics	heterogeneous
Information Requirements	Neighbour-based
Architecture	distributed
Link characteristics	Symmetric (bidirectional)
Connectivity flexibility	*k*-connectivity, flexible
Suitability for low-cost community-based networks	Not suitable due the assumption that every node possess computational ability to process the DT algorithm. No synchronisation is provided to aid in maintaining network connectivity.

algorithm in order to reduce its number of neighbours. The synchronisation required to aid in the maintenance of network connectivity is disregarded. A situation could occur where two neighbouring nodes are in differing stages in their execution of the distributed NTC algorithm and the result is that the link between them is broken due to the lack of synchronisation in the execution of the algorithm. Additional characteristics of this work are listed in Table 2-10.

ii. Topology Control of Multihop Wireless Networks using Transmit Power Adjustment (Ramanathan and Rosales-Hain, 2000)

This work aimed to create a fully-connected wireless multi-hop network topology whilst minimising the maximum transmission power utilised. Two centralised topology control algorithms were presented to achieve the authors' aim.

Table -2-11 – Characteristics of CONNECT (Ramanathan and Rosales-Hain, 2000)

Algorithm Characteristics	**Value**
Node characteristics	heterogeneous
Information Requirements	location-based
Architecture	centralised
Link characteristics	Symmetric (bidirectional)
Connectivity flexibility	Nonflexible (*k*=1)
Suitability for low-cost community-based networks	Not suitable due to the use of a central node for computation, the need for global knowledge and location information and unnecessarily high latency.

The first algorithm, named CONNECT was designed to create *k*-connectivity, where *k*=1. This results in the formation of a connected network that contains the minimum amount of route redundancy and the creation of many critical nodes, whose failure would result in the partitioning of the network. These critical nodes are also potential bottlenecks for the network's throughput. The characteristics of the CONNECT topology control algorithm is presented in Table 2-11.

The second centralised topology control algorithm named BICONNECT was designed to create a *k*-connected network where *k*=2. The resultant network is an improvement on the CONNECTed version as the route redundancy is improved and the impact of the performance bottlenecks that previously existed has been reduced. The characteristics of the BICONNECT topology control algorithm are presented in Table 2-12.

Table -2-12 – Characteristics of BICONNECT (Ramanathan and Rosales-Hain, 2000)

Algorithm Characteristics	Value
Node characteristics	heterogeneous
Information Requirements	location-based
Architecture	centralised
Link characteristics	Symmetric (bidirectional)
Connectivity flexibility	Nonflexible (*k*=2)
Suitability for low-cost community-based networks	Not suitable due to the use of a central node for computation, the need for global knowledge and location information and unnecessarily high latency.

The authors make no mention of the central node selection criteria and the subsequent process of informing all the nodes in the network of the identity of the central node. Due to the centralised nature of the CONNECT and BICONNECT topology control algorithms, they suffer from the same drawbacks that are discussed in our review of (Hu, 1993). These drawbacks impact severely on the practicality of the CONNECT and BICONNECT algorithms.

iii. Analysis of a Cone-Based Distributed Topology Control Algorithm for Wireless Multi-Hop Networks (CBTC) (Li, et al, 2001), (Li, et al, 2005)

This work introduces the Cone-Based Topology Control algorithm that allows each node to individually adjust its transmission power. Each node divides its transmission range into distinct cones according to some value of α (angle of a cone in degrees).

Table -2-13 – Characteristics of CBTC (Li, et al, 2001)

Algorithm Characteristics	**Value**
Node characteristics	heterogeneous
Information Requirements	direction-based
Architecture	Distributed
Link characteristics	bi-directional (after optimisation)
Connectivity flexibility	Flexible, dependent on α
Suitability for low-cost community-based networks	Not suitable because of the necessary ability to determine the relative directions of its neighbours. The lack of synchronisation between neighbours is also a stumbling block.

CBTC attempts to minimise the maximum transmission power required to ensure that the node has a neighbour in every cone in its transmission range.

Although CBTC is a distributed algorithm, it cannot be utilised on low-cost, resource-constrained wireless multi-hop network nodes because it requires that nodes possess the ability to estimate the direction from which transmissions are being received. This ability aids in identifying the cones in which its neighbours reside.

The focus on minimising each node's transmission range of every node in the network may also inadvertently lead to a reduction in the redundancy of the network, thereby increasing the possibility of bottleneck nodes that negatively affect the network's throughput. The lack of synchronisation between neighbouring nodes is a concern as CBTC requires feedback from its neighbours in the form of acknowledgements to

Hello messages. The failure of a neighbour to respond to a broadcasted Hello message may inadvertently cause an increase in the broadcast node's transmission power as it seeks to find a neighbour in that particular cone. A summary of CBTC's characteristics is listed in Table 2-13.

iv. Fault-Tolerant and 3-Dimensional Distributed Topology Control Algorithms in Wireless Multi-Hop Networks (Bahramgiri, et al, 2002)

The CBTC algorithm was extended via CBTC-3D by improving the redundancy of the network topology as well as considering a 3-dimensional scenario such as the interior of a multi-storeyed building (Bahramgiri, et al, 2002). The review of (Li, et al, 2001), (Li, et al, 2005) indicated that the original CBTC algorithm reduced the redundancy in the network, therefore reducing the fault-tolerant nature of the network. Bahramgiri, et al (2002), aimed to guarantee the fault-tolerant nature of the network by ensuring that the resulting network preserves *k*-connectivity. The ability to detect neighbours in 3-dimensional space was also introduced by using the concept of cones that was introduced in the original CBTC algorithm.

The disadvantages of this work include the necessary requirement that the nodes in the network possess the ability to determine the relative directions of its neighbours in 3-dimensional space. Another disadvantage is the lack of synchronisation between neighbouring nodes that could inadvertently increase the transmission power required to maintain the *k*-connectivity of the network. Additional characteristics of this topology control algorithm are shown in Table 2-14.

Table -2-14 – Characteristics of CBTC-3D (Bahramgiri, et al, 2002)

Algorithm Characteristics	**Value**
Node characteristics	heterogeneous
Information Requirements	direction-based
Architecture	Distributed
Link characteristics	bi-directional (after optimisation)
Connectivity flexibility	Flexible, dependent on α
Suitability for low-cost community-based networks	Not suitable because of the necessary ability to determine the relative directions of its neighbours in 3-dimensional space. The lack of synchronisation between neighbours is also a stumbling block.

v. Load-Sensitive Transmission Power Control in Wireless Ad-hoc Networks (Park and Sivakumar, 2002)

Prior work has dealt with utilising the minimum power required to achieve *k*-connectivity. This work differs in that it aims at adjusting the transmission power of every node in the network to achieve the maximum possible throughput. The two topology control algorithms proposed in this work determine the optimal transmission power for each node in the network based on the traffic load that each node experiences, the total number of nodes in the network and the area covered by the network.

The first algorithm, named Common Power Control (CPC) assigns a common transmission power to all the nodes in the network. CPC avoids the use of a centralised node by having each node determine its own optimal transmission power

Table -2-15 – Characteristics of CPC (Park and Sivakumar, 2002)

Algorithm Characteristics	Value
Node characteristics	homogeneous
Information Requirements	Neighbour-based with traffic load.
Architecture	Distributed
Link characteristics	both bi-directional and directional
Connectivity flexibility	inflexible
Suitability for low-cost community-based networks	Not suitable due to the use of the broadcast mechanism and the excessive overhead created. Also requires knowledge of the area covered by the network.

based on local information and subsequently floods the network with this information via advertisements. All the nodes in the network then adopt the largest advertised transmission power.

CPC does possess some disadvantages that compromise its practicality. First, the use of the broadcast mechanism to determine the largest transmission power does not guarantee that all the nodes in the network will receive the largest transmission power advertisement. Flooding is an inherently unreliable communication mechanism due to the potential for collisions, which result in the loss of transmission power adjustments. This phenomenon is compounded as the size of the network grows. As a result either one of two situations may occur; either a suboptimal transmission power is globally adopted if that particular transmission power advertisement was received by all the nodes in the network, or, differing transmission powers will be utilised in differing

Table -2-16 – Characteristics of IPC (Park and Sivakumar, 2002)

Algorithm Characteristics	**Value**
Node characteristics	heterogeneous
Information Requirements	Neighbour-based with traffic load
Architecture	Distributed
Link characteristics	both bi-directional and directional
Connectivity flexibility	inflexible
Suitability for low-cost community-based networks	Not suitable due to the use of the transmission load as a transmission power adjustment criterion. Could lead to network topology instability as the traffic load fluctuates.

regions of the network. Both situations may result in a less than maximum network throughput despite the excessive overhead created by the broadcasting mechanism.

Second, the use of the traffic load as one of the transmission power adjustment criteria is also a disadvantage. The traffic load is dynamic and thus may change rendering a node's transmission power advertisement obsolete during the time taken to receive the common transmission power. Additional characteristics of the CPC algorithm are listed in Table 2-15.

The second algorithm, named Independent Power Control (IPC) was designed to allow each node in the network to choose its own transmission power, thus avoiding the latency and overhead introduced by CPC. IPC chooses its transmission power in exactly the same manner as CPC; they both utilise the traffic load. IPC suffers from two important disadvantages; first, it disregards the synchronisation required to aid in maintaining the connectivity of the network during the process of applying the algorithm. Second, the use of the traffic load as a transmission power adjustment criterion could lead to network topology instability due to the dynamic nature of the traffic load experienced at a network node. Additional characteristics of IPC are listed in Table 2-16.

vi. A Cooperative Nearest Neighbours Topology Control Algorithm for Wireless Ad Hoc Networks (Gerharz, et al, 2003)

This work represents the first topology control algorithm with the potential to be deployed on low cost, resource-constrained wireless multi-hop network nodes. The proposed algorithm does not require specialised hardware and has minimal computational needs.

This algorithm uses local neighbourhood information to control the number of neighbours of each node in the network. This results in each node independently adjusting its transmission power to satisfy the algorithm. The algorithm does not guarantee that each node will have k neighbours but it rather tries to maintain the number of neighbours within k_{min} <= number of neighbours (k) <= k_{max}. The choice of the value for k_{min} is not discussed which may result in the optimal network topology not being created. The authors suggest that kmax = k_{min} + 6. An additional criticism of this work is the disregard for the synchronisation required to aid in ensuring that the topology control process does not result in a partitioned network in some scenarios. Additional characteristics of this work are listed in Table 2-17.

Table -2-17 – Characteristics of (Gerharz, et al, 2003)

Algorithm Characteristics	Value
Node characteristics	heterogeneous
Information Requirements	Neighbour-based
Architecture	Distributed
Link characteristics	bi-directional
Connectivity flexibility	Flexible, bounded [$k_{min};k_{max}$]
Suitability for low-cost community-based networks	Suitable despite not synchronising the transmission power adjustment. Requires only local neighbourhood information and does not require specialised hardware.

vii. XTC: A Practical Topology Control Algorithm for Ad-Hoc Networks (Watenhofer and Zollinger, 2004)

The distributed XTC algorithm caters for 3-dimensional situations and allows a node to order its neighbours by decreasing link quality. Each node creates its neighbour order and exchanges it with its neighbouring nodes. The nodes in the network determine their local neighbourhood (the collection of neighbouring nodes) after receiving the neighbour orders from all of its neighbours.

In general terms node A builds or maintains a direct communication link to node B if node A has no other neighbour (node C) that can more easily reach node B. Whilst this process does ensure that links with the highest quality are maintained, it does add to the computational complexity of the algorithm, because the neighbours orders of neighbouring nodes must be consulted when deciding on the local neighbourhood.

Table -2-18 – Characteristics of XTC (Watenhofer and Zollinger, 2004)

Algorithm Characteristics	**Value**
Node characteristics	heterogeneous
Information Requirements	Neighbour-based
Architecture	Distributed
Link characteristics	bi-directional
Connectivity flexibility	inflexible
Suitability for low-cost community-based networks	Not suitable due to the overhead incurred during the exchange of neighbour orders and the computational complexity introduced during their processing. The lack of synchronisation is also a disadvantage.

Additional disadvantages of this approach include the overhead created during the process of exchanging a node's neighbour orders amongst its neighbours, the reduction in the redundancy of the network created by the XTC algorithm, as well as the lack of synchronisation amongst neighbouring nodes. The characteristics of this algorithm are listed in Table 2-18.

viii. Design and Analysis of an MST-Based Topology Control Algorithm (Li, Hou and Sha, 2005)

LMST constructed a global network topology by having each network node construct its own local MST independently. The algorithm guards against a situation in which a node has too many neighbours by enforcing an upper bound of 6 on the number of neighbours considered. This allows for the creation of a global network topology

Table -2-19 – Characteristics of LMST (Li, Hou and Sha, 2005)

Algorithm Characteristics	Value
Node characteristics	homogeneous
Information Requirements	Neighbour-based, location based
Architecture	distributed
Link characteristics	Uni-directional and bi-directional. Can be optimised to ensure complete bidirectionality
Connectivity flexibility	Flexible, bounded to a maximum node degree of 6
Suitability for low-cost community-based networks	Not suitable, the algorithm is too severe in its optimisation and results in the creation of a high number of critical links, thereby increasing the probability of network partitioning.

where the node degree is bounded by 6, thereby reducing the MAC-level interference and contention.

LMST in its basic form created a network topology that may consist of both uni-directional and bi-directional links. Uni-directional links do not allow for the proper functioning of the Medium Access Control (MAC) mechanisms of the IEEE 802.11 standard. LMST addresses this problem by providing an optional optimisation that ensures that all the links in the network are bi-directional. A summary of LMST is found in Table 2-19.

Simulation of LMST (Li and Hou, 2005) showed that it reduces the MAC-level contention, but at the expense of the overall redundancy and resulting reliability of the network. LMST is shown to achieve an average node degree of 2 which does not fall within the optimal range of neighbours [4 – 9], depending on the assumptions made and the overall network model utilised.

ix. Interference-Efficient Topology Control in Wireless Ad Hoc Networks (Wu and Liao, 2006)

The goal of this work was to produce an interference-efficient wireless multi-hop network topology via the use of the Low Interference-Load Neighbourhood Forest (LILNF) algorithm. The interference load refers to the number of nodes that contribute interference to a node[2].

LILNF was a centralised algorithm which required the interference load of each node in the network to be sent to a centralised node for processing. This algorithm attempted to minimise the interference load of every node in the network whilst maintaining the network connectivity. Network connectivity is ensured by checking that a path between every pair of nodes in the network exists.

The LILNF algorithm does have its disadvantages, which include the use of a centralised node for the necessary computations, the assumption that such a node exists, the communication overhead incurred during the transmission of every node's local neighbourhood information, as well as the delay in receiving the output from the central node. The characteristics of LILNF can be found in Table 2-20.

[2] The interference load should not be confused with the node degree since interference can be contributed by nodes outside of the local neighbourhood.

Table -2-20 – Characteristics of LILNF (Wu and Liao, 2006)

Algorithm Characteristics	Value
Node characteristics	homogeneous
Information Requirements	Neighbour-based with interference load
Architecture	Centralised
Link characteristics	bidirectional
Connectivity flexibility	inflexible
Suitability for low-cost community-based networks	Not suitable due tc the use of a central node for computation, the need for global knowledge and the communication overhead created.

An extension to the LILNF topology control algorithm, the Low Interference-Load Spanner Topologies algorithm, is an attempt to maintain a low interference network topology whilst ensuring that the path lengths between every pair of nodes in the network are below a certain threshold.

Due to LILST's centralised nature, it suffers from all of the disadvantages of LILNF despite its consideration for the average path length in the network. This consideration merely increases the computational complexity of the algorithm. The characteristics of LILST can be found in Table 2-21.

Table -2-21 – Characteristics of LILST (Wu and Liao, 2006)

Algorithm Characteristics	**Value**
Node characteristics	homogeneous
Information Requirements	Neighbour-based with interference load
Architecture	Centralised
Link characteristics	bidirectional
Connectivity flexibility	inflexible
Suitability for low-cost community-based networks	Not suitable due tc the use of a central node for computation, the need for global knowledge and the communication overhead created. Adds to the computational complexity of LILNF.

x. Localized Fault-Tolerant Topology Control in Wireless Ad Hoc Networks (Li and Hou, 2006)

This work aimed at preserving the redundancy of the wireless multi-hop network by preserving its *k*-connectivity. Both a centralised and distributed algorithm were devised with this goal in mind.

Fault-tolerant Global Spanning Subgraph ($FGSS_k$) is a centralised topology control algorithm that allows for the connectivity levels to be specified (*k*-connectivity). Connectivity is preserved by minimising the maximum transmission power of the nodes in the network. By virtue of this min-max property, $FGSS_k$ can maximise the lifetime of the network when nodes with self-contained power sources are considered.

Table -2-22 – Characteristics of FGSS$_k$ (Li and Hou, 2006)

Algorithm Characteristics	Value
Node characteristics	homogeneous
Information Requirements	Neighbour-based, location based
Architecture	Centralised
Link characteristics	Symmetric (bidirectional)
Connectivity flexibility	*k*-connectivity, flexible
Suitability for low-cost community-based networks	Not suitable due to the use of a central node for computation, the need for global knowledge and the need for location information.

FGSS$_k$ requires a single network node (preferably centrally situated) to possess global knowledge of the network. This requirement is in direct conflict with the philosophy of wireless ad-hoc networking. Additionally, the collection and distribution of the information required by the algorithm is likely to consume a high proportion of the limited bandwidth available if practically implemented.

The practical implementation of FGSS$_k$ in a low-cost wireless multi-hop network is likely to pose a number of challenges. First, a central network node for the purpose of collecting the required information and subsequently disseminating the information needed to form the required network topology (this information will need to be multicast to every other node in the network) is required. Second, the network throughput is likely to deteriorate due to the transmission of the control data necessary for the functioning of FGSS$_k$ algorithm. Third, the selection of the optimal update interval is a trade-off between the responsiveness of the topology control algorithm

Table -2-23 – Characteristics of $FLSS_k$ (Li and Hou, 2006)

Algorithm Characteristics	**Value**
Node characteristics	homogeneous
Information Requirements	Neighbour-based, location based
Architecture	Distributed
Link characteristics	Symmetric (bidirectional)
Connectivity flexibility	*k*-connectivity, flexible
Suitability for low-cost community-based networks	Not suitable due to the lack of synchronisation.

and the overall network throughput and lastly, the need for location information either via GPS or triangulation methods are unrealistic requirements for low cost, resource-constrained network nodes. The characteristics of the $FGSS_k$ algorithm can be found in Table 2-22.

In response to some of the challenges posed by the centralised $FGSS_k$ algorithm, the distributed Fault-tolerant Local Spanning Subgraph ($FLSS_k$) was introduced. $FLSS_k$ alleviates the need for a central authority by distributing the algorithm amongst all the nodes in the network. Each node has the ability to adjust its own transmission power. **Hello** messages are used to collect the data used by the algorithm. $FLSS_k$ possesses the ability to ensure that all the transmission links in the network topology are bi-directional. Table 2-23 presents the other relevant characteristics of $FLSS_k$

Although the effectiveness of $FLSS_k$ is proven via simulation, a practical implementation is presently not feasible due to the computational complexity involved in forming a Minimum Spanning Tree (MST). An additional concern relates to the lack of synchronisation amongst neighbouring nodes. A lack of synchronisation amongst nodes in differing stages of the execution of the algorithm may result in a node's inability to respond to a request for information issued by one of its neighbours. Such a situation results in the use of incomplete local neighbourhood information to compute a node's transmission power, thereby resulting in the creation of a sub-optimal network. The characteristics of $FLSS_k$ are shown in Table 2-23.

2.7 Summary of Work Related to Topology Control

Topology Control algorithms were devised to automate the process of ensuring the creation and subsequent maintenance of optimal wireless multi-hop network topologies. These algorithms often have competing design criteria and as a result they are usually a compromise between a node's transmission range, its node degree (number of neighbours), the network throughput, the interference levels experienced and the average number of distinct paths between every source and destination pair in the network.

The review of proposed topology control algorithms has identified that none are unsuitable for low-cost, resource-constrained wireless multi-hop network nodes due to the algorithms' computational complexity, high communication overhead and information requirements that require capabilities currently not available for low cost, resource-constrained wireless multi-hop network nodes. These topology control algorithms can also lead to network instability due to a lack of synchronisation amongst neighbouring nodes (possibly resulting in a partitioned network) whilst converging in deployed scenarios.

The review process, aided with the two specially formulated frameworks, has provided the design criteria for a practical topology control algorithm that can be used by low cost, resource-constrained wireless multi-hop network nodes. These design criteria are briefly introduced in Section 2.8.

2.8 Summary of the Required Design Criteria for a Topology Control Algorithm for Low Cost, Resource-Constrained Wireless Multi-hop Network Nodes

The literature reviews of the optimal node degree (number of neighbours) and topology control algorithms as well as the use of specially formulated literature analysis frameworks has identified the design criteria necessary for a practical topology control algorithm for low cost, resource-constrained network nodes. A list of the identified design criteria follows. Detailed discussions on the design criteria can be found in Chapter Four.

The identified design criteria, listed in no particular order, are:

i. The use of a distributed algorithm;

ii. Minimal computation;

iii. Minimal communication overhead;

iv. Minimal information requirements (we advocate only the use of the node degree);

v. Minimal latency;

vi. Maintenance of network connectivity;

vii. Maintenance of the optimum levels of redundancy;

viii. Provision of synchronicity between neighbouring nodes, and

ix. Heterogeneous transmission radii.

These criteria listed above are used in the design of our practical topology control algorithm entitled “Token-based Topology Control”.

Despite the nine design criteria listed, the list is incomplete. The missing criterion in this list is the optimal minimum number of neighbours[3] that each network node should maintain (wherever possible) in a practical network topology. This problem is addressed as the subject of Chapter Three.

[3] The optimum minimal number of neighbours ensures that the optimal levels of redundancy are maintained in a practical wireless multi-hop network topology.

CHAPTER THREE

PERFORMANCE ANALYSIS OF THE OPTIMAL NUMBER OF NEIGHBOURS IN A STATIC WIRELESS MULTI-HOP NETWORK

3.1 Overview

The literature review conducted in Chapter Two into research on the optimal number of neighbours has identified the lack of simulation using widely accepted simulation tools. Also lacking was the use of network topologies that leverage the arrangement of houses and buildings. The “magic numbers” as well as their upper and lower bounds were almost exclusively proven via mathematical modelling and the random distribution of the network nodes.

In this chapter we evaluate five practical wireless multi-hop network topologies, (refer to Fig. 1.4), in order to determine the required optimal number of neighbours. The results of this evaluation will be used as the lower bound for the optimal number of neighbours that our proposed topology control algorithm will be required to maintain.

3.2 Simulation Methodology

The overall goal of the experiments conducted was to determine the optimal minimum average node degree (number of neighbours) based on the network topologies shown in Fig.3. The performance of the network topologies was based on their ability to facilitate the routing protocol’s creation of routes to the intended destinations as well as the subsequent delivery of data. The performances of the selected networks were evaluated under "perfect" conditions (where none of the nodes failed) as well as under the scenario where randomly chosen critical nodes[4] were chosen to be disconnected[5] for a maximum period of 150s.

These evaluations are based on the simulation of 30 wireless nodes spread over a rectangular 1000m x 600m flat space for 900s of simulated time. The wireless nodes in this study were modelled on a Linksys WRT54G[6] version 2 wireless router (Linksys Inc, 2007) using ns-2. A copy of the simulation script where the Linksys WRT54G was modelled can be found in Appendix A and B. This particular router is popular amongst community-based wireless user groups around the world and deployments of this router as a wireless multi-hop network node (along with open-source firmware) span the globe (Meraka Institute, 2005), (Tibetan technology Centre, 2005), (Lancaster Mesh, 2006).

For relative comparisons between network topologies, identical network loads were applied to each topology. Ns-2 allows traffic loads to be pre-generated and used as input into the overall simulation model. Sixteen unique traffic loads were generated resulting in sixteen simulation runs per network topology with the number of source-destination pairs varying from 15 to 28.

All data was collected using purpose-written scripts as well as Tracegraph (Tracegraph, 2007), which is a tool for analysing the trace files generated by each

[4] Network nodes that are intermediaries for any two other nodes in the network. Their failure may result in the use of an alternate route or in the worst case, the failure to reach the intended destination.
[5] Term used to refer to node failure
[6] See Appendix D for Linksys WRT54G specifications

simulation run in ns-2. Only results that fell within a 90% confidence interval for the number of data packets sent are considered. It is anticipated that the use of the confidence interval will aid in the credibility of the results reported.

The following metrics were chosen to evaluate the relative performances of the selected wireless multi-hop network topologies. They are:

i. *Packet Delivery Ratio (PDR)*: the percentage of application layer packets containing unique packet IDs received at the intended destinations as well as the average packet delivery per second;

ii. *Routing Overhead*: the number of routing packets transmitted. Only unique packet IDs are taken into account despite the number of hops traversed, and

iii. *Average End-to-End Delay*: the delay experienced en-route from source node to destination node.

The results per network topology will be presented in the forthcoming sections. Since the aim of this experiment was to determine the optimal average node degree, the results for the network topologies will be manipulated to reflect the performance per node degree (refer to Table 3-1 for the manipulation from the network topology to the node degree).

3.3 Simulation Environment

The Network Simulator-2 (ns-2) (http://www.isi.edu/nanam/ns) (version 2.29 running on an Ubuntu Linux 6.06 LTS operating system) was chosen to conduct this study due to its support for the IEEE 802.11 standards with many subsequent patches/updates published by the ns-2 user community to improve the IEEE 802.11 simulation model (Marco Fiore, 2004). This has resulted in its popularity (Kurkowski, et al, 2005) within the wireless network and Mobile Ad Hoc Network (MANET) research communities.

In this section we describe the models of the various layers of the IEEE 802.11 protocol stack and the criteria for choosing the network topologies.

3.3.1 Physical and Data Link Layer Model

As mentioned earlier some of the updates provided for ns-2 help to model the noise experienced by wireless signals operating in the 2.4GHz band. For the purposes of this study we assume the use of omni-directional antennas with a gain of 4dBi resulting in a transmission range of approximately 120m when combined with the two-ray ground signal reflection model.

3.3.2 Medium Access Control

The simulation's link layer model is not completely based on the IEEE 802.11 MAC protocol defined in the standard. Our link layer model is instead based on the link layer that is implemented with most real-world IEEE 802.11 equipment (Linksys Inc, 2007). The MAC protocol defined in the IEEE 802.11 standard follows the Request-To-Send, Clear-To-Send, Data and Acknowledgement sequence. Commercial products are shipped with the request-to-send clear-to-send (RTS/CTS) protocol turned off by default, thereby resulting in its extremely limited use. Additionally, research shows that RTS/CTS impedes the performance of a wireless multi-hop network (Xu, et al, 2002), (Wu and Hou 2005).

3.3.3 Packet Buffering Model

Every wireless multi-hop network node in the simulation contains a buffer (queue) containing both data and control packets that are awaiting transmission. The buffer is able to accommodate 50 packets and implements the drop-tail queue management algorithm which requires minimal management. In addition, the queue is configured to afford a higher priority to the routing protocol's control packets.

3.3.4 Data Traffic Model

Constant bit rate (CBR) traffic sources were chosen to simulate the Application Layer communication between nodes in the network. Despite lacking realism (most real-world Application Layer traffic is of a bursty nature), it was deemed that the use of CBR traffic would not have impacted on the relative abilities of the network topologies being investigated to facilitate the delivery of the packets to their intended destinations.

The User Datagram Protocol (UDP) was chosen as the transport layer delivery protocol due to it's the minimal overhead created during the delivery process compared to the Transmission Control Protocol. Thus the CBR traffic sources are delivered using UDP's best-effort delivery model.

A sending rate of 4 packets per second was chosen with the number of CBR traffic sessions between source-destination pairs varying from 15 to 28. The traffic sessions end either when the simulation run ends (after a period of 900s) or a maximum of 1000 packets is transmitted. A minimal packet size of 64 bytes was employed.

3.3.5 Multi-Hop Network Routing Protocol

The Ad hoc On-Demand Distance Vector (AODV) (Perkins, et al, 2003) routing protocol was chosen to aid in the performance evaluation of the network topologies depicted in Fig. 3. Although AODV is an on-demand routing protocol, it does offer some advantages in stationary wireless multi-hop networks which includes:

i. Routes to destinations are created only when necessary;

ii. On-demand routing protocols typically react well to the link failures that invariably occur even in stationary networks;

iii. AODV favours less congested routes to their shorter counterparts, and

iv. Real-world implementations of the protocol exist and can be used in test-beds to validate the results obtained and presented in this work.

No changes were made to the default settings provided by ns-2 version 2.29 for the AODV routing protocol.

3.3.6 Criteria for Choosing Network Topologies

Two criteria for the wireless multi-hop network topologies chosen to be used in this study were identified. These criteria were:

Table -3-1 – Average node degree of each network topology

	Random Topology	Ribbon Topology	Spine Topology	Sparse Hex Topology	Square Grid Topology
Average Node Degree	3	2	2	6	4

i. The network topologies chosen must leverage the regular, uniform arrangement of buildings in a typical South African community (refer to Fig. 1.3), and

ii. A variety in the average number of neighbours was sought (thereby varying the levels of contention for the transmission medium). This requirement was necessary for determining the optimal average number of neighbours for a stationary wireless multi-hop network.

3.4 Simulation and Experiments Results

This section presents the results obtained from our investigation into the optimal number of neighbours in a wireless multi-hop network that leverages the layout of a typical South African community. The results for both the "perfect" and node failure scenarios are offered. These results are presented based on the network topology as well as the average node degree (number of neighbours). The average node degrees for each network topology can be found in Table 3-1. Both the Ribbon and Spine topologies have the same average node degree and thus the average of the two results will be used for an average node degree of 2. Note that the average node degrees are independent of the two scenarios being considered.

Table -3-2 – Application Layer packets per network topology per simulation run

	Random Topology		Ribbon Topology		Spine Topology		Sparse Hex Topology		Square Grid Topology	
	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection
Run 1	23636	23577	24695	23619	23588	23631	23593	23614	23878	23890
Run 2	13777	13725	13785	13758	13766	13738	13743	13748	13768	13750
Run 3	18255	18259	18240	18264	18268	18263	18251	18265	18254	18259
Run 4	15613	15629	15649	15642	15622	15605	15625	15617	15627	15607
Run 5	19963	19982	19958	19974	19972	19958	19959	19966	19963	19984
Run 6	16470	16445	16473	16466	16450	16470	16477	16475	16468	16446
Run 7	22360	22364	22341	22338	22344	22352	22370	22370	22354	22374
Run 8	19893	19860	19880	19881	19850	19897	19851	19892	19869	19861
Run 9	18570	18562	18576	18559	18528	18545	18541	18552	18554	18532
Run 10	11783	11785	11776	11796	11780	11764	11781	11770	11780	11769
Run 11	15447	15406	15430	15413	15411	15440	15432	15419	15430	15419
Run 12	15420	15409	15402	15416	15420	15407	15417	15418	15415	15425
Run 13	15413	15445	15422	15422	15457	15430	15429	15417	15430	15445
Run 14	19578	19561	19581	19571	19850	19578	19564	19591	19643	19656
Run 15	18342	18303	18274	18292	18303	18288	18299	18317	18305	18294
Run 16	24695	24664	24678	24704	24670	24689	24685	24707	24684	24678

As highlighted in the Methodology, only the results of those simulation runs that fell within the 90% confidence interval for the number of Application Layer packets sent were reported. The simulation runs for each network topology are identified in Table 3-2. The columns represent each of the network topologies being investigated whilst the rows represent the number of Application Layer packets sent during each simulation run for each network topology. The simulation runs that fell within the confidence interval are highlighted in grey.

3.4.1 Experiment 1: Packet Delivery Ratio (PDR)

The purpose of this experiment was to determine the network's ability to deliver the data packets being sent. A PDR of 0% represents the total failure of the network to deliver its data packets whilst a PDR of 100% shows that all the data packets in the network were delivered.

Table -3-3 – Results for the PDR per network topology per simulation scenario

	Random Topology		Ribbon Topology		Spine Topology		Sparse Hex Topology		Square Grid Topology	
	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection
Run 3	92.81	77.93	88.75	59.05	91.75	3.54	95.20	99.98	100.00	96.12
Run 9	94.96	85.58	87.86	65.47	90.85	16.51	99.99	96.45	100.00	97.43
Run 15	95.63	86.47	92.48	66.98	95.48	22.84	99.99	88.16	100.00	96.75
Average	**94.47**	**83.34**	**89.69**	**63.84**	**92.68**	**14.31**	**98.40**	**94.87**	**100.00**	**96.77**

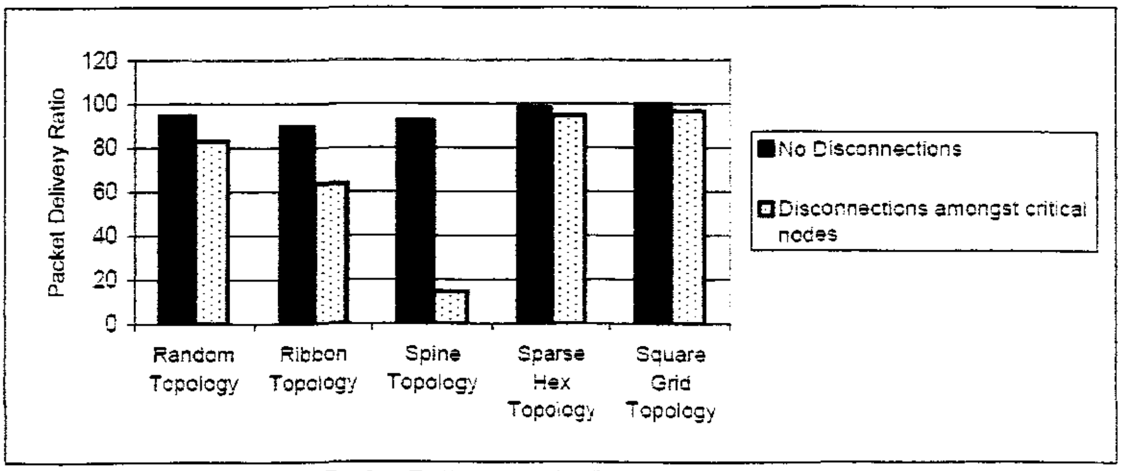


Fig. 3.1 – Packet Delivery Ratio (PDR) per network topology

Results

Table 3-3 shows the simulation results for the PDR that fell within the 90% confidence interval. The PDR was obtained as the percentage of Application Layer packets that arrived at their intended destinations during the course of the simulation runs. The results for both the "perfect" and node disconnection scenarios are displayed in Table 3-3. These results were plotted to produce Fig. 3.1 and then manipulated to determine the relationship between the PDR and the average node degree in Fig. 3.2.

Fig. 3.1 depicts the PDRs for the wireless multi-hop network topologies that were evaluated. It can be clearly seen that for the "perfect" scenario in which no node

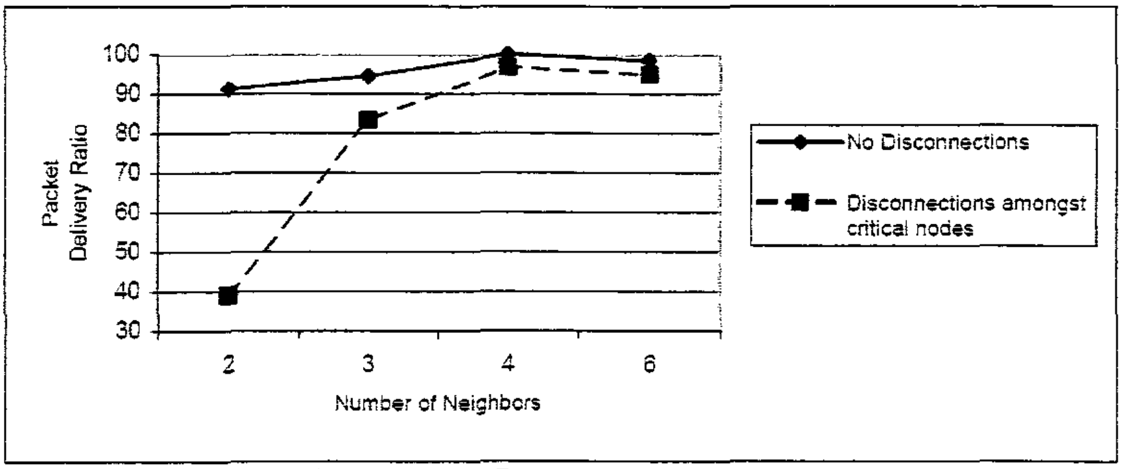


Fig. 3.2 – Packet Delivery Ratio versus the average node degree

disconnections are experienced, all of the network topologies perform well; delivering a minimum of 89.69% of all the Application Layer packets that were sent.

The node disconnection scenario showed that some network topologies handled the disconnections better than others did. A general trend emerged where the topologies with the higher average node degrees (such as the Square Grid and Sparse Hex topologies) emerged with better PDRs than those with lower average node degrees (such as the Random, Ribbon and Spine topologies). This trend is clearly illustrated in Fig. 3.2.

The reason for this observed trend is the ability to find an alternate route if the route that was being utilised was disrupted due to a node disconnection. The network topologies with the higher node degrees tend to have multiple routes from any source node to any destination node thereby ensuring the redundancy of the network and its ability to deliver packets to their destinations despite node disconnections.

An additional observation was the average node degree of 4 delivered the most Application Layer packets in both the "perfect" and node disconnection scenarios.

Table -3-4 – Results for the Routing Overhead per network topology per simulation scenario

	Random Topology		Ribbon Topology		Spine Topology		Sparse Hex Topology		Square Grid Topology	
	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection	"Perfect"	Node Disconnection
Run 3	762	5756	234	2578	262	5198	1501	1642	1862	3726
Run 9	706	3994	222	1889	249	4256	1962	4172	2367	4763
Run 15	763	2483	249	2167	279	3530	1740	4224	2471	5002
Average	**743**	**4198**	**235**	**2212**	**263**	**4328**	**1734**	**3346**	**2233**	**4497**

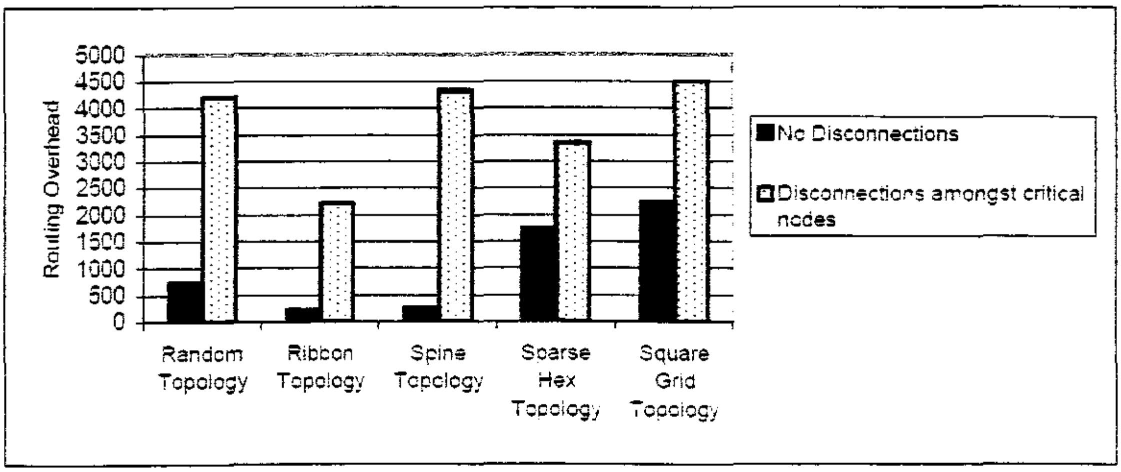


Fig. 3.3 – Routing Overhead per network topology

3.4.2 Experiment 2: Routing Overhead

The purpose of this experiment was to determine the amount of routing protocol overhead that was created to ensure the PDR that was achieved in the experiments. The routing overhead was measured as the total number of unique AODV control messages that were sent during the simulation run.

Results

The results for both the "perfect" and node disconnection scenarios are presented in Table 3-4. The graph in Fig. 3.3 was produced and then manipulated to determine the relationship between the Routing Overhead and the average node degree in Fig. 3.4.

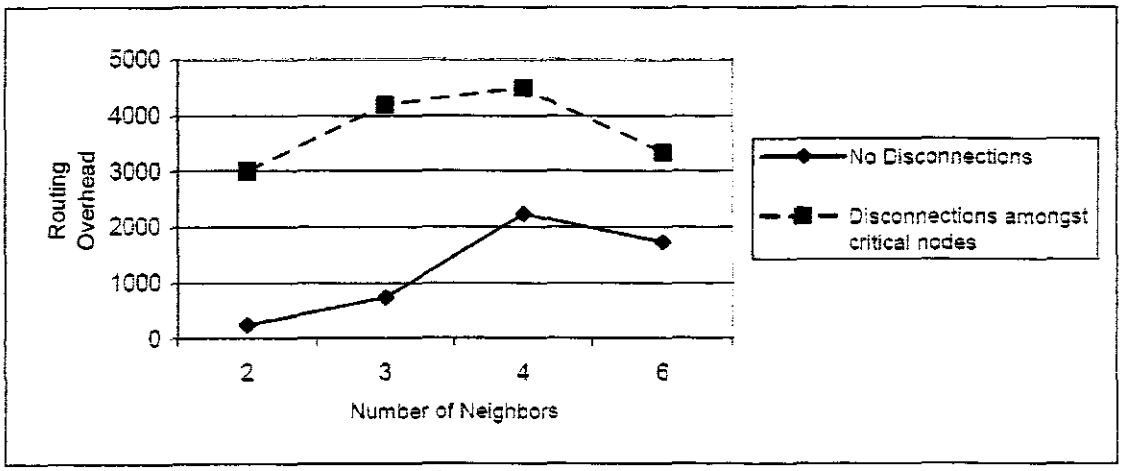


Fig. 3.4 – Routing Overhead versus the average node degree

Fig. 3.3 depicts the routing overhead generated by the routing protocol during the delivery of the Application Layer packets between source nodes and destination nodes in the network topologies that were evaluated.

The "perfect" scenario illustrated that the networks with the lower average node degrees generated the least routing overhead whilst those networks with higher average node degrees generated the most routing overhead. The amount of routing overhead generated was related to the redundancy of the network. Networks with lower redundancy issued fewer route requests due to there being a greater probability that one of the intermediary nodes would already possess the required routing information since all traffic has to pass through a select few of these intermediary nodes. Fig. 3.4 bears testimony to this observed phenomenon.

The node disconnection scenario showed that the network topologies with lower average node degrees experienced a large increase in the amount of routing overhead generated. Fig. 3.3 presents the spike in routing protocol activity that the nodes with the lower average node degrees experienced in this scenario. The percentage increases for the Random, Ribbon and Spine topologies were recorded at 565%, 941% and

1646% respectively. This is a direct result of the comparative lack of redundancy in these networks. The increases in the routing overhead of these network topologies comprised additional route requests as well as the route error messages that were generated as a result of the node disconnections.

The networks with higher average node degrees reacted better to the node disconnection scenario with the Sparse Hex and Square Grid topologies reporting increases of 193% and 201% respectively (based on a comparison to the "perfect" scenario). These comparatively low increases (also illustrated in Fig. 3.4) are as a result of the superior redundancy levels that these two network topologies enjoy and the increase in overhead is comprised mainly of the additional route requests generated as a result of the node disconnections.

3.4.3 Experiment 3: End-to-End Delay

The purpose of this experiment was to determine the average time taken to deliver the Application Layer packets from the source node to the intended destination node.

Results

The results for both the "perfect" and node disconnection scenarios are displayed in Table 3-5. These results were plotted to produce Fig. 3.5 and then manipulated to determine the relationship between the End-to-End Delay and the average node degree in Fig. 3.6. Note that the results are reported in milliseconds.

Table -3-5 – Results for the End-to-End Delay per network topology per simulation scenario

	Random Topology		Ribbon Topology		Spine Topology		Sparse Hex Topology		Square Grid Topology	
	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection
Run 3	6.6	115.1	12.0	221.0	9.2	293.9	3.4	2.6	2.4	25.4
Run 9	5.0	104.5	12.4	284.8	9.8	91.0	3.4	36.1	2.8	20.6
Run 15	5.2	120.6	30.0	252.6	9.2	130.3	3.2	35.1	2.8	19.7
Average	**5.6**	**113.4**	**18.2**	**252.8**	**9.4**	**171.7**	**3.4**	**24.6**	**2.7**	**21.9**

Table -3-6 – Average Path Length per network topology per simulation scenario

	Random Topology		Ribbon Topology		Spine Topology		Sparse Hex Topology		Square Grid Topology	
	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection
Path Length (hops)	7	**6**	**15**	**11**	**11**	7	**5**	**5**	**5**	**5**

Fig. 3.6 plots the end-to-end delay experienced during the process of delivering the Application Layer packets. This process cost includes the time taken to establish a route as well as the actual delivery of the Application Layer packets, via intermediate nodes, to their intended destinations. The results obtained for the “perfect” scenario in which there are no disconnections shows that there is some proportionality relationship between the path lengths (refer to Table 3-6) and the end-to-end delays experienced. This relationship was also found to be true for the node disconnection scenario as well.

The average node degree affected the end-to-end delay experienced as illustrated in Fig. 3.6. The networks with lower average node degrees experienced greater end-to-end-delays than those with higher average node degrees. This situation became much more acute when node disconnections were taken into account. The Random, Ribbon and Spine topologies experienced a 2025%, 1389% and 1827% increase respectively

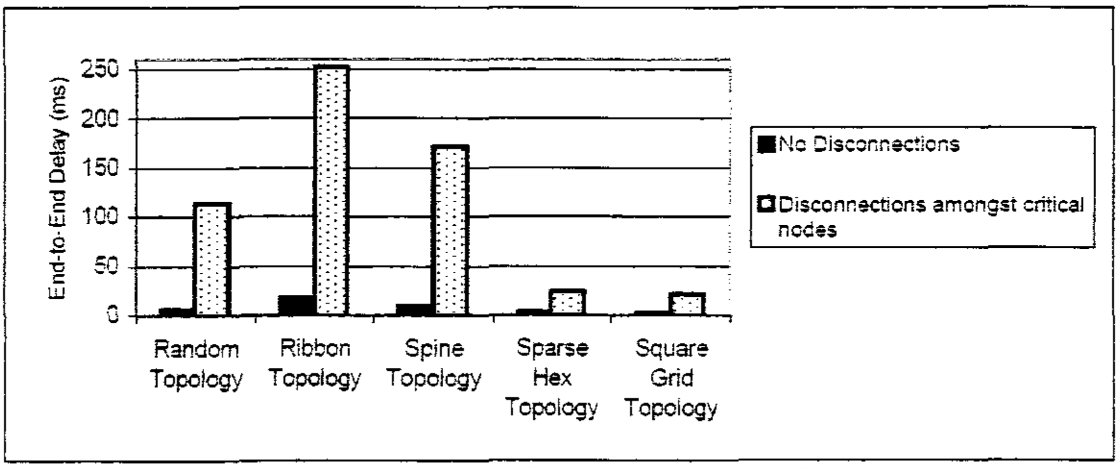


Fig. 3.5 – End-to-End Delay per network topology

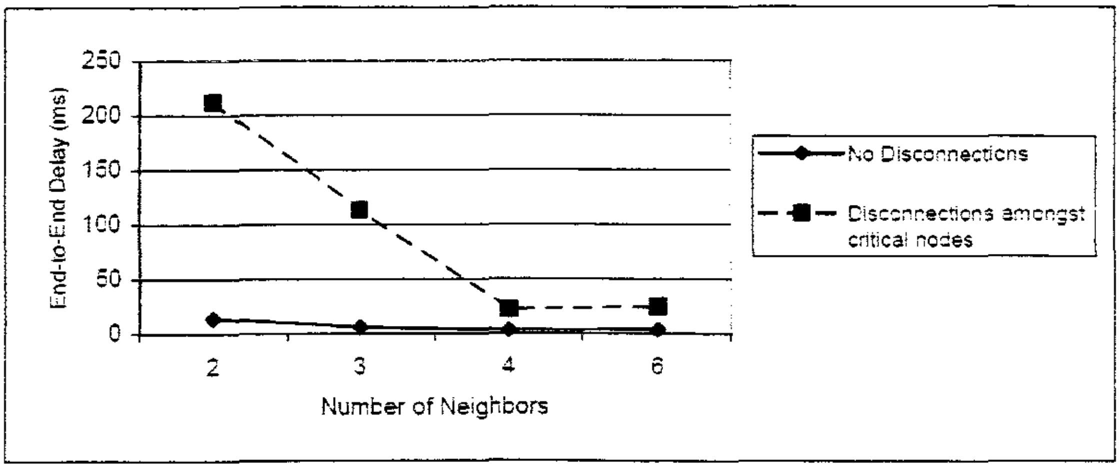


Fig. 3.6 – End-to-End Delay versus the average node degree

in the reported delay. This drastic increase was attributed to the use of more congested (although shorter[7]) alternate routes to the intended destination and the time taken to request new routes (this includes the multiple retries afforded by the AODV routing protocol). The cumulative effect of multiple simultaneous transmissions only exacerbated the delay experienced.

The networks with the higher average node degrees experienced substantially lower increases in the end-to-end delay experienced (clearly indicated in Fig. 3.6). The Sparse Hex and Square Grid topologies experienced increases of 724% and 811%

[7] AODV chooses the least congested route instead of the shortest route since the shortest route is not always the fastest route.

respectively when compared to the "perfect" scenario. These two networks benefited from their high levels of route redundancy as well as the fact that the average path lengths remained unchanged despite the use of alternate routes to the intended destinations.

3.5 Simulator and Experimental Limitations

Simulation experiments are at best an approximation of the real world. Thus there are bound to be assumptions made in an effort to model the environment being considered. This section highlights the assumptions made, the limitations on the experiments conducted as well as all inherent limitations of the simulation tool that was utilised. It is acknowledged that one or more of the assumptions made and the limitations of the experiments and simulation tool could have affected the results presented.

The assumptions and limitations are:

i. Lack of realistic Application Layer modelling;

   A constant bit rate model was utilised whereas realistic Application Layer traffic resembles a variable bit rate traffic stream.

ii. The terrain was assumed to be flat with no obstacles

   Realistic terrain models consider the elevation of the nodes as well objects such as trees, etc;

iii. The nodes in the Ribbon, Spine, Sparse Hex and Square Grid topologies were uniformly spaced, and

iv. The nodes in the network were stationary.

Table -3-7 – Performance Summary

	Number of Neighbours (node degree)							
	2		**3**		**4**		**6**	
	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection	“Perfect”	Node Disconnection
PDR	1	1	2	2	4	4	3	3
Routing Overhead	4	4	3	2	1	1	2	3
Delay	1	1	2	2	4	4	3	3
Score	**6**	**6**	**7**	**6**	**9**	**9**	**8**	**9**

v. Power consumption was not considered as a factor

vi. All the nodes employ the same transmission power

vii. The IEEE 802.11 RTS/CTS mechanism was disabled

3.6 Summary

Table 3-7 summarises the performances of the network topologies surveyed with regards to their average node degrees. Ratings range from 1 to 4 with 1 representing the worst performance and 4 representing the best performance. Table 3-7 helps to determine the range for the optimum minimal node degree. From the experiments reported in Sections 3.4.1 until 3.4.3 as well as the concise summary provided in Table 3-7, we can recommend that the minimum number of neighbours that a potential topology control algorithm should maintain must be at least 4. The difference in the performances of networks (as highlighted by the ratings) with average node degrees less than 4 highlight the crucial role that topology control algorithm play in maintaining the optimal network performance.

Based on these findings the Token-based Topology Control algorithm will be designed to maintain a minimum of 4 nearest neighbours whenever possible. This finding considers the problem posed at the end of Section 2.8 and completes the list of design criteria presented in the same section. The finding of the optimal node degree of 4, determined in this chapter, agrees with an existing result reported in (Wan and Yi, 2004) for the same total number of nodes.

CHAPTER FOUR

TOKEN-BASED TOPOLOGY CONTROL (TbTC)

4.1 Overview

The application of tokens in networking is not a new phenomenon and can be traced back to the late 1960s with the introduction of Token Bus (IEEE 802.4) and Token Ring (IEEE 802.5) networks (Tomsho, et al, 2002). The purpose of the token was to reduce the collisions experienced when two or more terminals on a network proceeded to transmit data simultaneously by allowing only the terminal in possession of the token to transmit. This characteristic of token-based networks allowed for the synchronisation of communications on the network and is also useful for the synchronisation of the communication (between neighbouring multi-hop network nodes) necessary during the process of performing a topology control algorithm.

Thus, the purpose of introducing a token into the topology control process is to ensure that the neighbouring nodes of a node that possesses the token are not busy executing their own instances of the topology control algorithm concurrently. This will avoid a situation where neighbouring nodes fail to respond to the requests for information because they themselves are awaiting the responses to their own requests, thereby leading to the creation of sub-optimal network topologies or even deadlock.

A topology control algorithm that incorporates the use of a token, entitled "Token-based Topology Control" is presented in this chapter. The algorithm has been designed with the necessary criteria identified in the related literature as well as the

Table -4-1 – Characteristics of Token-based Topology Control

Algorithm Characteristics	**Value**
Node characteristics	heterogeneous
Information Requirements	Neighbour-based
Architecture	Distributed
Link characteristics	Both directional and bidirectional
Connectivity flexibility	*k*-connectivity, where $k >= 4$
Suitability for low-cost community-based networks	Suitable due to the provision of minimal computation, minimal communication overhead and synchronicity between neighbours via the use of a token.

optimal average node degree identified in Chapter Three. A brief overview of our algorithm (based on the framework developed in Section 2.5.2) is given in Table 4-1.

Details of the algorithm as well as the experiments conducted to measure the effectiveness of the algorithm are presented in subsequent sections of this chapter. This chapter ends with the observed limitations of our Token-based Topology Control algorithm.

4.2 Design Criteria and Assumptions

Section 2.8 listed the design criteria that were identified during the review of the related literature. In this section we provide the motivation for our choice of design criteria and subsequently detail any assumptions made. The design criteria discussed below are not addressed in any particular order and no inferences should be made as to the position of individual criterion.

4.2.1 Distributed Algorithm

The choice of a distributed algorithm alleviates the need for instituting a process for the election of a central node that performs the topology control algorithm. This process entails the call for suitable candidate central nodes and the subsequent notification of all the nodes in the network of the identity of the elected central node. The process outlined above generates excessive communication overhead which may affect the network's delivery of data.

A distributed algorithm is favoured because it also avoids the latency that is incurred when a network node has to wait for the return of the processed data from the central node in order to adjust its own transmission power. This latency reduces a centralised algorithm's ability to respond to dynamic changes in the network topology.

4.2.2 Minimal Computation

Most previously proposed topology control algorithms are based on the manipulation of graphs in order to determine the optimal network topology via the optimisation of the network graph. The computations required for these optimisations are beyond the computational capacity of the low cost, resource-constrained wireless multi-hop network nodes being considered.

The proposed algorithm will only require the ability to compute the optimal number of neighbours based on their relative distances which are determined via the received power levels detected by the hardware and inserted into the higher layer packets.

4.2.3 Node degree information requirements

Some of the previously proposed topology control algorithms required Global Positioning System (GPS) capabilities, or the relative directions from a node to its neighbours. These information requirements are not feasible for a topology control algorithm that is designed for use with low cost, resource-constrained network nodes. The network nodes being considered do not possess GPS capabilities, as it would increase the cost of ownership of the node, neither do they possess the ability to determine the relative directions of neighbouring nodes.

The network nodes being considered do possess the ability to count the number of neighbours (node degree) and use this information as input into the proposed topology control algorithm.

4.2.4 Heterogeneous Transmission Radius

The requirement for heterogeneous transmission radii minimises the total power consumption of the network, whilst also minimising the interference experienced by nodes in the network. A heterogeneous transmission radius allows each node in the network to optimally adjust its transmission power so that it is able to communicate with the required number of neighbours. As a result the levels of interference are reduced to the minimum determined by the levels of redundancy maintained.

4.2.5 Maintenance of network connectivity

Despite topology control's purpose to reduce interference and maximise throughput whilst maintaining the optimal levels of redundancy in the network, an often implicit requirement is that the network topology remains connected. A partitioned network topology must be avoided because it trivialises the effectiveness of any topology control algorithm as no traffic can be routed between the partitioned elements of the network.

4.2.6 Synchronicity between neighbours

The correct execution of a topology control algorithm involves the exchange of messages between a node and its neighbours in order to collect the necessary information. Previously proposed topology control algorithms avoided provision for situations, where two or more neighbouring network nodes were in differing stages of the concurrent execution of their own instances of the topology control algorithm. In the situation described above, where no form of synchronicity is provided, a node could issue a request for information and receive no reply because its neighbouring nodes were busy executing their own instances of the topology control algorithm. This leads to the incomplete collection of information and the subsequent creation of a sub-optimal wireless multi-hop network topology.

4.3 Proposed Algorithm

The Token-based topology control algorithm can be decomposed into three components, each contributing to the overall functionality of the algorithm. These

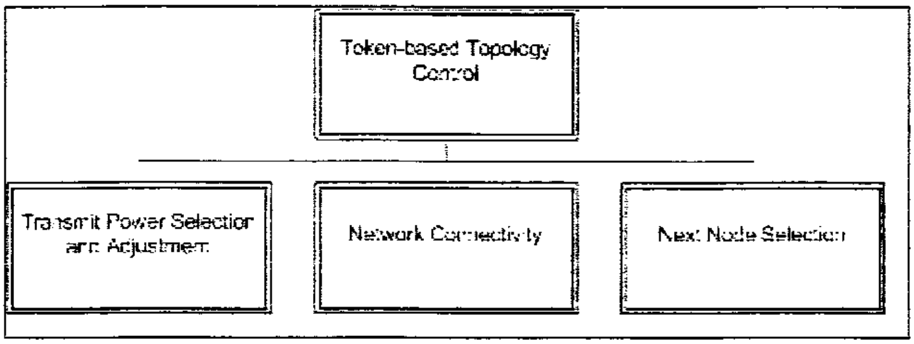

Token-based Topology Control
Transmit Power Selection and Adjustment
Network Connectivity
Next Node Selection

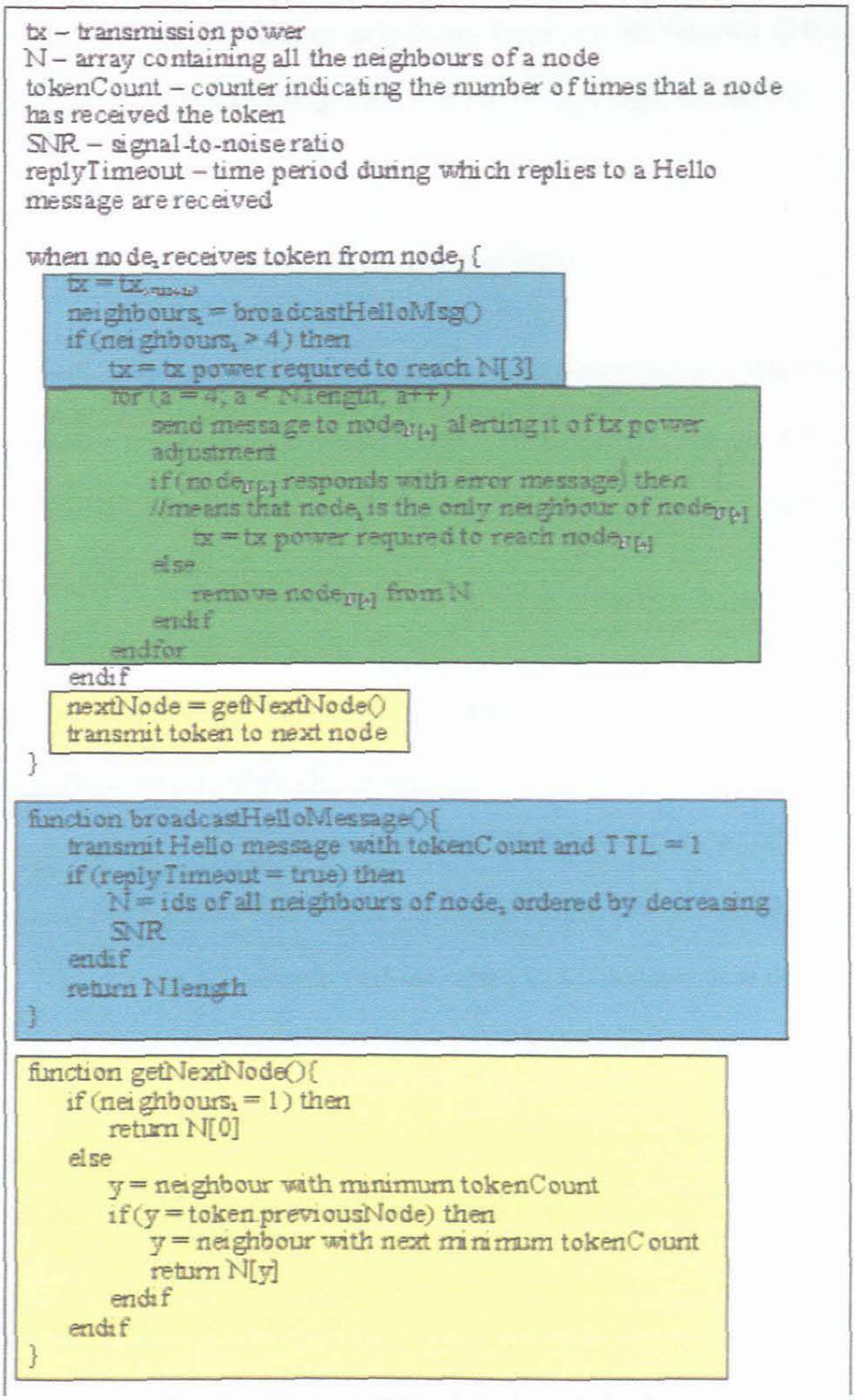
Fig. 4.1 – Components of the Token-based Topology Control Algorithm

three components are: the *transmit power selection and adjustment* component, the *next node selection* component, and the *network connectivity* component (refer to Fig. 4.1). Each component is discussed separately in the order in which they are likely to be invoked.

4.3.1 The Transmit Power Selection and Adjustment Component

The most important aspect of any topology control algorithm is the ability to adjust the transmission powers of the nodes in the network. The Token-based Topology Control algorithm selects the appropriate transmission power per node by limiting the number of neighbouring nodes that are within transmission range to a minimum of four wherever possible.

Each node in the network initially uses their maximum transmission power. The optimal transmission power is selected based on the signal-to-noise ratio (SNR) that is detected from the replies to a broadcast Hello message. The optimal transmission power is the power required to reach the neighbouring node with the fourth-strongest SNR, dependent on the node having four or more neighbours. The transmission power is then subsequently adjusted to this optimal level.



```
tx – transmission power
N – array containing all the neighbours of a node
tokenCount – counter indicating the number of times that a node
has received the token
SNR – signal-to-noise ratio
replyTimeout – time period during which replies to a Hello
message are received

when $node_a$ receives token from $node_j$ {
    tx = $tx_{max}$
    $neighbours_a$ = broadcastHelloMsg()
    if ($neighbours_a$ > 4) then
        tx = tx power required to reach N[3]
        for (a = 4; a < N.length; a++)
            send message to $node_{N[a]}$ alerting it of tx power
            adjustment
            if ($node_{N[a]}$ responds with error message) then
            //means that $node_a$ is the only neighbour of $node_{N[a]}$
                tx = tx power required to reach $node_{N[a]}$
            else
                remove $node_{N[a]}$ from N
            endif
        endfor
    endif
    nextNode = getNextNode()
    transmit token to next node
}

function broadcastHelloMessage(){
    transmit Hello message with tokenCount and TTL = 1
    if (replyTimeout = true) then
        N = ids of all neighbours of $node_a$ ordered by decreasing
        SNR
    endif
    return N.length
}

function getNextNode(){
    if ($neighbours_a$ = 1) then
        return N[0]
    else
        y = neighbour with minimum tokenCount
        if (y = token.previousNode) then
            y = neighbour with next minimum tokenCount
            return N[y]
        endif
    endif
}
```

Fig. 4.2 – Token-based Topology Control algorithm

The process of selecting the optimal transmission power requires the broadcasting of a Hello message by a node$_a$ possessing the token. All neighbouring nodes that receive the broadcast reply to it with their individual token counts inserted into the replies.

Node$_a$ determines the four-nearest neighbours based on the detected SNRs of the reply messages. This process is highlighted in blue in both Figs. 4.2 and 4.3.

4.3.2 The Network Connectivity Component

The transmission power selection and adjustment component ensured that the number of neighbours of a node never exceeded four wherever possible, but this is not enough to ensure that the new network topology remains connected (a route exists between every source and every destination in the network).

The network connectivity component contacts each node within the original maximum transmission range that would not fall within the adjusted transmission range and attempts to determine whether these nodes have other alternate neighbouring nodes within their own transmission ranges. This process, highlighted in green in Figs. 4.2 and 4.3, aims to find and retain nodes that rely only on node$_a$ as an intermediate relay node, thereby maintaining the network connectivity.

4.3.3 The Next Node Selection Component

This component is invoked once node$_a$ has successfully adjusted its own transmission power and must pass the token to the next recipient. The aim of this component is to ensure only one node in a local neighbourhood possesses a token and therefore only one node in a local neighbourhood is able to execute its topology control algorithm.

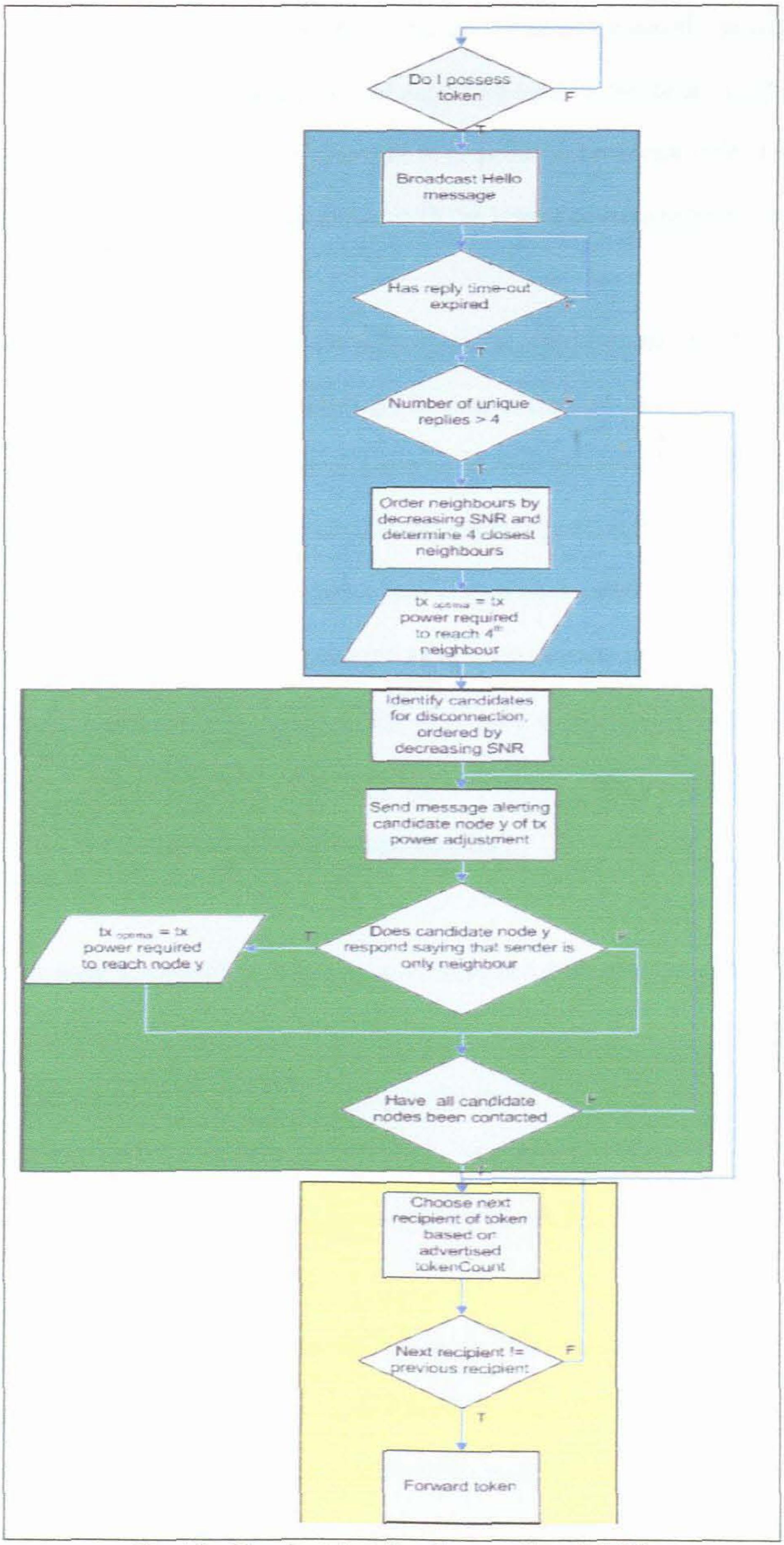

Do I possess token
F
T
Broadcast Hello message
Has reply time-out expired
F
T
Number of unique replies > 4
F
T
Order neighbours by decreasing SNR and determine 4 closest neighbours
tx optimal = tx power required to reach 4th neighbour
Identify candidates for disconnection, ordered by decreasing SNR
Send message alerting candidate node y of tx power adjustment
Does candidate node y respond saying that sender is only neighbour
T
F
tx optimal = tx power required to reach node y
Have all candidate nodes been contacted
F
T
Choose next recipient of token based on advertised tokenCount
Next recipient != previous recipient
F
T
Forward token

Fig. 4.3 – Flowchart depicting the execution of TbTC

The next recipient of the token is chosen by the node that currently possesses the token. The Token-based Topology Control algorithm records the token count of all of its neighbours by examining the replies sent in response to broadcast Hello messages. The next recipient chosen is the neighbour with the lowest advertised token count.

So how do we ensure that the next recipient is not the previous recipient of the token? A disregard for such a situation hinders the fairness of the next node selection process because the potential exists for some nodes to receive the token significantly more often than other nodes. TbTC implements the process of "neighbour control" where the solution lies within the token itself, which is imprinted with the identity of the last node visited prior to visiting the current node. The current recipient of the token determines the next recipient as described above and compares the identities of the previous recipient and the next selected recipient. The next recipient is only valid if its identity does not match that of the previous recipient.

The "next node selection" component in conjunction with the "neighbour control" process (highlighted in yellow in Figs. 4.2 and 4.3) ensures fairness by not selecting the previous recipient of the token as the next recipient unless the current recipient of the token has only one neighbour. In such a situation our restriction on the next recipient must be relaxed in order to guarantee the circulation of the token.

Fig. 4.4 depicts the sequence of events that occur when Node1 decides that Node3 is one of its 4 closest neighbours and Node2 is not. Before Node1 reduces its transmission power to break direct contact with Node2, Node1 needs to ensure that Node2 will not be disconnected from the network as a result of the transmission

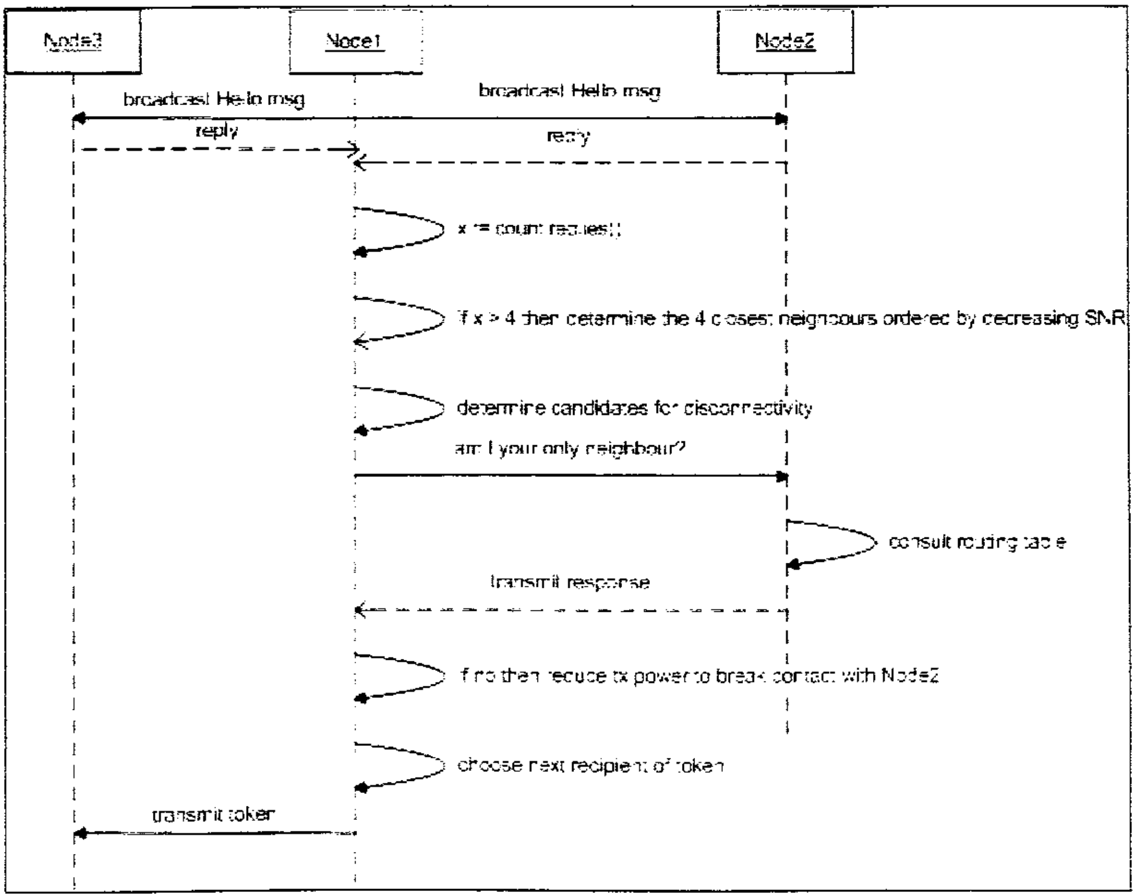

Node3
Node1
Node2
broadcast Hello msg
broadcast Hello msg
reply
reply
x = count requests()
if x > 4 then determine the 4 closest neighbours ordered by decreasing SNR
determine candidates for disconnectivity
am I your only neighbour?
consult routing table
transmit response
if no then reduce tx power to break contact with Node2
choose next recipient of token
transmit token

Fig. 4.4 – Sequence diagram depicting the execution of TbTC

power adjustment. The sequence diagram in Fig. 4.4 also highlights the lack of intensive computation which was one of the design criteria discussed in Section 4.2.

The TbTC algorithm and the design criteria used in its creation were presented in this section. The algorithm was decomposed in three components that must be invoked sequentially in order from left to right (refer to Fig. 4.1) to ensure the correct execution of the algorithm.

Note that the TbTC algorithm depicted in Fig. 4.2 can be easily converted to an object-oriented form via the conversion of the algorithm into a Node class. This class will contain the abstraction of the data used by TbTC (TbTC's attributes) in the form

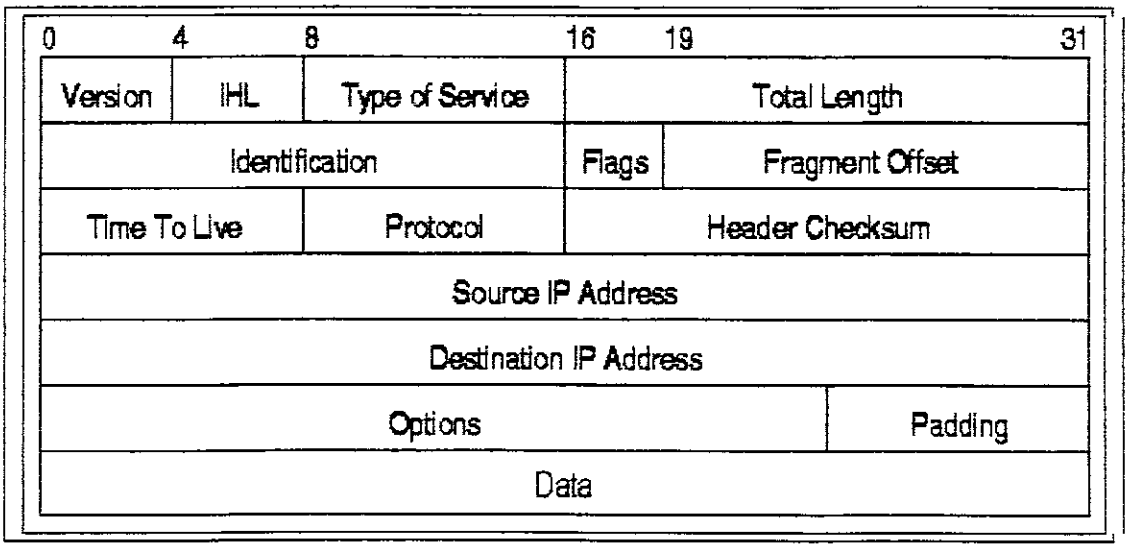


Fig. 4.5 – Internet Protocol (IP) Packet Structure

of *get* and *set* methods as well as the conversion of the pre-existing functions into methods defining the node's behaviour when in possession of the token.

4.4 The Packet Structure of the Token Employed

The TbTC algorithm is heavily dependent on the token packet. Thus the token packet must be readily distinguishable from both data packets and the control packets generated by the routing protocol employed. An additional criterion that the token packet must adhere to is that it should be lightweight in nature, thereby ensuring that it is not broken up into smaller packets for easier transmission.

The token packet employed in TbTC is based on the Internet Protocol (IP) packet structure as depicted in Fig. 4.5, with minor modifications made to distinguish the token from normal IP packets. A discussion of the modified IP packet fields follows.

The *Type of Service* field is 8 bits long and provides an indication of the desired quality of service. The major choice is a trade-off between low delay, high reliability

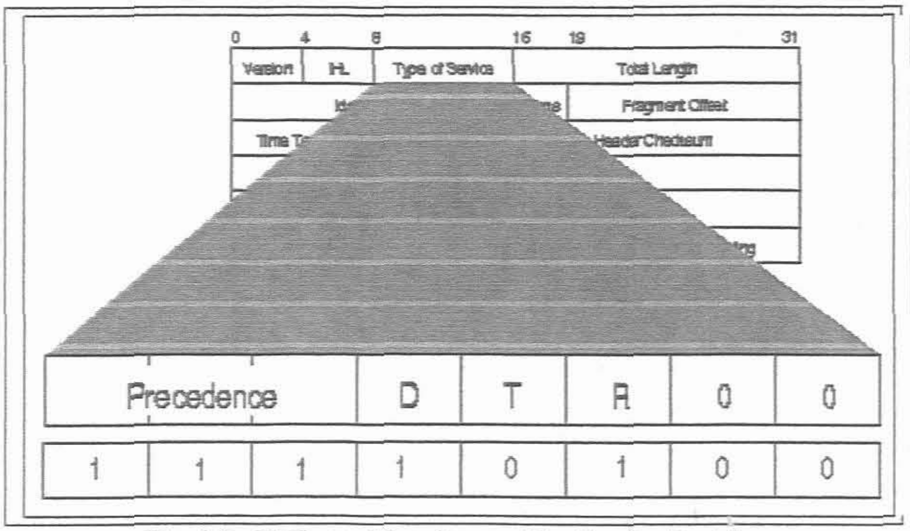


Fig. 4.6 – IP Type of Service specification for the token

and high throughput. The first three bits (0 - 2) of this field stores the precedence value. Bit 3 is set to indicate low delay. Bit 4 is set to indicate high throughput and Bit 5 is set to indicate high reliability. Bits 6 and 7 are reserved for future use. The 3-bit precedence options are:

000	Routine	**100**	Flash Override
001	Priority	**101**	CRITIC/ECP
010	Immediate	**110**	Internetwork Control
011	Flash	**111**	Network Control

Fig. 4.6 shows the settings of the Type of Service field in the token packet.

The *Total Length* field is 16 bits long and represents the length of the datagram.

The *Flags* field is 3 bits long and contains three control flags. Bit 0 is reserved, Bit 1 is set to indicate that the datagram should not be fragmented and Bit 2 is set to indicate whether more fragments of the same datagram exist. The flags options are:

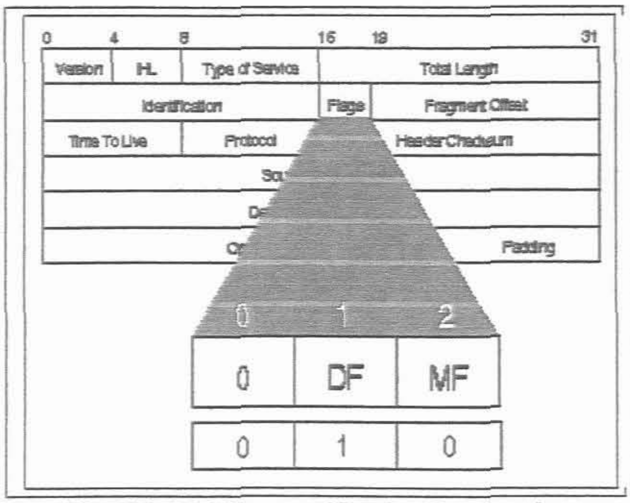


Fig. 4.7 – IP Flags specification for the token

Bit 0: reserved, must be zero

Bit 1 0 = May Fragment 1 = Don't Fragment

Bit 2 0 = Last Fragment 1 = More Fragments

Fig. 4.7 shows the settings of the Flags field in the token packet

The *Time to Live* field is 8 bits long and represents the maximum time that the datagram is allowed to reside in the network. This field is set to 1 to ensure that every node that receives the token regenerates it, setting the time to live to 1, before passing it on.

The *Source IP Address* field is 32 bits long and identifies the node that is sending the token.

The *Destination IP Address* field is 32 bits long and identifies the node that is receiving the token.

The *Data* field represents the payload of the IP datagram and TbTC uses this field to store the token count of the node that is sending the token. The token count is used to ensure that the wireless multi-hop network nodes are visited as fairly as possible.

This section identified the modifications made to the IP header to distinguish the token from normal IP datagrams. In the next section the performance of TbTC is analysed.

4.5 Performance Analysis of Token-based Topology Control Algorithm

This section details the analysis undertaken to determine the effectiveness of the Token-based Topology Control Algorithm. The simulation methodology used is described before the results obtained are presented and the limitations of both the simulator and experiments are highlighted. The results obtained with TbTC are for the network after the token has visited all the network nodes at least once.

4.5.1 Simulation Methodology

The simulation methodology employed is similar to that employed in the experiments conducted in Chapter Three. The evaluations conducted were based on the simulation of 30 wireless nodes spread randomly over a rectangular 1000m x 600m flat space for 900s of simulated time. The wireless nodes in this study were modelled on a Linksys WRT54G version 2 wireless router[8] (Linksys Inc, 2007). This particular router is popular amongst community-based wireless user groups around the world and

[8] See Appendix C for the ns-2 simulation script used

deployments of this router as a wireless multi-hop network node (along with open-source firmware) span the globe (Meraka Institute, 2005), (Tibetan technology Centre, 2005), (Lancaster Mesh, 2006).

The simulations were performed on networks with no form of topology control as well as networks that employed the Token-based Topology Control algorithm. For relative comparisons between the two sets of networks, identical network loads were applied to each set of networks. Ns-2 allows for traffic loads to be pre-generated and used as input into the overall simulation model. Sixteen unique traffic loads were generated resulting in sixteen simulation runs per network with the number of source-destination pairs varying from 15 to 28.

All data was collected using purpose-written scripts as well as Tracegraph (Tracegraph, 2007) which is a tool for analysing the trace files generated by each simulation run in ns-2. Only results that fell within a 90% confidence interval for the number of data packets sent are considered. It is anticipated that the use of the confidence interval will aid in the credibility of the results reported. Confidence intervals are used due to the inherent imprecision of a single simulation run. The Central Limit Theorem can be used to calculate the confidence interval when multiple simulation runs are considered. In addition to providing an expected range, the confidence interval can help determine if the two data sets are statistically equivalent.

The following metrics were chosen to evaluate the relative performance of a network before and after the Token-based Topology Control algorithm can be applied to it. They are:

i. *Packet Delivery Ratio (PDR)*: the percentage of application layer packets containing unique packet IDs received at the intended destinations as well as the average packet delivery per second;

ii. *Routing Overhead*: the number of routing packets transmitted. Only unique packet IDs are taken into account despite the number of hops traversed;

iii. *Average End-to-End Delay*: the delay experienced en-route from source node to destination node;

iv. *Power Consumption*: the cumulative power consumption when transmitting packets on the network. The power consumed during the processing of packets is disregarded, and

v. *Network Traversal (Hop Count)*: this metric measures the number of hops taken before all the nodes in the network have been visited by the token at least once.

4.5.2 Simulation Results

This section presents the results obtained from our investigation into the effectiveness of the TbTC algorithm. The results for both the networks that did and did not employ the algorithm are offered. As highlighted in the Methodology, only the results of those simulation runs that fell within the 90% confidence interval for the number of Application Layer packets sent are reported.

The simulation runs for each network type are identified in Table 4-2. The columns represent each of the network types being investigated whilst the rows represent the number of Application Layer packets sent during each simulation run for each

Table -4-2 – Application Layer packets per simulation run

	Without TbTC	With TbTC
Run 1	23627	23615
Run 2	13731	13736
Run 3	18252	18246
Run 4	15648	15631
Run 5	19959	19963
Run 6	16490	16473
Run 7	22327	22304
Run 8	19900	19893
Run 9	18572	18550
Run 10	11784	11776
Run 11	15431	15420
Run 12	15403	15396
Run 13	15414	15441
Run 14	19595	19580
Run 15	18296	18328
Run 16	24712	24698

network type. The simulation runs that fell within the confidence interval are highlighted in grey.

4.5.2.1 Experiment 1: Packet Delivery Ratio

The purpose of this experiment was to determine the network's ability to deliver the data packets being sent. A PDR of 0% represents the total failure of the network to deliver its data packets whilst a PDR of 100% shows that all the data packets in the network were delivered.

Table -4-3 – PDRs without and with TbTC

	Without TbTC	**With TbTC**
Run 3	20441(86.52%)	12246 (67.12%)
Run 9	18571 (99.99%)	7611 (41.03%)
Run 15	16296 (89.07%)	6679 (36.44%)
Average	**17039 (92.74%)**	**8845 (48.14%)**

Results

Table 4-3 shows the PDRs for the network before and after the TbTC algorithm can be applied to it. This data is plotted to produce Fig. 4.8.

Fig. 4.8 shows a sharp decrease in the ability of the network to deliver data packets to their intended destinations. This finding seems inconsistent with the results obtained in Chapter Three where we showed that the best performing wireless multi-hop network was one that had an average node degree of 4[9]. The sharp decrease can be attributed to TbTC's creation of uni-directional links and the routing protocol's reaction to such situations.

A side-effect of the TbTC algorithm is the possibility for the creation of unidirectional links due to the distributed nature of TbTC. Since TbTC has no centralised management and control, each node in the network is responsible for its own transmission power adjustment such that the objectives of TbTC are met. This situation results in the creation of a network with heterogeneous transmission power assignment amongst the nodes that constitute the network, resulting in the lack of guarantees that all the links in the network will be bi-directional.

[9] The TbTC algorithm was based on the results obtained in Chapter Three

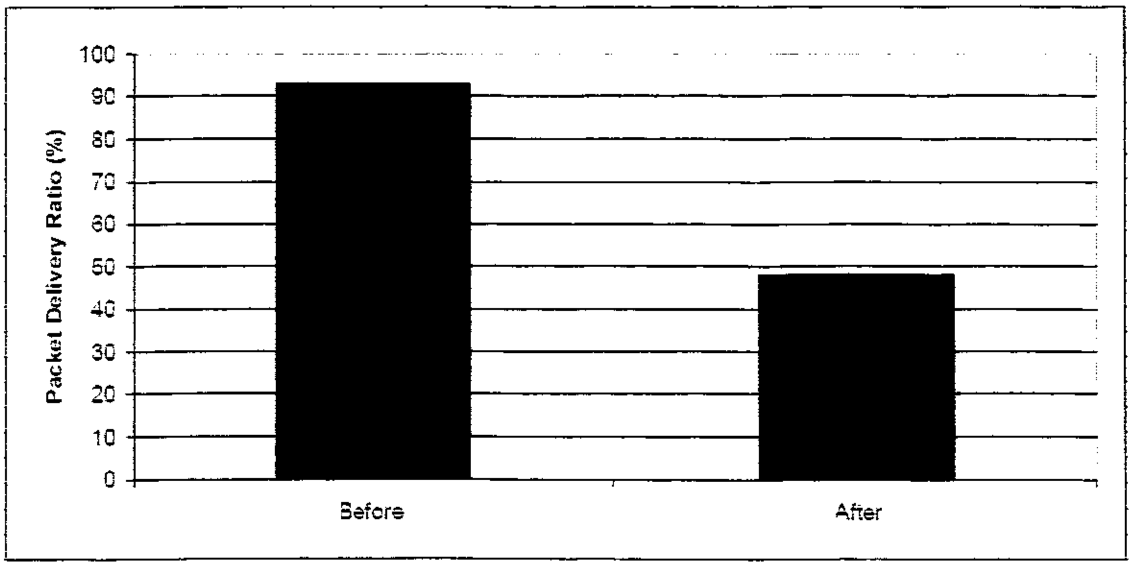


Fig. 4.8 – PDRs before and after TbTC

The AODV routing protocol employed in the networks considered was designed to work in wireless multi-hop networks possessing bi-directional links only and thus does not handle uni-directional links very well. AODV follows that RREQ-RREP-DATA sequence when routing data through the network. The RREQ-RREP process assumes that all the links in the network are bi-directional because the RREPs are transmitted in the reverse path that is setup during the propagation of RREQs that aim to find a route to the intended destination. If the path used to reach an intended destination contains one or more uni-directional links, the ability of the RREP control packets to reach the source node is affected. Thus the route for data transfer cannot be established and the end result is that the PDR of the network is adversely affected, as seen in Fig. 4.8.

4.5.2.2 Experiment 2: Routing Overhead

The purpose of this experiment was to determine the amount of routing protocol overhead that was created to establish the transmission routes between sources and

Table -4-4 – Routing overhead before and after TbTC

	Without TbTC	With TbTC
Run 3	270	1264
Run 9	305	1607
Run 15	1473	1022
Average	**683**	**1298**

destinations. The routing overhead was measured as the to'al number of unique AODV control messages that were sent during the simulation run.

Results

Table 4-4 shows the routing overhead created by AODV before and after the TbTC algorithm can be applied to the network. The data is depicted graphically in Fig. 4.9.

Fig. 4.9 shows an approximate doubling of the routing overhead generated by AODV after the TbTC algorithm has been applied to the network. This increase in routing overhead can be attributed to the uni-directional links that are inadvertently created during the distributed execution of the TbTC algorithm.

The AODV routing algorithm was created on the assumption that all the links in the network are bi-directional and by extension that all the possible paths in the network are bi-directional. The uni-directional links created by TbTC affect the operation of AODV. AODV is designed to follow the least congested path to the destination and

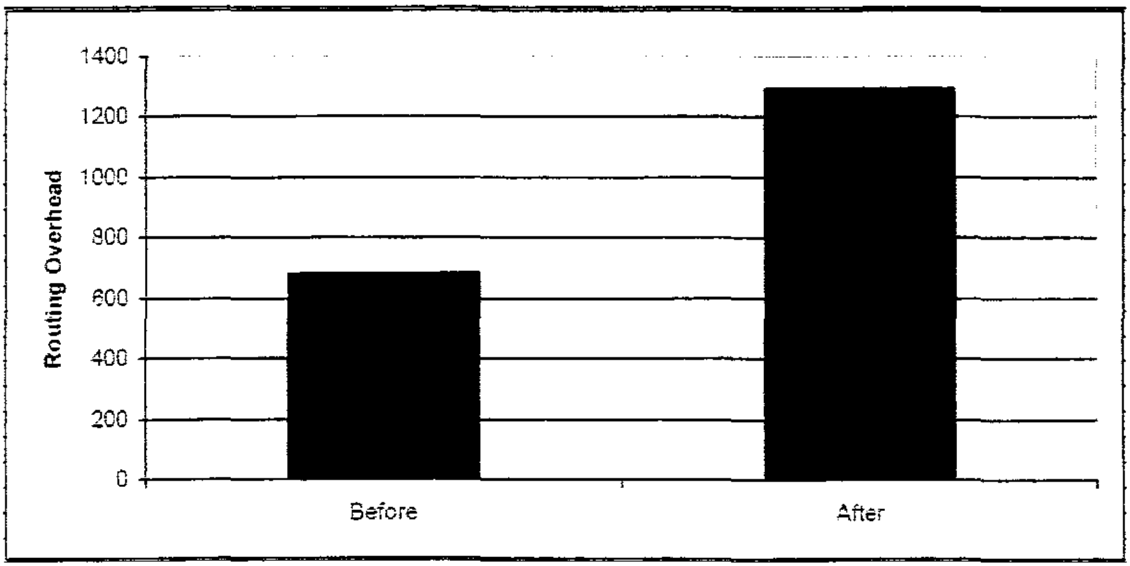


Fig. 4.9 – Routing Overhead before and after TbTC

assumes that the paths taken by RREQ messages are then reversed for the transmission of RREP messages[10].

Uni-directional links may allow the transmission of RREQ messages in the forward direction but will not allow the transmission of RREP messages along the reverse path. This situation triggers two further RREQ retransmissions by AODV before the issuing of a network-wide RREQ broadcast, thereby significantly increasing the routing overhead generated.

An additional cause for the increase in routing overhead is the increase in the possibility for buffer overflows that arises due to the reduction in route redundancy that is caused by the TbTC's creation of uni-directional links. A decrease in route redundancy creates more bottleneck nodes at which the possibility for buffer overflows is increased, thereby resulting in the retransmission of AODV control messages thus contributing to the increase in routing overhead.

[10] Enforcing AODV's need for bi-directional paths between sources and destinations

Table -4-5 – End-to-End Delay before and after TbTC

	Without TbTC	With TbTC
Run 3	2.74	3.02
Run 9	3.66	2.12
Run 15	3.17	3.25
Average	**3.19**	**2.80**

4.5.2.3 Experiment 3: End-to-End Delay

The purpose of this experiment was to determine the average time taken to deliver the Application Layer packets from the source node to the intended destination node. Note that the results are reported in milliseconds.

Results

Table 4-5 shows the average end-to-end delay experienced in the transmission of Application Layer packets in the forward direction, from the source to the destination. This data is plotted to produce Fig. 4.10.

Fig. 4.10 shows that TbTC reduces the delay experienced by Application Layer packets travelling from their sources to destinations (once the AODV routing protocol has established a transmission path), despite the average path length remaining unchanged (see Fig. 4.11). The reduction in delay can be attributed to the decrease in interference brought about as a result of the lessening of node transmission powers. Reduced transmission powers result in the decrease of the node's interference range and therefore the number of collisions that occur due to interference.

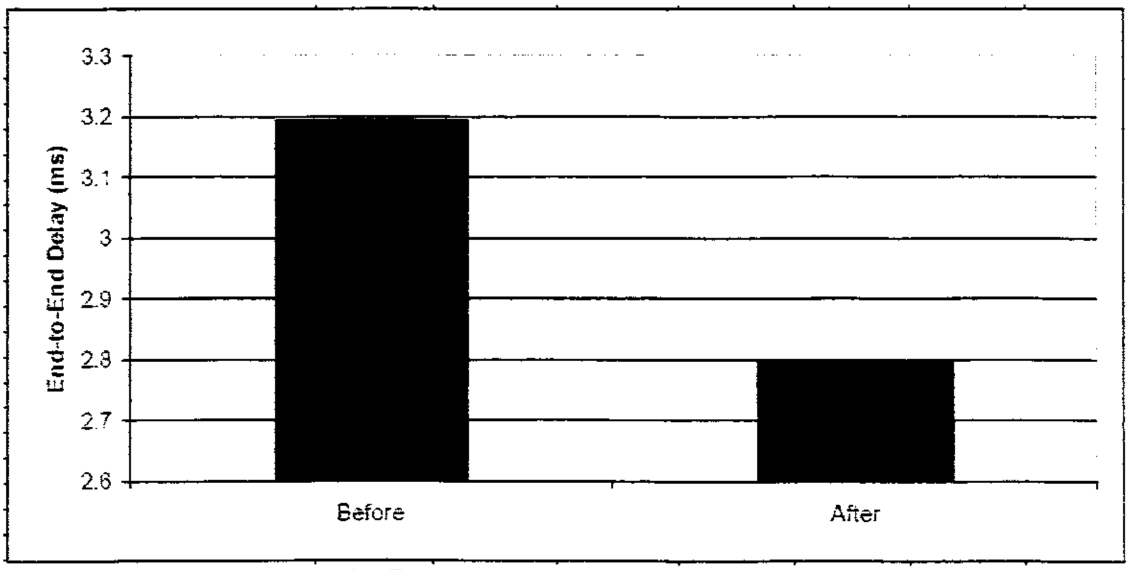


Fig. 4.10 – End-to-End Delay before and after TbTC

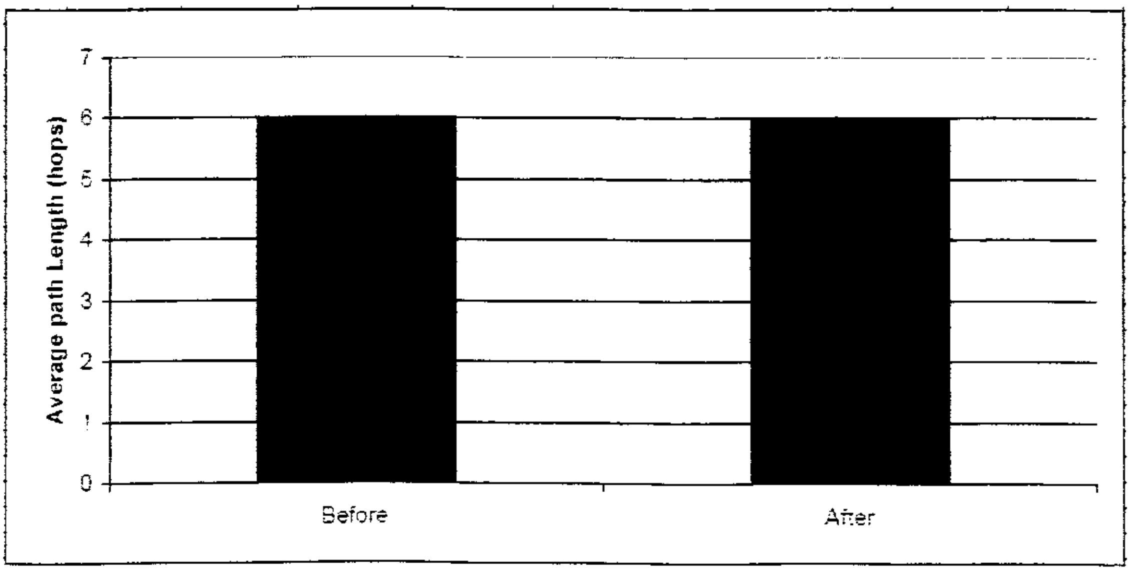


Fig. 4.11 – Average Path Length before and after TbTC

The reduction of the node's transmission powers also affects the end-to-end delay experienced by Application Layer packets due to the reduction in the average node degree of the network, as depicted in Fig. 4.12. A smaller average node degree results in less contention for the transmission medium thereby allowing packets to be sent faster at each intermediate node in the path, rendering the reduction in average path length not necessary as depicted in Fig. 4.11.

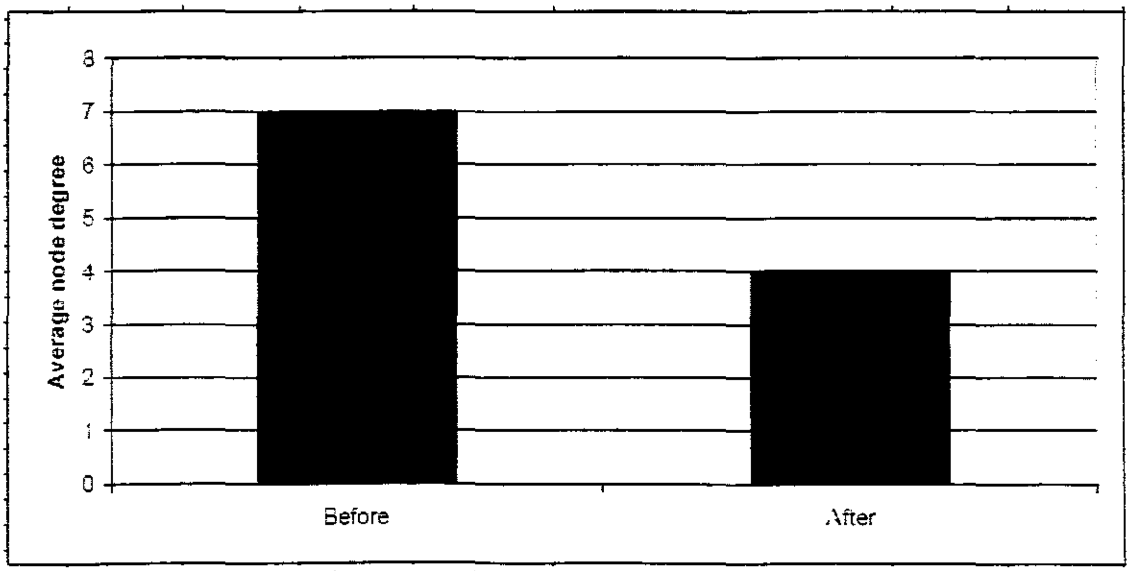


Fig. 4.12 – Average Node degree before and after TbTC

4.5.2.4 Experiment 4: Power Consumption

The purpose of this experiment was to determine the impact that the TbTC protocol had on the cumulative power consumed by the nodes in the network when performing both broadcasts and unicasts. The results reported ignore the energy consumed in the processing of signals and are reported in watts.

Results

Table 4-6 shows the cumulative power consumed before and after the TbTC algorithm was applied to a 30-node network. This data is plotted to produce Fig. 4.13.

Fig. 4.13 shows that TbTC produces a 42% reduction in the power consumed by the network. This reduction is a result of the heterogeneity in the transmission powers selected by the nodes in the network via the use of the TbTC algorithm. Each node adjusts itself to a transmission power sufficient to maintain 4 neighbours (wherever possible).

Table 4-6 – Power Consumption before and after TbTC

	Before TbTC	After TbTC
Power Consumption (w)	1.89287202	1.10378207

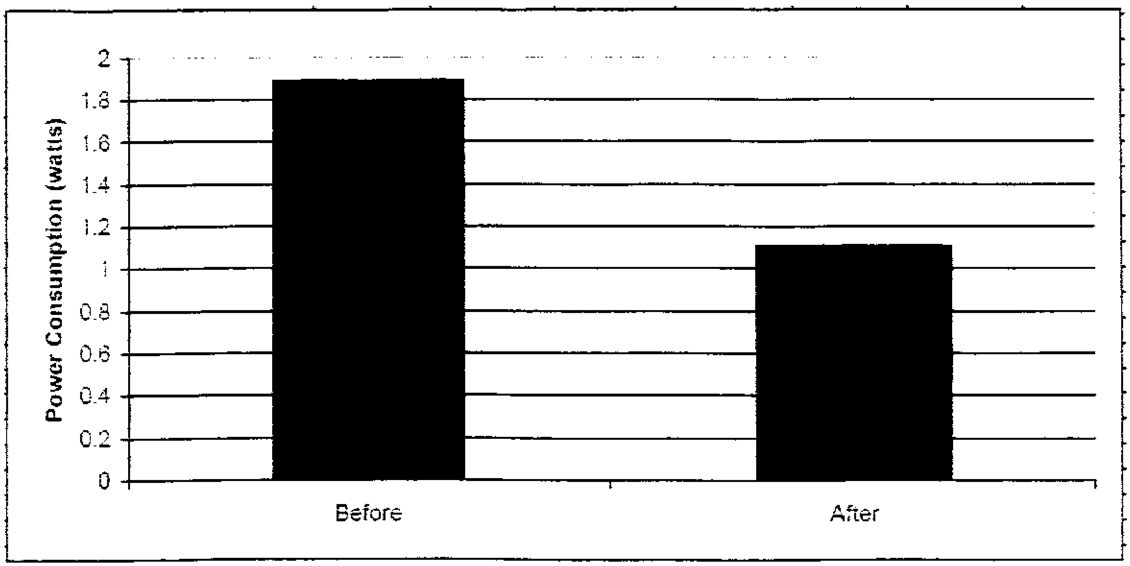


Fig. 4.13 – Power Consumption before and after TbTC

The power savings that are achieved by the TbTC algorithm reduce the total cost of ownership of the nodes involved, since the owners of these wireless multi-hop network nodes must eventually pay for the electricity consumed by the nodes. In situations where the nodes are battery-powered, the lower power consumption achieved by the TbTC algorithm lengthens the uptime of the node.

4.5.2.5 Experiment 5: Network Traversal (Hop Count)

The purpose of this experiment was to evaluate the effectiveness of the *neighbour control* process (described in Section 4.3.3) embedded within the *next node selection* component. The results reported reflect the number of hops taken before all the nodes in the network are visited by a token at least once. The hop count is an indirect measure of the time taken before all the nodes in the network apply their instances of the TbTC algorithm.

Table 4-7 – Hop Count of a 30 node network

	Before TbTC		After TbTC	
	Without Neighbour Control	With Neighbour Control	Without Neighbour Control	With Neighbour Control
Hop Count	56	41	43	42

Results

Table 4-7 shows the number of hops taken by the token to visit all the nodes in the network at least once. The hop counts of four different scenarios are tabulated to produce Fig. 4.14.

Fig. 4.14 highlights the effectiveness of the *neighbour control* process embedded within the *next node selection* component of TbTC. The *neighbour control* process reduces the number of hops taken by the token such that all the nodes in the network have been visited at least once. This effect is more profoundly felt in networks that have not had the TbTC algorithm applied to them, resulting in a 37.5% decrease in the hop count.

The significant decrease in hop count in pre-TbTC networks is accounted for by the intelligent token forwarding mechanism called *neighbour control*. Neighbour control only allows the token to be returned to the previous recipient of the token if the current recipient of the token has only one neighbour. This restriction avoids situations where the token oscillates between two nodes, thus increasing the number of hops required by the token to visit every network node at least once.

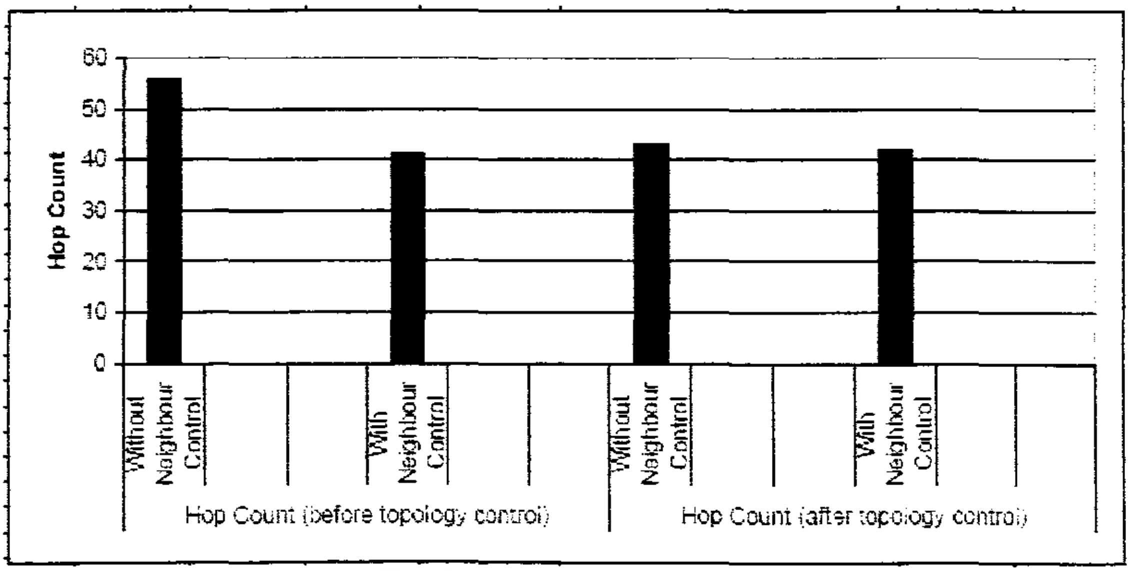


Fig. 4.14 – Hop Count of the token employed in TbTC

Token oscillation occurs when the token count of a particular node lags significantly behind the token counts of the other nodes in its local neighbourhood. This situation results in token oscillating between the two network nodes with the lowest token counts in that particular local neighbourhood until their respective token counts reach the levels of the other nodes in the neighbourhood. The elimination of token oscillation (except for those cases where the current recipient of the node has only one neighbour) accounts for the observed difference in hop count.

The marginal decrease in hop count observed in post-TbTC networks is as yet unaccounted for but it is suspected that the reduction in the average node degree of the network reduces the possibility of token oscillation by ensuring that no network node's token count lags significantly behind those of the nodes in its local neighbourhood.

4.5.3 Limitations of the Simulator and the Experiment

Simulation experiments are at best an approximation of the real world. Thus there are bound to be assumptions made in an effort to model the environment being considered. This section highlights the assumptions made, any limitations on the experiments conducted as well as any inherent limitations of the simulation tool that was utilised. It is possible that one or more of the assumptions made and the limitations of the experiments and simulation tool could have affected the results presented.

The assumptions and limitations are:

i. The number of nodes in the network is known in advance in order to determine the optimal number of neighbours. If not, then the algorithm developed by (Wan and Yi, 2002) for determining the optimal number of neighbours as a function of the network size can be used;

ii. The token is regenerated at every recipient node, but never lost;

iii. Lack of realistic Application Layer modelling

A constant bit rate model was utilised whereas realistic Application Layer traffic resembles a variable bit rate traffic stream;

iv. The terrain was assumed to be flat with no obstacles

Realistic terrain models consider the elevation of the nodes as well objects such as trees, etc;

v. The nodes in the network were stationary, and

vi. The IEEE 802.11 RTS/CTS mechanism was disabled.

4.6 Token-based Topology Control Algorithm Limitations

Here we consider the Token-based Topology Control Algorithm's limitations that affect the performance as well as the effectiveness of the algorithm. Listed below are the limitations discovered during the algorithm's development and simulation.

i. The probability, *p*, of a node receiving a token decreases as the size of the network increases.

$p = \frac{1}{N}$ where N is the number of nodes in the network

By extension, the interval between two visits of a token to the same node increases as the size of the network increases. This situation means that TbTC does not scale well when utilised in non-hierarchical wireless multi-hop networks.

Two approaches can be taken to ensure greater scalability: either the number of tokens circulating around the non-hierarchical network increases as the network grows, or, some clustering algorithm is utilised to create a hierarchical network based on the creation of clusters. Several clustering algorithms have been developed to ensure effective network management and examples include (Kleinrock and Kamoun, 1980), (Miyao, et al, 1986), (Ramanathan and Steenstrup, 1998), (Krishnan, et al, 1999). The TbTC algorithm can then be applied to each cluster independently, increasing the probability of receiving a token to $p = \frac{1}{N}$ where N is the number of nodes in the cluster. At any point in time the number of tokens in the network will equal the number of clusters.

ii. TbTC currently has no mechanisms for determining the number of nodes in the network and therefore cannot determine the optimal number of neighbours if the size of the network is unequal to 30 nodes.

iii. The optimal transmission power is reduced after evaluating the SNRs of replies from neighbouring nodes to a broadcast Hello message from node$_a$. Obstacles in the form of trees and buildings may affect the SNR values detected at node$_a$, thereby resulting in the selection of a sub-optimal transmission power.

iv. The Token-based Topology Control algorithm is currently not suitable for mobile wireless multi-hop networks because the nodes can only perform the algorithm when in possession of the token. Mobile nodes should be able to determine their optimum transmission powers either whilst in motion or shortly after coming to a halt (either temporary or permanent) and the use of the token-based algorithm would not react to changes in the node's location. A change in the node's location could potentially prevent a node from ever receiving the token because the node may be moving away from the region of the network in which the token is currently circulating.

v. The Token-based Topology Control Algorithm may cause either full or partial network partitioning (due to uni-directional links) in certain scenarios. Fig. 4.15 shows a network topology before the algorithm can be applied to it and Fig. 4.16 shows the same network subsequent to the application of the algorithm. It can be clearly seen that the bi-directional link between nodes A and B becomes a uni-directional one. This situation occurs when node B receives the token and adjusts its transmission power

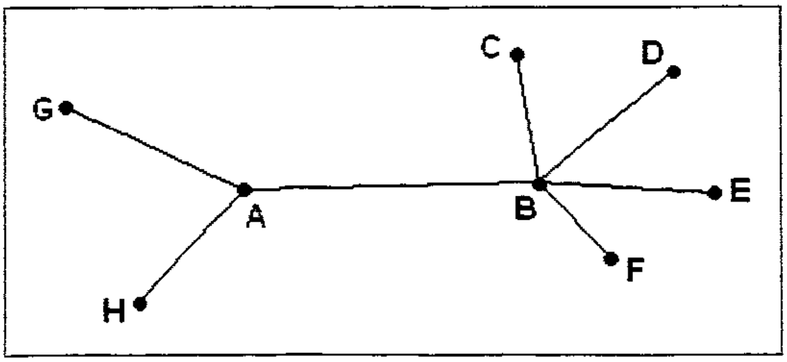


Fig. 4.15 – Network topology prior to network partitioning

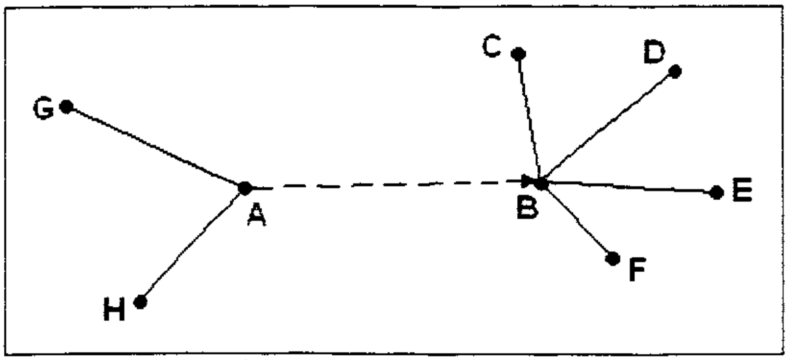


Fig. 4.16 – Network topology after partial network partitioning

to reach only nodes C, D, E and F. During this process B would have contacted A to determine whether B is A's only neighbour. A would have responded negatively allowing B to adjust its transmission power. Node A in the meanwhile has three neighbours within its transmission range and therefore will not adjust its transmission power upon receipt of the token, resulting in the uni-directional link between A and B.

The situation described above (and shown in Figs. 4.15 and 4.16) results in partial network partitioning because data can still be transferred from G to E but not vice-versa. The token-based topology algorithm could also result in the worst case scenario where no link (either uni-directional or bi-directional) exists between A and B if A had four or more neighbours within closer proximity than B.

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

5.1 Conclusion

This work presents an analysis of work done in determining the optimal number of neighbours for wireless multi-hop networks as well as an analysis of prior work in the field of topology control. A simulation was conducted to determine the optimal number of neighbours and together with the two purpose-developed theoretical frameworks for both the optimal number of neighbours and topology control, yielded the design criteria for a topology control algorithm suitable for low cost, resource-constrained wireless multi-hop network nodes.

The Token-based Topology Control (TbTC) algorithm was proposed, taking into account the above-mentioned design criteria as well as the results obtained from our experiment conducted (in Chapter Three) into the optimal number of neighbours (node degree) in a wireless multi-hop network. An evaluation of TbTC's performance was subsequently undertaken by comparing its performance against that achieved by a network not regulated by a topology control algorithm.

The TbTC algorithm comprised of three components: 1) *transmission power and adjustment*, 2) *network connectivity* and 3) *next node selection component*. A key feature of the algorithm is the use of a token to control its execution. This execution restriction imposed by the token ensured that only one node in the local neighbourhood (either within a cluster or the set of nodes within transmission range) executed the algorithm. The restriction ensures that none of the neighbouring nodes are executing their instances of the topology control algorithm, thereby reducing their abilities to respond to the request for information issued by the node in possession of the token. The use of the token also ensures that the node possessing the token obtains the most complete picture of its local neighbourhood resulting in the selection of a close-to-optimal transmission power.

To achieve the goal of this research work the following objectives were set: First, to identify WMNs with varied average node degrees under both the "perfect" and random critical node disconnection scenarios. The successful completion of these two objectives resulted in the optimal number of permitted neighbouring nodes which was then employed in the TbTC algorithm crafted in fulfilment of the third objective of this research.

The successful simulation of TbTC in order to evaluate its performance culminated in the achievement of the fourth objective of the research. This performance was compared to the performance of a network wherein TbTC was not employed. The simulation results showed that TbTC improved upon the Delay experienced and the power consumption of the network but performed badly with regards to Packet Delivery Ratio and Routing Overhead. TbTC's neighbourControl process was also shown to significantly reduce the number of hops necessary for all the network nodes to obtain the token and execute their instances of the algorithm.

5.2 Future Work

The disappointing performance of TbTC with regards to the Packet Delivery Ratio and Routing Overhead was mainly attributed to the creation of uni-directional links as a result of the heterogeneous transmission power assignment. Future work will attempt to eliminate the creation of uni-directional links as well as investigate the relationship between the total number of nodes in the (peer-to-peer) network and the optimal number of tokens in circulation.

The real-world implementation of the TbTC protocol on the test-bed being constructed at the University of Zululand is also envisaged. The incorporation of the TbTC routing protocol with an existing clustering algorithm is also recommended in order to allow the TbTC algorithm to be utilised in large-scale sensor network deployments.

Future work will explore the integration of TbTC with an existing Medium Access Control scheme that is able to leverage the token's ability to restrict data transmission, thereby reducing the total interference in the network. Also envisaged is the transfer of next node selection logic (currently residing in the network nodes) to reside within the token. Thus the token is viewed as a mobile agent that is able to determine the next node that it should be sent to. This approach has the advantage of reducing the processing performed by the node.

BIBLIOGRAPHY

Akella, A., Judd, G., Seshan, S., Steenkiste, P., 2005. Self Management in Chaotic Wireless Deployments. *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking, Cologne, Germany, 28 August 2005.* New York, USA: ACM Press, pp 185 – 199.

Allen, W., Martin, A., Rangarajan, A., 2005. Designing and Deploying a rural Ad-Hoc Community Mesh Network Testbed. *Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary,* 15 – 17 November 2005.

Bahramgiri, M., Hajiaghayi, M., Mirrokni, V. S., 2002. Fault-Tolerant and 3-Dimensional Distributed Topology Control Algorithms in Wireless Multi-Hop Networks. *Proceedings of the Eleventh International Conference on Computer Communications and Networks*, 14 October 2002. pp 392 – 397.

Bhagwat, P., Raman, B., Sanghi, D., 2004. Turning 802.11 Inside-Out. *ACM SIGCOMM Computer Communications Review*, 34 (1), 33 – 38.

Cardell-Oliver, R. 2003. *Why Flooding is Unreliable in Multi-hop, Wireless Networks.*. Unpublished.

Ferrari, G., Tonguz, O. K. 2004. Minimum Number of Neighbors for Fully Connected Uniform Ad Hoc Wireless Networks. *Proceedings of the IEEE Conference on Communications,* 20 June 2004. pp 4331 – 4335.

Gerharz, M., de Waal, C., Martini, P., James, P., 2003. A Cooperative Nearest Neighbours Topology Control Algorithm for Wireless Ad Hoc Networks. *Proceedings of the 12th International Conference on Computer Communications and Networks.* 20 October 2003. pp 412 – 417.

Hajek, B. 1983. Adaptive Transmission Strategies and Routing in Mobile Radio Networks. *Proceedings of the Seventeenth Annual Conference on Information Sciences and Systems.* 23 March 1983. pp 373 – 378.

Hou, T., Li, V.O.K., 1986. Transmission Range control in Multihop Packet Radio Networks. *IEEE Transactions on Communications,* 34 (1), 38 – 44.

Hu, L., 1993. Topology Control for Multihop Packet Radio Networks. *IEEE Transactions on Communications*, 41 (10), pp 1474 – 1481.

Jain, K., Padhye, J., Padhmanaban, V.N., Qiu, L., 2005. Impact of Interference on Multi-Hop Wireless Network Performance. *Wireless Networks*, 11 (4), 471 – 487.

Jangeun, J., and Sichitiu, M.L., 2003. The Nominal Capacity of Wireless Mesh Networks. *IEEE Wireless Communications*, 10 (5). 8 – 14.

Kleinrock, L., and Kamoun, F. 1980. Optimal Clustering Structures for Hierarchical Topological Design of Large Computer Networks. *Networks*, 10 (3), pp 221 – 248.

Kleinrock, L., and Silvester, J. 1978. Optimum Transmission Radii for Packet Radio Networks or why Six is a Magic Number. *National Telecommunications Conference*, December 1978. pp 4.3.1 – 4.3.5.

Krishnan, R., Ramanathan, R., Steenstrup, M. 1999. Optimization Algorithms for Large Self-Structuring Networks. *Proceedings of IEEE INFOCOM.* March 1999, pp 71 – 78.

Kurkowski, S., Camp, T., Mushell, N., Colagrosso, M. 2005. A Visualization and Analysis Tool for NS-2 Wireless Simulations: iNSpect, *Proceedings of the 13th IEEE International Symposium on Modeling Analysis, and Simulation of Computer and Telecommunication Systems*, 27 September 2005, pp. 503

- 506.

Lancaster Mesh 2006. *Main Page – LancasterMesh* [online]. Available from http://lancastermesh.co.uk/wiki/index.php/Main_Page [accessed on 28 May 2007].

Lee, S., Yu, Y., Nelakudithi, S., Zhang, Z.L., Chuah, C.N., 2004. Proactive vs Reactive Approaches to Failure Resilient Routing. *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, 7 March 2004.

Li, L., Halpern, J. Y., Bahl, P. Wang, Y. M., Wattenhofer, R. 2001. Analysis of a Cone-Based Topology Control Algorithm for Wireless Multi-Hop Networks. *Proceedings of the Twentieth Annual ACM Symposium on Principles of Distributed Computing*. pp 264 – 273.

Li, L., Halpern, J. Y., Bahl, P. Wang, Y. M., Wattenhofer, R. 2005. A Cone-Based Distributed Topology Control Algorithm for Wireless Multi-Hop Networks. *IEEE Transactions on Networking*, 13 (1), pp 147 – 159.

Li, N., Hou, J., Sha L. 2005. Design and Analysis of an MST-based Topology Control Algorithm. *IEEE Transactions on Wireless Communications*, 4 (3), pp 1195 – 1206.

Li, N., Hou, J.C., 2005. Localized Topology Control Algorithms for Heterogeneous Wireless Networks. *IEEE/ACM Transactions on Networking*, 13 (6), 1313 – 1324.

Li, N., Hou, J.C., 2006. Localized Fault-Tolerant Topology Control in Wireless Ad Hoc Networks. *IEEE Transactions on Parallel and Distributed Systems*, 17 (4), 307 – 320.

Linksys Inc 2007. *Linksys.com – Products/Wireless/Basic Networking/Broadband*

Routers/Wireless-G/WRT54G [online]. Available from www.linksys.com [accessed on 28 May 2007].

Marco Fiore 2004. *Marco Fiore* [online]. Available from http://www.tlc-networks.polito.it/fiore/ [accessed on 24 August 2006].

Meraka Institute 2005. *Mpumalanga Mesh - Wireless Africa* [online]. Meraka Institute. Available from: http://wirelessafrica.meraka.org.za/wiki/index.php/Mpumalanga_Mesh [Accessed on 22 November 2006].

Miyao, J., Ishida, K., Kikuno, T., Yoshida, N. 1986. Network Clustering Algorithm in Large Computer Networks. *Proceedings of the Nineteenth Hawaii International Conference on System Sciences*. 8 January 1986, pp 321 – 329.

Motorola Inc, 2005. *Mesh Networks*. Motorola Inc.

Mudali, P., Nyandeni, T.C., Adigun, M.O., 2007. A Performance Comparison of Wireless Multi-Hop Network Topologies Based on Average Node Degree. *To appear in South African Telecommunications and Network Applications Conference*, 19 September 2007.

Naghian, S. 2004. Mesh vs. Point-to-Multipoint Topology: A Coverage and Spectrum Efficiency Comparison. *15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 5 September 2004. pp 1048 – 1051.

Park, S., Sivakumar, R., 2002. Load-Sensitive Transmission Power Control in Wireless Ad-hoc Networks. *IEEE Global Telecommunications Conference*. 17 November 2002. pp 42 – 46.

Perkins, C., Belding-Royer, E., Das, S. 2003. *Ad hoc On-Demand Distance Vector*

(AODV) Routing [online]. The Internet Society. Available from: http://www.ietf.org/rfc/rfc3561.txt [Accessed 27 May 2006].

Philips, T., Panwar, S., Tantawi, A., 1989. Connectivity Properties of a Packet Radio Network Model. *IEEE Transactions on Information Theory*, 35 (5), pp 770 – 777.

Ramanathan, R., Rosales-Hain, R. 2000. Topology Control of Multihop Wireless Networks using Transmit Power Adjustment. *Proceedings of the Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, 26 March 2000. pp 404 – 413.

Ramanathan, R., Steenstrup, M. 1998. Hierarchically-Organized, Multihop Mobile Wireless Networks for Quality-of-Service Support. *ACM/Baltzer Mobile Networks and Applications Journal*. 3 (1), pp 101 – 119.

Saivichit, C., Wattanaparadorn, P. 2004. Criticality Analysis of Communication Networks in Anomaly Situations. *Proceedings of the IEEE Region 10 Conference*, 21 Nov 2000. pp 192 – 195.

Santi, P., 2005. Topology Control in Wireless Ad Hoc and Sensor Networks. *ACM Computing Surveys*, 37 (2), 164 – 194.

Srivastava, G., Boustead, P., Chicharo, J.F. 2004. Topology Control in Heterogeneous Ad-hoc Networks. 16 November 2004. pp 665 – 670.

Stallings, W. 2005. *Wireless Communications and Networks*. New Jersey: Pearson Prentice Hall.

Takagi, H., and Kleinrock, L. 1984. Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals. *IEEE Transactions on Communications*, 32 (2), pp 246 – 257.

The Network Simulator – ns-2 [online]. Available at http://www.isi.edu/nsnam/ns/

[accessed on 12 May 2006].

Tibetan Technology Center 2005. *Dharamsala Wireless Mesh – Forum| Tibetan technology Center* [online]. Available from http://www.tibtec.org/?q=taxonomy/term/3 [Accessed on 28 May 2007].

Tomsho, G., Tittel, E., Johnson, D. 2002. *Guide to Networking Essentials*. New Jersey: Course Technology.

Tracegraph 2007. *Network simulator NS-2 trace files analyzer – Tracegraph* [online]. Available from http://www.tracegraph.com [accessed on 28 May 2007].

Wan, P., and Yi, C., 2004. Asymptotic Critical Transmission Radius and Critical Neighbor Number for k-Connectivity in Wireless Ad Hoc Networks. *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 24 – 26 March 2004. pp 1 – 8.

Wattenhofer, R., Zollinger, A., 2004. XTC: A Practical Topology Control Algorithm for Ad-Hoc Networks. *Proceedings of the 18th International Parallel and Distributed Processing Symposium*, 26 – 30 April 2004. pp 216 – 224.

Wu, C., and Hou, T. 2005. The Impact of RTS/CTS on Performance of Wireless Multihop Ad Hoc Networks Using IEEE 802.11 Protocol. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 10 October 2005, pp 3558 – 3562.

Wu, K., and Liao, W. 2006. Interference-Efficient Topology Control in Wireless Ad Hoc Networks. *Proceedings of the 3rd IEEE Consumer Communications and Networking Conference*, 8 January 2006. pp 411 – 415.

Xu, K., Gerla, M., Bae, S. 2002. How Effective is the IEEE 802.11 RTS/CTS

Handshake in Ad Hoc Networks? *Proceedings of the Global Telecommunications Conference*, 17 November 2002, pp 72 – 76.

Xue, F., and Kumar, P., 2004. The Number of Neighbors Needed for Connectivity of Wireless Networks. *Wireless Networks*, 10 (2), pp 169 – 181.

Xue, F., and Kumar, P., 2006. On the θ-Coverage and Connectivity of Large Random Networks. *IEEE Transactions on Information Theory*, 52 (6), pp 289 – 299.

APPENDIX A – NS-2 Simulation Script for "Perfect" Scenario

```
#=======================================================
#DEFINE OPTIONS
#=======================================================
set val(chan) Channel/WirelessChannel ;- channel type
set val(prop) Propagation/TwoRayGround ;= radio-propagation model
set val(netif) Phy/WirelessPhy ;= network interface type
set val(mac) Mac/802_11 ;= MAC type
set val(ifq) Queue/DropTail/PriQueue ;= interface queue type
set val(ll) LL ;- link layer type
set val(ant) Antenna/OmniAntenna ;= antenna model
set val(ifqlen) 50 ;= max packet in ifq
set val(nn) 30 ;= number of wireless nodes
set val(x) 2000
set val(y) 600
set val(rp) AODV ;= routing protocol
set val(scen) "scen-1000x600-30-901-1-1" ;= scenario file
set val(cp) "cbr-30-29-1-64" ;= connection pattern file
set val(sim_duration) 900 ;= duration of the simulation run
set val(addr_type) flat ;- addressing type

LL set mindelay_ 50us ;
LL set delay_ 25us ;

Agent/Null set sport_ 0 ;
Agent/Null set dport_ 0 ;

Agent/CBR set sport_ 0 ;
Agent/CBR set dport_ 0 ;

Agent/UDP set sport_ 0 ;
Agent/UDP set dport_ 0 ;
Agent/UDP set packetSize 64 ; - 64 bytes

Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1 ;

#=======================================================
#ADDITIONAL OPTIONS based upon Linksys WRT54G specs
#=======================================================
# unity gain omnidirectional antennas  centered  n the node and 1.5m above it
Antenna/OmniAntenna set X_ 0 ;
Antenna/OmniAntenna set Y_ 0 ;
Antenna/OmniAntenna set Z_ 1.5 ;
Antenna/OmniAntenna set Gt_ 4.0 ;- transmit antenna gain
Antenna/OmniAntenna set Gr_ 4.0 ;= receive antenna gain
#==========================================================================
#DSSS (IEEE 802.11b)
Mac/802_11 set CWMin_ 31
Mac/802_11 set CWMax_ 1023
Mac/802_11 set RTSThreshold_ 3000 ;= bytes
Mac/802_11 set SlotTime_ 0.000020 ;=20us
Mac/802_11 set SIFS_ 0.000010 ;=10us
Mac/802_11 set PreambleLength_ 144 ;=72 bits
Mac/802_11 set PLCPHeaderLength_ 48 ;-48 bits
Mac/802_11 set PLCPDataRate_ 1.0e6
Mac/802_11 set dataRate_ 11Mb ;= rate for data frames
Mac/802_11 set basicRate_ 2Mb ;= rate for control frames
Mac/802_11 set aarf_ true ;= adaptive auto rate fallback
#==========================================================================
Phy/WirelessPhy set L_ 1.0 ;= system loss factor
Phy/WirelessPhy set freq_ 2.462e9 ;= channel 11. 2.462GHz
Phy/WirelessPhy set bandwidth_ 11Mb ;= 11 Mbps channel bandwidth
Phy/WirelessPhy set Pt_ 0.063095734 ;= transmission power in watts
Phy/WirelessPhy set CPThresh_ 5.0 ; - collision threshold
Phy/WirelessPhy set CSThresh_ 1.30835e-09 ;= carrier sense power
Phy/WirelessPhy set RXThresh_ 1.30835e-09 ;= receive power threshold
#==========================================================================
ErrorModel80211 noise1_ -104
ErrorModel80211 noise2_ -101
ErrorModel80211 noise55_ -97
```

```
ErrorModel80211 noise11_ -92
ErrorModel80211 shortpreamble_ 1

#==========================================================
#MAIN PROGRAM
#==========================================================

#
#Initialise global variables
#
set ns_ [new Simulator]
set tracefd [open sim_trace.tr w]
#set namtrace [open sim_trace.nam w]

$ns_ use-newtrace
$ns_ trace-all $tracefd
#$ns_ namtrace-all-wireless $namtrace $val(x) $val(y)

#setup topography object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)

#create god
set god_ [create-god $val(nn)]

# create channel #1
set chan_1_ [new $val(chan)]

#configure node
$ns_ node-config -addressingType $val(addr_type)
-adhocRouting $val(rp)
-llType $val(ll)
-macType $val(mac)
-ifqType $val(ifq)
-ifqLen $val(ifqlen)
-antType $val(ant)
-propType $val(prop)
-phyType $val(netif)
-channel $chan_1_
-topoInstance $topo
-agentTrace ON
-routerTrace ON
-macTrace ON
-movementTrace OFF

#=============================================================
#$mobilenode addif
#$mobilenode radius
#=============================================================
#Node set multiPath_ 1

for {set i 0} {$i < $val(nn) } {incr i} {
set node_($i) [$ns_ node]
$node_($i) shape "box"
$node_($i) random-motion 0 ;# disable random motion
#set aodv($i) [new Agent/rtProto/AODV]
#$aodv($i) node $node_($i)
$god_ new_node $node_($i)
}


#
#Provide X,Y,Z coordinates for wireless nodes by loading the scenario file
#
puts "Loading the scenario file"
source $val(scen)


#Define node initial position in nam
for {set i 0} {$i < $val(nn)} {incr i} {

# 20 defines the node size in nam, must adjust it according to your scenario
# The function must be called after mobility model is defined
$ns_ initial_node_pos $node_($i) 20
}


#
#Setup traffic flow between nodes
#
puts "Loading connection pattern"
source $val(cp)


#
#Tell nodes when the simulation ends
#
for {set i 0} {$i < $val(nn) } {incr i} {
$ns_ at $val(sim_duration).0 "$node_($i) reset";
}

$ns_ at $val(sim_duration).1 "stop"
#$ns_ at $val(sim_duration).01 "puts \"NS EXITING...\" ; $ns_ stop"


proc stop {} {
global ns_ tracefd
$ns_ flush-trace
close $tracefd
}

puts $tracefd "M 0.0 nn $val(nn) x $val(x) y $val(y) rp $val(rp)"
puts $tracefd "M 0.0 sc $val(scen) cp $val(cp)"
puts $tracefd "M 0.0 prop $val(prop) ant $val(ant)"

puts "Starting Simulation..."
$ns_ run
```

APPENDIX B – NS-2 Simulation Script for Disconnected Scenario

```
#======================================================
#DEFINE OPTIONS
#======================================================
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(nn) 30 ;# number of wireless nodes
set val(x) 2000
set val(y) 600
set val(rp) AODV ;# routing protocol
set val(scen) "scen-1000x600-30-901-1-1" ;# scenario file
set val(cp) "cbr-30-29-1-64" ;# connection pattern file
set val(sim_duration) 900 ;# duration of the simulation run
set val(addr_type) flat ;# addressing type
LL set mindelay_ 50us ;
LL set delay_ 25us ;
#LL set bandwidth_ 0 ;# not used
Agent/Null set sport_ 0 ;
Agent/Null set dport_ 0 ;
Agent/CBR set sport_ 0 ;
Agent/CBR set dport_ 0 ;
Agent/UDP set sport_ 0 ;
Agent/UDP set dport_ 0 ;
Agent/UDP set packetSize 64 ;# 64 bytes
Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1 ;
#======================================================
#ADDITIONAL OPTIONS based upon Linksys WRT54G specs
#======================================================
# unity gain onmidirectional antennas, centered in the node and 1.5m above it
Antenna/OmniAntenna set X_ 0 ;
Antenna/OmniAntenna set Y_ 0 ;
Antenna/OmniAntenna set Z_ 1.5 ;
Antenna/OmniAntenna set Gt_ 4.0 ;# transmit antenna gain
Antenna/OmniAntenna set Gr_ 4.0 ;# receive antenna gain
##########################################################################
#DSSS (IEEE 802.11b)
Mac/802_11 set CWMin_ 31
Mac/802_11 set CWMax_ 1023
Mac/802_11 set RTSThreshold_ 3000 ;# bytes
Mac/802_11 set SlotTime_ 0.000020 ;#20us
Mac/802_11 set SIFS_ 0.000010 ;#10us
Mac/802_11 set PreambleLength_ 144 ;#72 bits
Mac/802_11 set PLCPHeaderLength_ 48 ;#48 bits
Mac/802_11 set PLCPDataRate_ 1.0e6
Mac/802_11 set dataRate_ 11Mb ;# rate for data frames
Mac/802_11 set basicRate_ 2Mb ;# rate for control frames
Mac/802_11 set aarf_ true ;# adaptive auto rate fallback
##########################################################################
Phy/WirelessPhy set L_ 1.0 ;# system loss factor
Phy/WirelessPhy set freq_ 2.462e9 ;# channel 11. 2.462GHz
Phy/WirelessPhy set bandwidth_ 11Mb ;# 11 Mbps channel bandwidth
Phy/WirelessPhy set Pt_ 0.063095734 ;# transmission power in watts
Phy/WirelessPhy set CPThresh_ 5.0 ;# collision threshold
Phy/WirelessPhy set CSThresh_ 1.30835e-09 ;# carrier sense power
Phy/WirelessPhy set RXThresh_ 1.30835e-09 ;#
##########################################################################
ErrorModel80211 noise1_ -104
ErrorModel80211 noise2_ -101
ErrorModel80211 noise55_ -97
ErrorModel80211 noise11_ -92
ErrorModel80211 shortpreamble_ 1
#======================================================
#MAIN PROGRAM
#======================================================
#
```

```
#Initialise global variables
#
set ns_ [new Simulator]
set tracefd [open sim_trace.tr w]
#set namtrace [open sim_trace.nam w]
$ns_ use-newtrace
$ns_ trace-all $tracefd
#setup topography object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)
#create god
set god_ [create-god $val(nn)]
# create channel #1
set chan_1_ [new $val(chan)]
#configure node
$ns_ node-config -addressingType $val(addr_type)
-adhocRouting $val(rp)
-llType $val(ll)
-macType $val(mac)
-ifqType $val(ifq)
-ifqLen $val(ifqlen)
-antType $val(ant)
-propType $val(prop)
-phyType $val(netif)
-channel $chan_1_
-topoInstance $topo
-agentTrace ON
-routerTrace ON
-macTrace ON
-movementTrace OFF
for {set i 0} {$i < $val(nn) } {incr i} {
set node_($i) [$ns_ node]
$node_($i) shape "box"
$node_($i) random-motion 0 ;# disable random motion
$god_ new_node $node_($i)
}
#
#Provide X.Y.Z coordinates for wireless nodes by loading the scenario file
#
puts "Loading the scenario file"
source $val(scen)
###################################################
#Node Disconnection for a duration of 180secs
###################################################
$ns_ at 390 $node_(29) setdest 2000.0 2000.0 900.0
$ns_ at 570 $node_(29) setdest 570.0 335.0 900.0
$ns_ at 150 $node_(7) setdest 2000.0 2000.0 900.0
$ns_ at 330 $node_(7) setdest 347.0 269.0 900.0
$ns_ at 300 $node_(25) setdest 2000.0 2000.0 900.0
$ns_ at 480 $node_(25) setdest 221.0 465.0 900.0
#Define node initial position in nam
for {set i 0} {$i < $val(nn)} {incr i} {
# 20 defines the node size in nam, must adjust it according to your scenario
# The function must be called after mobility model is defined
$ns_ initial_node_pos $node_($i) 20
}
#
#Setup traffic flow between nodes
#
puts "Loading connection pattern"
source $val(cp)
#
#Tell nodes when the simulation ends
#
for {set i 0} {$i < $val(nn) } {incr i} {
$ns_ at $val(sim_duration).0 "$node_($i) reset";
}
$ns_ at $val(sim_duration).1 "stop"
#$ns_ at $val(sim_duration).01 "puts \"NS EXITING...\" ; $ns_ stop"
proc stop {} {
global ns_ tracefd
$ns_ flush-trace
close $tracefd
}
puts $tracefd "M 0.0 nn $val(nn) x $val(x) y $val(y) rp $val(rp)"
puts $tracefd "M 0.0 sc $val(scen) cp $val(cp)"
puts $tracefd "M 0.0 prop $val(prop) ant $val(ant)"
puts "Starting Simulation..."
$ns_ run
```

APPENDIX C – NS-2 Simulation Script for TbTC

```
#========================================================
#DEFINE OPTIONS
#========================================================
set val(chan) Channel/WirelessChannel ;- channel type
set val(prop) Propagation/TwoRayGround ;= radio-propagation model
set val(netif) Phy/WirelessPhy ;= network interface type
set val(mac) Mac/802_11 ;= MAC type
set val(ifq) Queue/DropTail/PriQueue ;= interface queue type
set val(ll) LL ;= link layer type
set val(ant) Antenna/OmniAntenna ;= antenna model
set val(ifqlen) 100 ;= max packet in ifq
set val(nn) 30 ;= number of wireless nodes
set val(x) 1000
set val(y) 600
set val(rp) AODV ;= routing protocol
set val(scen) "scen-1000x600-30-901-1-1" ;= scenario file
set val(cp) "cbr-30-29-16-64" ;= connection pattern file
set val(sim_duration) 900 ;= duration of the simulation run
set val(addr_type) flat ;- addressing type

LL set mindelay_ 50us ;
LL set delay_ 25us ;
#LL set bandwidth_ 0 ;# not used

Agent/Null set sport_ 0 ;
Agent/Null set dport_ 0 ;

Agent/CBR set sport_ 0 ;
Agent/CBR set dport_ 0 ;

Agent/UDP set sport_ 0 ;
Agent/UDP set dport_ 0 ;
Agent/UDP set packetSize 64 ;= 64 bytes

Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1 ;

#========================================================
#ADDITIONAL OPTIONS based upon Linksys WRT54G specs
#========================================================
# unity gain omnidirectional antennas centered in the node and 1.5m above :
Antenna/OmniAntenna set X_ 0 ;
Antenna/OmniAntenna set Y_ 0 ;
Antenna/OmniAntenna set Z_ 1.5 ;
Antenna/OmniAntenna set Gt_ 4.0 ;- transmit antenna gain (to be finalised)
Antenna/OmniAntenna set Gr_ 4.0 ;= receive antenna gain (to be finalised)
##########################################################################
#DSSS (IEEE 802.11b)
Mac/802_11 set CWMin_ 31
Mac/802_11 set CWMax_ 1023
Mac/802_11 set RTSThreshold_ 3000 ;= bytes
Mac/802_11 set SlotTime_ 0.000020 ;=20us
Mac/802_11 set SIFS_ 0.000010 ;=10us
Mac/802_11 set PreambleLength_ 144 ;-72 bits
Mac/802_11 set PLCPHeaderLength_ 48 ;-48 bits
Mac/802_11 set PLCPDataRate_ 1.0e6
Mac/802_11 set dataRate_ 11Mb ;= rate for data frames
Mac/802_11 set basicRate_ 2Mb ;= rate for control frames
Mac/802_11 set aarf_ true ;= adaptive auto rate failback
##########################################################################
Phy/WirelessPhy set L_ 1.0 ;= system loss factor
Phy/WirelessPhy set freq_ 2.462e9 ;= channel 11. 2.462GHz
Phy/WirelessPhy set bandwidth_ 11Mb ;= 11 Mbps channel bandwidth
#Phy/WirelessPhy set Pt_ 0.063095734 ;# transmission power in watts
Phy/WirelessPhy set CPThresh_ 5.0 ;- collision threshold (to be finalised)
Phy/WirelessPhy set CSThresh_ 1.30835e-09 ;= carrier sense power
Phy/WirelessPhy set RXThresh_ 1.30835e-09 ;= receive power threshold
##########################################################################
ErrorModel80211 noise1_ -104
ErrorModel80211 noise2_ -101
ErrorModel80211 noise55_ -97
ErrorModel80211 noise11_ -92
```

```
ErrorModel80211 shortpreamble_ 1
#==========================================================
#MAIN PROGRAM
#==========================================================


#
#Initialise global variables
#
set ns_ [new Simulator]
set tracefd [open sim_trace.tr w]
set node_() [$ns_ node]
#set namtrace [open sim_trace.nam w]

$ns_ use-newtrace
$ns_ trace-all $tracefd

#setup topography object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)

#create god
set god_ [create-god $val(nn)]

# create channel #1
set chan_1_ [new $val(chan)]

#configure node
$ns_ node-config -addressingType $val(addr_type)
-adhocRouting $val(rp)
-llType $val(ll)
-macType $val(mac)
-ifqType $val(ifq)
-ifqLen $val(ifqlen)
-antType $val(ant)
-propType $val(prop)
-phyType $val(netif)
-channel $chan_1_
-topoInstance $topo
-agentTrace ON
-routerTrace ON
-macTrace ON
-movementTrace OFF


for {set i 0} {$i < $val(nn) } {incr i} {
set node_($i) [$ns_ node]
$node_($i) shape "box"
$node_($i) random-motion 0 ;# disable random motion
#set aodv($i) [new Agent/rtProto/AODV]
#$aodv($i) node $node_($i)
$god_ new_node $node_($i)
}

#==========================================================
# individual node tx power settings
#==========================================================
[lindex [$node_(0) array get netif_] 1] set Pt_ 0.043354865
[lindex [$node_(1) array get netif_] 1] set Pt_ 0.029813807
[lindex [$node_(2) array get netif_] 1] set Pt_ 0.028177187
[lindex [$node_(3) array get netif_] 1] set Pt_ 0.026945266
[lindex [$node_(4) array get netif_] 1] set Pt_ 0.026945266
[lindex [$node_(5) array get netif_] 1] set Pt_ 0.036582539
[lindex [$node_(6) array get netif_] 1] set Pt_ 0.047429827
[lindex [$node_(7) array get netif_] 1] set Pt_ 0.037336172
[lindex [$node_(8) array get netif_] 1] set Pt_ 0.028397436
[lindex [$node_(9) array get netif_] 1] set Pt_ 0.037336172
[lindex [$node_(10) array get netif_] 1] set Pt_ 0.028397436
[lindex [$node_(11) array get netif_] 1] set Pt_ 0.046893318
[lindex [$node_(12) array get netif_] 1] set Pt_ 0.043354865
[lindex [$node_(13) array get netif_] 1] set Pt_ 0.029813807
[lindex [$node_(14) array get netif_] 1] set Pt_ 0.026754342
[lindex [$node_(15) array get netif_] 1] set Pt_ 0.0247569
[lindex [$node_(16) array get netif_] 1] set Pt_ 0.02638433
[lindex [$node_(17) array get netif_] 1] set Pt_ 0.026945266
```

```
[lindex [$node_(18) array get netif_] 1] set Pt_ 0.028397436
[lindex [$node_(19) array get netif_] 1] set Pt_ 0.0279613
[lindex [$node_(20) array get netif_] 1] set Pt_ 0.058530798
[lindex [$node_(21) array get netif_] 1] set Pt_ 0.034136851
[lindex [$node_(22) array get netif_] 1] set Pt_ 0.063420234
[lindex [$node_(23) array get netif_] 1] set Pt_ 0.034136851
[lindex [$node_(24) array get netif_] 1] set Pt_ 0.028397436
[lindex [$node_(25) array get netif_] 1] set Pt_ 0.034136851
[lindex [$node_(26) array get netif_] 1] set Pt_ 0.026945266
[lindex [$node_(27) array get netif_] 1] set Pt_ 0.035498497
[lindex [$node_(28) array get netif_] 1] set Pt_ 0.052627392
[lindex [$node_(29) array get netif_] 1] set Pt_ 0.08397436
########################################################################

#
#Provide X,Y,Z coordinates for wireless nodes by loading the scenario file
#
puts "Loading the scenario file"
source $val(scen)


#Define node initial position in nam
for {set i 0} {$i < $val(nn)} {incr i} {

# 20 defines the node size in nam, must adjust it according to your scenario
# The function must be called after mobility model is defined
$ns_ initial_node_pos $node_($i) 20
}

#
#Setup traffic flow between nodes
#
puts "Loading connection pattern"
source $val(cp)

#
#Tell nodes when the simulation ends
#
for {set i 0} {$i < $val(nn) } {incr i} {
$ns_ at $val(sim_duration).0 "Snode_(Si) reset";
}

$ns_ at $val(sim_duration).1 "stop"
#Sns_ at Sval(sim_duration).01 "puts \"NS EXITING...\" ; Sns_ stop"


proc stop {} {
global ns_ tracefd
$ns_ flush-trace
close $tracefd
}

puts $tracefd "Trace file after implementing Token-based Topology Control"
puts $tracefd "M 0.0 nn Sval(nn) x Sval(x) y Sval(y) rp Sval(rp)"
puts $tracefd "M 0.0 sc Sval(scen) cp Sval(cp)"
puts $tracefd "M 0.0 prop Sval(prop) ant Sval(ant)"
puts "Starting Simulation..."
$ns_ run
```

APPENDIX D – Linksys WRT54G Specifications

CPU Speed: 200MHz

Flash Size: 4 Mb

RAM: 16Mb

RF Power Output: 18dBm max

IEEE 802.11b, IEEE 802.11g

13 Channels