2010-03-01

Improving Multicast Communications Over Wireless Mesh Networks

Brian Keegan
Technological University Dublin

Improving Multicast Communications over Wireless Mesh Networks

by

Brian Keegan
M.Phil, B.Eng.

A thesis submitted to the Dublin Institute of Technology
for the degree of

Doctor of Philosophy

Supervisor: Dr. Mark Davis

School of Electronic and Communications Engineering

March 2010

Abstract

In wireless mesh networks (WMNs) the traditional approach to shortest path tree based multicasting is to cater for the needs of the poorest performing node i.e. the maximum permitted multicast line rate is limited to the lowest line rate used by the individual *Child* nodes on a branch. In general, this means fixing the line rate to its minimum value and fixing the transmit power to its maximum permitted value. This simplistic approach of applying a single multicast rate for all nodes in the multicast group results in a sub-optimal trade-off between the mean network throughput and coverage area that does not allow for high bandwidth multimedia applications to be supported.

By relaxing this constraint and allowing multiple line rates to be used, the mean network throughput can be improved. This thesis presents two methods that aim to increase the mean network throughput through the use of multiple line rates by the forwarding nodes. This is achieved by identifying the *Child* nodes responsible for reducing the multicast group rate. The first method identifies specific locations for the placement of relay nodes which allows for higher multicast branch line rates to be used. The second method uses a power control algorithm to tune the transmit power to allow for higher multicast branch line rates. The use of power control also helps to reduce the interference caused to neighbouring nodes.

Through extensive computer simulation it can be shown that these two methods can lead to a four-fold gain in the mean network throughput under typical WMN operating conditions compared with the single line rate case.

Declaration

I certify that this thesis which I now submit for examination for the award of ____________________, is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

This thesis was prepared according to the regulations for postgraduate study by research of the Dublin Institute of Technology and has not been submitted in whole or in part for an award in any other Institute or University.

The work reported on in this thesis conforms to the principles and requirements of the Institute's guidelines for ethics in research.

The Institute has permission to keep, to lend or to copy this thesis in whole or in part, on condition that any such use of the material of the thesis be duly acknowledged.

Signature_________________________________Date______________

Acknowledgements

This thesis would not have been possible without the support of my fantastic wife Helen Sheridan, who has only ever known me as a student.

I would like to express my gratitude to my supervisor Dr. Mark Davis for giving me the opportunity to study at CNRI for so many years. All the staff at CNRI, past and present have influenced the work put into this thesis in some form or other. In particular I owe a huge debt of gratitude to my good friend Dr. Karol Kowalik for his guidance and enthusiasm to finish this project.

This thesis is the accumulation of many years of work and experience. I would like to give special thanks to Pat Bennett who first opened my eyes to the world of electronics when I was around 10 years of age by taking the back off a radio and exposing the treasures inside.

I would like to thank all of the staff in the Electrical Installations Department who were extremely encouraging and helpful during my Electrical Trades, in particular Mr. Gerry Farrelly. Likewise, in the School of Electronics and Communications Engineering I would like to express my thanks to all of the staff who had input in my full time education at DIT. In particular, Mr. Paul Tobin and Dr. Max Ammann. I would also like to thank Professor Gerald Farrell for his encouragement and for asking the difficult question, “So what?”. Special thanks goes to Professor Hugh Byrne at the Focas Institute.

I owe a debt of gratitude to the staff and management at MacArdle McSweeney Associates, in particular Mr. Damien Egan. I am extremely grateful to the helpful staff whom I had the pleasure of working with at Cisco, WNBU during my internship. In particular I would like to give special thanks to Mr. Ashish Kathapurkar and Mr. Mike Joyce.

My friends and family have given me overwhelming encouragement and support throughout the entire period of my full time education, some of whom were unfortunate to get the job of proof reading or worse, to listen to my rants about my progress. Jason O'Connell, Gemma Sheridan, Anne O'Hara, Eoin Kelly, Marek Bykowski, Stuart Wallace and Stephen Sheridan in particular were subjected to the most abuse in this matter.

Most of all I would like to thank my parents, Gerry and Sylvia Keegan, they are both simply fantastic. Everything I know and do is because of them.

On March 31st 2010 the most original creation I was ever involved with finally arrived. Sophia Keegan was delivered shortly before midnight and continues to amaze me every day.

List of Figures

List of Tables

Abbreviations and Acronyms.

ACK	Acknowledgement
AODV	Ad-hoc On-demand Distance Vector
AP	Access Point
ARF	Auto Rate Fallback
ARSM	Auto Rate Selection Multicast Mechanism
ATIM	Announcement Traffic Indication Message
Bagg	Aggregated Traffic Bandwidth
BCD	Bottleneck Collision Domain
BS	Base Station
BSS	Basic Service Set
CBT	Core-Based Tree
CC	Central Controller
CN	Contention Node
CTS	Clear To Send
DIFS	Distributed Inter-Frame Space
DS	Distribution System
DSR	Dynamic Source Routing
DVMRP	Distance Vector Multicast Routing Protocol
ESS	Extended Service Set
ETT	Expected Transmission Time
ETX	Expected Transmission Count
FGMP	Forwarding Group Multicast Protocol
FSL	Free Space Loss

GA	Genetic Algorithm
GP	Grid Position
GTNetS	The Georgia Tech Sensor Network Simulator
IBSS	Independent Basic Service Set
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
MAC	Medium Access Control
MANET	Mobile Ad Hoc Network
MAP	Mesh Access Point
MCT	Minimum Cost Tree
METX	Multicasting ETX
mETX	modified ETX
MILP	Mixed Integer Linear Programming
MinCont	Minimum Contention
MinDist	Minimum Distance
MinHop	Minimum Hop
MinPower	Minimum Power
MNT	Minimum Number of Transmissions
MOSPF	Multicast extension to Open Shortest Path First
MP	Mesh Points
MPLS	Multi-protocol Label Switching
MST	Minimum Steiner Trees
NAV	Network Allocation Vector
NEST	Network Simulator Testbed

ODMRP	On Demand Multicast Routing Protocol
OLSR	Optimised Link State Routing
OSPF	Open Shortest Path First
PDF	Probability Density Function
PER	Packet Error Rate
PHY	Physical Layer
PIM-DM	Protocol Independent Multicast – Dense Mode
PIM-SM	Protocol Independent Multicast – Sparse Mode
PktPair	Packet Pair Delay
PNC	Piconet Controller
PP	Packet Pair
QoS	Quality of Service
REAL	Real and Large
RF	Radio Frequency
RN	Relay Node
RTS	Request To Send
RTT	Round Trip Time
Rx	Receiver
SINR	Signal-to-Interference-plus-Noise Ratio
SNR	Signal to Noise Ratio
SPP	Success Probability Protocol
SPT	Shortest Path Tree
STA	Station
TCP	Transmission Control Protocol
TG	Task Group

TP	Throughput
Tx	Transmit
VINT	Virtual Inter-Network Testbed
WDS	Wireless Distributed System
WLAN	Wireless Local Area Network
WMN	Wireless Mesh Network
WPAN	Wireless Personal Area Network

1 Introduction

In recent years the deployment of Wireless Mesh Networks (WMNs) has grown in popularity in many metropolitan areas. The deployment of such networks has allowed clients to gain access to publicly available broadband networks. The implementation of WMNs requires that backhaul services (traditionally carried by wired networks) be maintained via wireless mesh points. Because of their structure, WMNs provide an excellent means for targeting a large group of end users or simply to relay data. This can be achieved by means of broadcasting or more specifically multicasting. The wireless broadcast/multicast advantage [WNE00] provides an efficient means of distributing streaming data such as multimedia applications to large groups. The lack of standards and support for multicasting over WMNs makes this area very challenging as well as providing much scope for improvement.

In the following chapters we set out to describe the work carried out as part of a PhD research thesis on multicasting over WMNs. Specifically, the focus of this work is multicasting over WMNs using shortest path trees and the optimisation of such networks. The area of research concerning multicast routing over WMNs is considered to be in its infancy which leaves us with much scope for research and development. Currently there is no support for multicast routing over WMNs in the existing IEEE 802.11 standard [IEE07]. However, at present the IEEE 802.11s amendment [IEE09] is being developed to allow interoperability between heterogeneous mesh network devices. In the Internet Engineering Task Force (IETF), the Mobile Ad Hoc Network (MANET) work group has standardised many multihop routing protocols [IET09]. It is our intention to outline the existing challenges of multicast routing over WMNs, to describe how we intend to adapt the network topology and path selection techniques and to show how we will implement and validate these techniques.

1.1 Problem Statement

In wireless networks the ability to transmit data to all nodes within communications range using a single transmission is known as the broadcast/multicast advantage. In multicast networks, transmissions to the multicast group will involve a single transmission at one of the available physical layer (PHY) line rates. Furthermore, multicast over wireless networks is classed as an unreliable service [RKD06] and does not support acknowledgements (ACKs) or channel reservation mechanism (i.e. RTS/CTS).

Shortest path trees are considered to be optimal for developing multicast networks with minimum delay [Ngu08]. However, such network trees often assume a fixed line rate which is typically set by the network administrator to match the needs of the application serving the group. In order to ensure that all members of the multicast group successfully receive the data, the transmission should take place at the lowest available data rate (i.e. the rate available to the group member with the lowest quality link). By fixing the transmission rate during multicast sessions the multicast advantage is not fully exploited and therefore the performance of such networks can be said to be sub-optimal [ChM05] [CMQ06].

Multicast is a bandwidth-conserving technology specifically designed to reduce traffic by simultaneously delivering a single stream of information. The most significant benefits of multicasting can be seen in high bandwidth applications such as multimedia transmissions where a single transmission can be used (as opposed to multiple, bandwidth consuming, unicast transmissions). By applying a uniform fixed transmission rate, based on the lowest available rate, it will not be possible to support such (high bandwidth) applications.

1.2 Objectives and Contributions

By analysing the construction of multicast trees, links can be identified which result in low line rates being used. This is achieved by identifying the *Child* nodes responsible for reducing the multicast group rate. In this thesis the author proposes two methods that aim to increase the mean network throughput through the use of multiple line rates by the forwarding nodes. The first method identifies specific locations for the placement of additional relay nodes which allows for higher multicast branch line rates to be used. The second method uses a power control algorithm to tune the transmit power to allow for higher multicast branch line rates. The use of power control also helps to reduce the interference caused to neighbouring nodes. A mean network throughput performance increase of 4 to10 times over the single fixed line rate scenario is achieved when using the power adaptation algorithm. The major contributions of this thesis are:

- A novel method of adding relay nodes designed for multicast applications in WMNs.

- A novel power control method designed for multicast applications operating in WMN environment.
- Implementation of a novel connection-oriented simulator for WMN.
- Extensive evaluation of routing protocols using state of the art link cost metrics when applied to multicast traffic in multirate WMNs.
- Implementation of statistical and visual analysis of topological influence on the performance of the evaluated mechanisms.

1.3 Organisation

This thesis is organised as follows.

Chapter 2 describes the main technologies used throughout the course of the research by introducing the general technical background regarding wireless networks before concentrating on the operation of multicasting. This chapter describes the multicast advantage as well methods of developing multicast trees and spanning trees in general. An overview of simulation techniques is given as well as a brief discussion regarding channel models and search optimisation methods.

Chapter 3 provides a summary of some of the open issues regarding WMNs and more specifically multicasting over WMNs. The chapter highlights the recent advances in research through a thorough literature review, regarding WMNs and how it applies to this thesis.

Chapter 4 describes the methodology and design approach used throughout the course of this thesis. A detailed description is given regarding each stage of the simulation process. A full description of the optimisation algorithms is given as well as all assumptions regarding the simulation model. Source code for each simulation as well as scripts used to process data files can be found in Appendix E. Copies of diagrams used in this chapter can be found in Appendix G.

Chapter 5 presents the results in 3 main sections. The first section details the characteristic performance and progressive design of what is termed the Basic Model. The following two sections present the results for adapting the network topology through introducing relay nodes and through tuning the transmit power. A comparison of the performance of the fixed line rate network and that of the power optimised network is given in order to further highlight the advantages of our approach. A brief description of a practical implementation of the power adaptation is then discussed. A full set of diagrams used throughput this chapter can be found in Appendix G.

Chapter 6 presents a summary of the main findings and conclusions from the work carried out. It also suggests areas of further research.

Appendix: Copies of all diagrams, plots, data and source code used throughout this thesis is included in the appendix. The appendix includes a CD-ROM containing all relevant material.

2 Technical Background

This section presents a summary of the main technologies used throughout the course of the research. The section starts by introducing the general technical background regarding wireless networks before concentrating on the operation of multicasting. Our research is primarily concerned with the operation of multicasting over wireless mesh networks (WMNs) hence we describe the multicast advantage in addition to methods of developing multicast trees and spanning trees in general. Our investigation of multicast networks leads us to develop some simulation models. To this end we compare a selection of simulators that have been developed for both academic and commercial use. An integral part of our algorithm for optimising the multicast performance of WMNs involves tuning the transmit power, therefore we describe a channel model in order to introduce the concepts of radio propagation. Finally we describe methods used for optimal search techniques. We will later see in section 4.4 how we use these techniques in order to find an optimal spanning tree using our algorithm.

2.1 Wireless Mesh Networks (WMNs)

Wireless networks have become omnipresent, with widespread applications in the public, military, and business sectors. With cheap and reliable products, Wi-Fi and Bluetooth have created and developed new mass markets. Similar to the evolution of wired networks, current wireless networks form isolated communication groups without any interconnection between them. In market-relevant wireless technologies, network control is often centralised. According to IEEE 802.15.3 standard, Bluetooth, WiMAX and Wireless Personal Area Networks (WPANs) form star topologies with a Central Controller (CC), Base Station (BS) or Piconet Controller (PNC) in the centre. At present most of the deployed IEEE 802.11 Wireless Local Area Networks (WLANs) operate in infrastructure mode where a central Access Point (AP) is present. Although channel access in such configurations is decentralised, all traffic in the network flows via the AP. In contrast, under the IEEE 802.11 ad-hoc mode, stations send their traffic directly to the target destination, which must be within the ad-hoc network. More recently however, the Wi-Fi Alliance announced details [WFA09] for the specification of Wi-Fi Direct, under which Wi-Fi enabled devices will be able to connect directly to each other via point-to-point communication. The wireless networks proposed by the WiMedia Alliance [WMA08] and a proposal for a mesh-distributed coordination function [ZWE05] are the only ones that operate under decentralised control. All others use a single central node that is responsible for relaying traffic to destinations in and out of the local network [WMB06].

2.1.1 Architecture Overview

In 2004, a task group (TG) was formed to define the Extended Service Set (ESS) Mesh Networking Standard. The standard draft amendment (802.11s) has emerged as a single proposal selected from various proposal characteristics from various organisations [IEE09]. The goal of the committee is to develop an IEEE 802.11 extended service set (ESS) mesh that would be built on top of the current 802.11a/b/g standards using the IEEE 802.11 wireless distributed system (WDS). The nodes will be able to automatically discover each other and form mesh networks that support both broadcast/multicast and unicast delivery using radio-aware metrics. For security, all of the APs will be controlled by a single logical administrative entity using the IEEE 802.11i-based mechanism. Quality of service (QoS) standards will also be built into the standards to enable the network to prioritise between different classes of traffic. According to [IEE06], a WLAN Mesh is defined as an IEEE 802.11-based wireless distribution system (WDS) which forms part of a distribution system (DS). The WDS will consist of a set of two or more Mesh Points (MP) interconnected via IEEE 802.11 links and communicating via the WLAN Mesh Services. A WLAN Mesh can support zero or more entry points (Mesh Portals), automatic topology learning and dynamic path selection (including multiple hop paths). Mesh networks have advantageous properties in terms of robustness, range extension and node density. However, mesh networks also have potentially significant disadvantages. In particular, large power consumption (in the case of mobile nodes) and security vulnerabilities are typical problems with such networking topologies.

In most WLAN deployments today, there is a clear distinction between the devices that comprise the network infrastructure and the devices that are clients that simply use the infrastructure to gain access to network resources. The most common WLAN infrastructure devices deployed today are access points (APs) that provide a number of services. For example, they provide support for power saving by means of buffering traffic. APs also provide support for authentication services and access to the wired network. APs are usually directly connected to a wired network (e.g. through an IEEE 802.3

Ethernet interface). They simply provide wireless connectivity to client devices rather than utilising wireless connectivity themselves. Client devices, on the other hand, are typically implemented as stations (STAs) that must associate with an AP in order to gain access to the network. These STAs are dependent on the AP with which they are associated in order to communicate with other STAs.

In its current state all existing wireless standards need bridging (relaying in layer 2) or routing (relaying in layer 3) functionality to connect with other networks that can be based on wire or be wireless. As a common way to bridge data in current IEEE 802.11 networks most existing WLAN APs provide an Ethernet port to interconnect the WLAN segment with the wired IEEE 802.3 segment. An example of the non-mesh WLAN deployment model and device classes are illustrated in Figure 2-1 below. The wired network provides connectivity to other APs. Data can be forwarded from the source to the final destination with the APs working as bridging devices that use the wired network for frame exchange. Devices that use radio to forward (relay) data between different 802.11 Basic Service Sets (BSSs) work similarly as shown in Figure 2-1 except that the wired link is replaced by a wireless link.

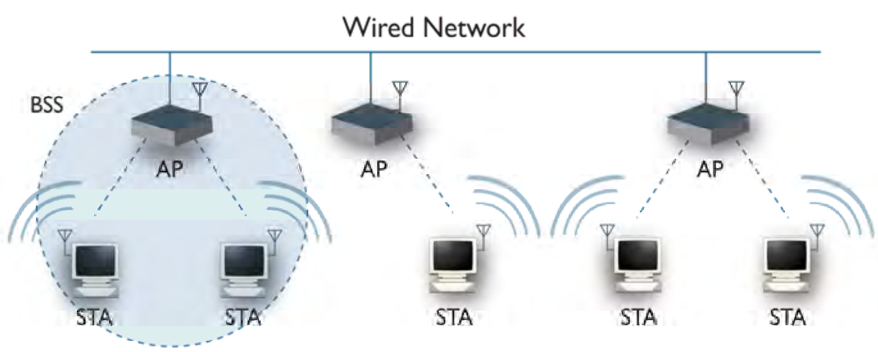


Figure 2-1: WLAN BSS deployment model.

[IEE06] states that there is no reason why many of the devices under consideration for use in WLANs, cannot support much more flexible wireless connectivity. Dedicated infrastructure class devices such as APs should be able to establish peer-to-peer wireless links with neighbouring APs to establish a mesh backhaul infrastructure, without the need for a wired network connection to each AP. Moreover, in many cases devices traditionally categorised as clients should also be able to establish peer-to-peer wireless links with neighbouring clients and APs in a mesh network. In some cases, these mesh-enabled client devices will even provide the same services as APs to help STAs gain access to the network. In this way, the mesh network extensions in this specification blur the lines between infrastructure and client devices in some deployment scenarios. Furthermore, with the recently announced [WFA09] Wi-Fi Direct specification lines of distinction will become even less clear.

The architecture specified above [IEE06] divides wireless nodes into two major classes, mesh class nodes and non-mesh class. Mesh class nodes are capable of supporting mesh services, while the non-mesh class includes simple client STAs. Mesh class nodes can optionally also support AP services and can be managed or unmanaged.

An example of a WLAN Mesh Network (WMN) is illustrated in Figure 2-2. Any devices that support mesh services are mesh points (MPs). A mesh point can be either a dedicated infrastructure device or a user device that is able to fully participate in the formation and operation of the WMN. A special type of Mesh Point is the mesh access point (MAP), which provides AP services in addition to mesh services. Simple STAs associate with Mesh APs to gain access to the WMN. Simple STAs do not participate in WMN services such as path selection and forwarding, etc. Mesh points can operate at various levels of functionality. Not all mesh points need to use the full mesh services. Also services like routing can be used partially or not at all.

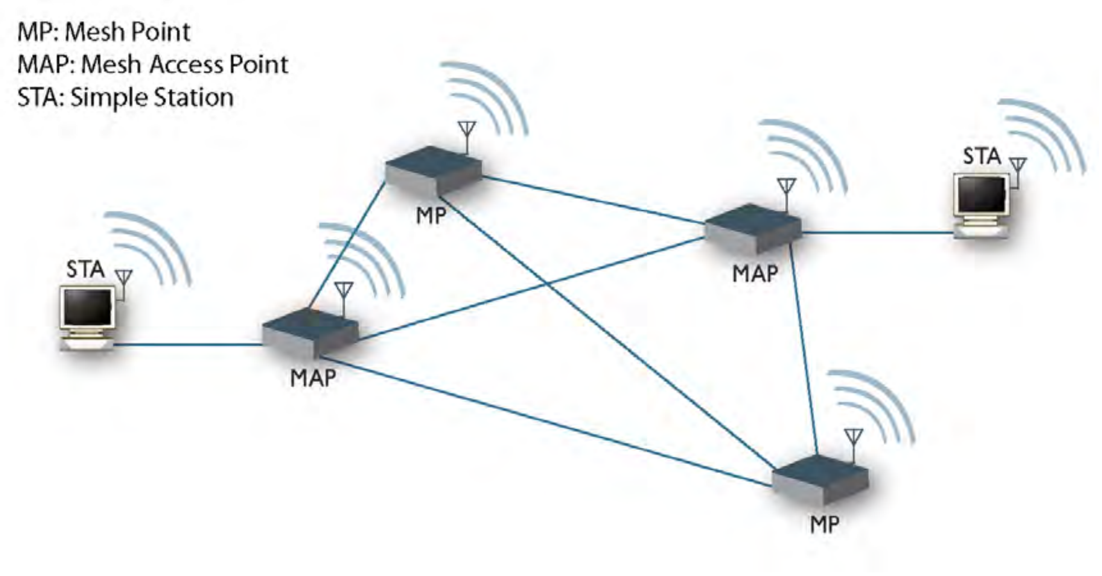


Figure 2-2: Example of a WLAN Mesh Network.

The basic characteristic of a WMN is the capability to relay frames from one device to another. Figure 2-3 shows an example of the coverage extension of Internet access via relaying devices. To be able to relay data from a source device to the final destination device, sufficient addressing information must be provided.

In IP networks, the network address is used to forward data by means of multi-hop from source to destination. IP routers exchange information on their attached networks and advertise known routes. Routers are interconnected and provide the relaying service for devices in their attached networks. In an IP network, the source device requests relaying of a frame by its local network serving router, a Relay Node (RN) or gateway.

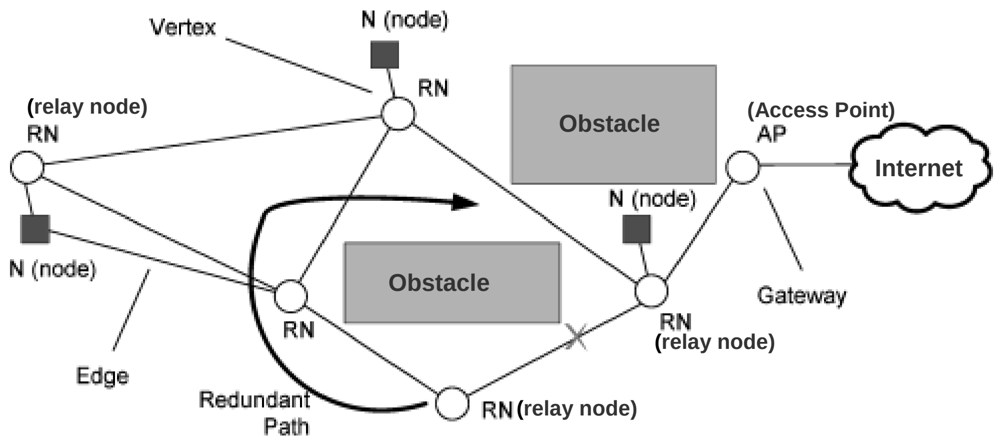


Figure 2-3: Example of coverage extension using a WMN [WMB06].

Several routers along the routing path forward the frame. At the final router, the frame is delivered to the destination inside an attached local network. In an IP network, devices use the network mask to identify devices outside the local network. Figure 2-4 shows a single router that interconnects several IP networks. Routing decisions are based on network addresses.

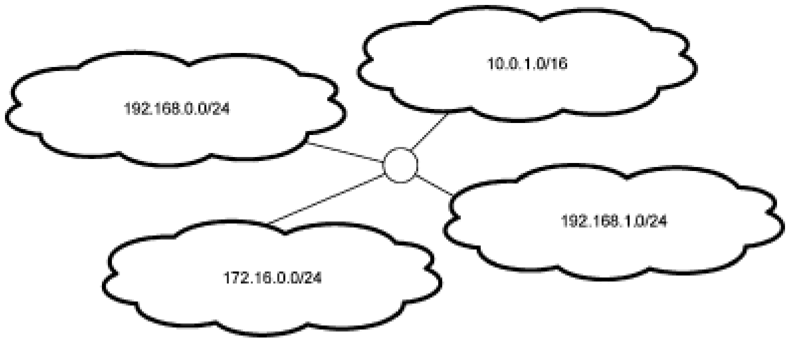


Figure 2-4*:* Single router acting as an IP network interconnect [WMB06].

For destinations inside the local network, direct frame exchange is possible. The local network forms a subset of all existing devices. Thus, devices are able to communicate with a subset of all devices only. To mutually address each other, devices use broadcast messages in a local network. Hence a local network is also referred to as a broadcast domain. Within a broadcast domain, each device communicates with other devices without the help of any intermediate node. For destinations outside the broadcast domain, devices rely on routers to forward their frames. Hence, routers represent a set of devices. In terms of addressing, they work as proxies for their attached networks. A set of devices can be addressed through a single router. This hierarchy of relaying capable and non-capable devices ensures that the size of the routing tables remain manageable.

In contrast to single-hop networks, where most of the traffic is directed to and received from a central device, WMNs can have no hierarchy. The wireless medium is a shared resource that is used by all entities of the mesh network. In some cases the wireless medium can even be shared with non-mesh capable devices that are also served by the mesh network. Other neighbouring or co-located non-mesh networks can be present, especially in WLAN and WPAN environments, which usually use unlicensed frequency bands. The common resource that is shared among the devices participating in a WMN can be a hostile environment.

As most wireless technologies tend to define layers 1 and 2 of the ISO/OSI Reference Model only, their topology is flat. Therefore, wireless standards usually define a single broadcast domain only and no routing function is defined. Any frame relaying needs to be handled by higher layer protocols. While in traditional wireless single-hop networks all devices are in either mutual reception range or have a common central neighbour (usually the AP), in WMNs multiple direct and indirect neighbours can exist that do not necessarily have an intersection of their sets of neighbours. The Medium Access Control (MAC) protocol supporting the WMN needs to take this into account. Furthermore, a WMN introduces multi-hop links inside the broadcast domain. To enable higher protocols to work transparently over a wireless mesh network, routing needs to be handled by each relay device in the broadcast domain. The identification of possible hops from source to destination is called routing in the IP layer. To distinguish routing from the respective function in the mesh MAC layer, it is referred to as path selection. However, the basic function remains the same namely that devices need to determine who their neighbours are and to propagate the information across the network by relaying frames.

WMNs can consist of devices mutually unknown to each other. These devices can mutually provide services in terms of path selection and frame forwarding. Therefore, security support in WMNs is more complex than in single-hop networks. Trustful relationships between devices will not always exist. Hence, end-to-end security support differs from single-hop link security. If unknown devices participate in the mesh network, path selection will become impossible. Invalid path information can be provided by attackers; hence frames will be relayed to false devices. In all wireless mesh networks, a hop counter will prevent infinite frame forwarding and loops.

2.1.2 Classification of Wireless Mesh Networks

WMNs can operate with or without a hierarchical structure. In a flat hierarchy, any wireless device in the network is able to forward frames. In such networks, a device does not solely operate as a sink or source of traffic, but can also accept packets that are not directed to itself in order to relay frames to neighbouring devices. Each device in such networks needs path selection functionality and the capability to support multi-hop traffic.

In hierarchical mesh networks, only mesh capable devices provide the mesh networking service to other non-mesh capable devices that do not have relaying capabilities. Non-mesh devices associate with the mesh devices. Typically, mesh capable devices are APs. A hierarchical mesh network is sufficient for static mesh networks where fixed APs form the backhaul mesh network to provide ESS service for mobile client devices. Only the mesh devices need extra resources such as memory, enhanced computing power and multiple transceivers to be able to form the WMN. As APs are fixed and connected to a mains power supply, power-saving algorithms are not a concern. This situation is quite different for mobile devices that need to optimise the use of their battery power. In addition, location aware packet relaying protocols can be applied to exploit the fixed nature of the network.

With regard to frequency channels used, WMNs in comparison to the BSS can operate in-band or out-of-band for the purpose of signalling. WMNs operate on single or multiple frequency channels. In a single channel mesh network, single-hop frames (inside a BSS) as well as multi-hop frames (in an ESS) propagate in the same channel. Coexistence support is necessary then, and traffic segregation is needed to provide the mesh network with the necessary resources to relay frames generated remotely when competing with frames that propagate locally to a BSS.

Multiple channels are operated using a single radio or multiple radios in mesh devices. Thus traffic segregation is possible, where single-hop and multi-hop frames are transmitted on different frequency channels. Achieving separation by operating BSSs on channels different from that used for meshing (ESS) will be inferior, in terms of overall capacity, to dynamic channel assignment or even sharing common channels. Figure 2-5 provides a classification of WMNs in terms of the numbers of channels and radios used and the manner in which channels are shared between BSS and ESS services.

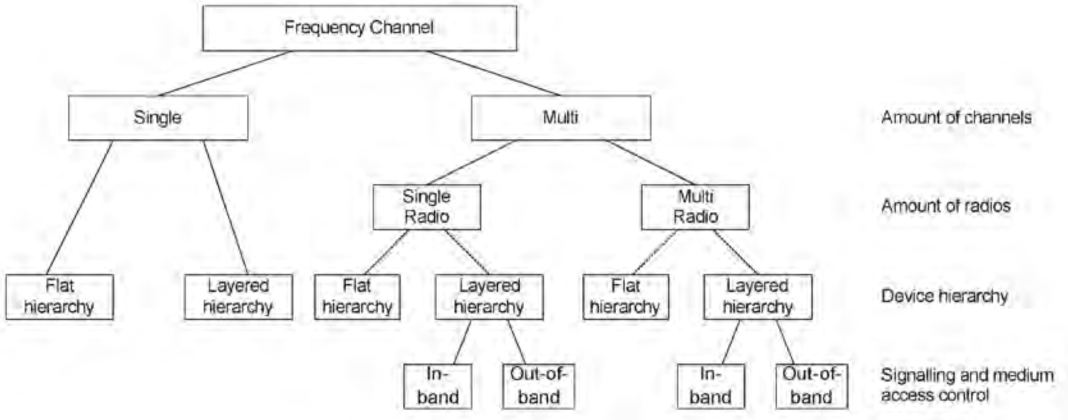


Figure 2-5*:* Wireless Mesh Network classification [WMB06].

Another classification of mesh networks can be derived from the MAC protocol used. IEEE 802.11 uses an asynchronous medium access mechanism under decentralised control, while IEEE 802.16 and 802.15.3 are based on synchronised medium access.

2.1.3 General Problem Statement

New challenges emerge from the advent of WMNs. In contrast to single-hop networks, the transmission of a multi-hop frame reduces the end-to-end data throughput while also increasing the overall latency/delay. This characteristic multi-hop nature of WMNs can severely impact their performance. While efficient routing algorithms have been developed for wired networks, self-interference of relayed frames and unpredictable path metrics are the main challenges for developing new and efficient routing algorithms for WMNs. While mesh networks are broadcast in nature (every node within broadcast range can potentially receive a transmitted packet), this feature has both advantages and disadvantages. Consequently, for multirate multicast WMNs the “Crybaby” problem exists where the whole multicast group suffers due to the problems of one member.

2.1.4 Path Selection

Bellman-Ford, Dijkstra and Floyd-Warshall [CLR02] provide generic routing algorithms that form the basis of most existing routing protocols in wired networks. Examples of the metrics used by these protocols to calculate the optimum route are hop count, link speed, cost for transiting traffic, and delay. These metrics are used to provide the weight for edges when applying graph theory.

Since the bit error ratio is usually negligible for wired and optical networks, the data rate and delay tend to be relatively constant, compared to route updating time intervals or frame transmission duration. Consequently, routing algorithms in the wired Internet do not take frequent changes of topology and link speed into account.

With wireless networks, topology (i.e. the connectivity between nodes) and link speed can change rapidly. Roaming devices and moving obstacles can cause frequent topology and link cost changes in both infrastructure based and ad-hoc WMNs, causing changes in the load of relay nodes and mutual interference of network internal nodes and with nodes of foreign networks. Path metrics of wired networks appear insufficient for WMNs. *Vertices* (wireless devices or nodes) and *edges* (links between wireless devices) cannot be considered stable in wireless networks resulting in frequent change of the topology. This is in contrast to wired networks where the status, availability and characteristics of vertices and edges change slowly. Depending on the network size and application, routing graphs in wired networks have long periods of stability ranging from hours to months or even years. In wireless networks such stability is unlikely to be achieved.

Appropriate path metrics for mesh networks that can provide for more accurate path selection decisions will additionally consider:

- Packet error probability that depends on signal-to-interference-plus-noise ratio (SINR) reflecting the current PHY mode used for a given link, antenna gains, transmission power, background noise level, and frame length used by the MAC protocol.
- Congestion status of receiving relay node in the mesh network.
- Availability of a relay node on a certain frequency channel.
- Bandwidth needed for transmission.

In wireless networks, all the path selection metrics mentioned above are time variant and will change dramatically within a short duration. Therefore, to calculate the optimum path at any given time, it would be impractical to estimate the required parameter values of each metric. Furthermore, this information is available in the MAC layer only. Existing standards do not provide an interface to support information exchange with the routing layer to provide these parameters. Hence, ad-hoc path selection (routing) in WMNs must operate blindly.

Apart from the working assumptions made by the Internet Engineering Task Force (IETF) group "Mobile Ad-hoc Networks" [IET09], [MaC06], WMNs developed using IEEE technology only cover layers 1 and 2 and must provide transparency to higher layers. The ad-hoc routing protocols developed at MANET cannot be used in WMNs, since frame forwarding is performed in the IP layer. The IEEE aims at path selection (routing) protocols realised in the MAC ("layer 2.5") to provide multi-hop forwarding of unicast, multicast and broadcast frames in the MAC layer. The WMN is considered a single LAN segment that forms a unified broadcast domain.

The ad-hoc routing protocols for WMNs developed by MANET of the IETF reside in the IP layer. Since no interfaces for parameter values exchange exists within the MAC layer, routing decisions are based on a small set of routing metrics. Since the IP layer lacks these metrics for decisions on the preferable paths, the MANET routing protocols use frequent IP broadcast frames to exchange topology information between the relay nodes involved, where IP (layer 3) broadcast frames are mapped onto MAC frames with the receiver field set to the broadcast address. Inside the broadcast domain, which is limited by the actual transmission range of the broadcasting device, other devices are periodically being informed about the senders routing tables and its view of the network topology.

2.1.5 Medium Access Control

Apart from single-hop networks, relaying in multi-hop networks introduces new problems that cannot be solved by just applying single-hop MAC protocols multiple times in sequence. A WMN can be seen as the sum of a number of continuously overlapping neighbouring single-hop networks. In single-hop wireless networks all devices in the network are either within the mutual receiving range or have a common central device that is within the receiving range of all other devices. However, a WMN provides frame exchange among devices that are not within the mutual receiving range. In such cases the source and the final destination nodes will not be able to exchange information directly. Hence, coordination of their channel access in an area larger than that of a single-hop network is needed. The hidden and exposed node problems, when not handled properly, cause much more severe problems in WMNs than with single-hop networks.

2.2 IP Multicasting

There are three fundamental types of IPv4 addresses: unicast, broadcast, and multicast. A unicast MAC address is intended for one device on a network segment (or sub-network). A broadcast address is used to send a datagram to every device connected to a network segment. A multicast address is designed to enable the delivery of datagrams to a set of hosts that have been configured as members of a multicast group in various scattered sub-networks. A multicast frame is destined for a device within a dynamic multicast group on a network segment.

Multicasting over LAN is not connection oriented. A multicast datagram is delivered to destination group members with the same “best-effort” reliability as a standard unicast IP datagram. This means that a multicast datagram is not guaranteed to reach all members of the group, or arrive in the same order relative to the transmission of other packets.

Traditional IP communications allow a host to send packets to another host (unicast transmissions) or to all hosts (broadcast transmissions). IP Multicast provides a third communication alternative: allowing a host to send packets to a group that is made up of a subset of the hosts on the network. IP Multicast is a bandwidth-conserving technology specifically designed to reduce traffic by simultaneously delivering a single stream of information to potentially thousands of corporate recipients or homes. By replacing copies for all recipients with the delivery of a single stream of information, IP Multicast is able to minimise the burden on both sending and receiving hosts and reduce overall network traffic. Within a multicast network, routers are responsible for replicating and distributing multicast content to all hosts that are listening to a particular multicast group (see Figure 2-6).

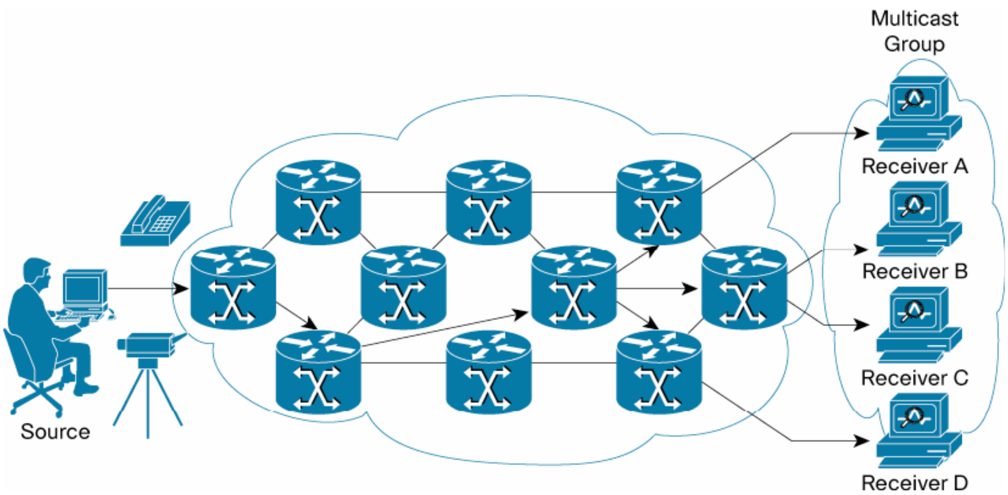


Figure 2-6: Multicast transmission over IP to many receivers [CWP07].

IP Multicast solutions offer benefits relating to the conservation of network bandwidth. In the case of a high-bandwidth application, such as MPEG video, IP Multicast can benefit situations with only a few receivers because a few video streams would otherwise consume a large portion of the available network bandwidth. Even for low-bandwidth applications, IP Multicast conserves resources when transmissions involve thousands of receivers. Additionally, IP Multicast is the only non-broadcasting alternative for situations that require simultaneously sending information to more than one receiver. This allows multicast receivers to join more than one multicast group in order to receive from multiple sources.

For low-bandwidth applications, an alternative to IP Multicast can involve replicating data at the source. This solution, however, can deteriorate application performance, introduce latencies and variable delays that impact users and applications, and require expensive servers to manage the replications and data distribution. Such solutions also result in multiple transmissions of the same content, consuming an enormous amount of network bandwidth. For most high-bandwidth applications, these same issues make IP Multicast the only viable option. A summary of real-time and non real-time applications which use multicasting is given in Table 2-1.

	Real Time	**Non-Real Time**
Multimedia	IPTV Live Video Video Conferencing Live Internet Audio IP Surveillance	Replication Video, Web Servers, Kiosks Content Delivery
Data-Only	Stock Quotes News Feeds White Boarding Interactive Games e-learning	Information Delivery Server to server Server to desktop Database replication Software distribution

Table 2-1*:* Types of IP Multicast Applications.

IP Multicast is supported in IPv4 and IPv6 networks, Multi-protocol Label Switching (MPLS) VPNs as well as mobile and wireless networks. IP Multicast capabilities can be deployed using a variety of different protocols, conventions, and considerations suited to the different network environments just mentioned. Multicast services can also be deployed across multiple protocol platforms and domains within the same network.

An IP Mobility platform extends the network with traditional fixed-line access to an environment that supports mobile wireless access. Multicast, from the point of IP Mobility, is a network service or application. Within an IP Mobility environment, IP Multicast can be employed to deliver content to users with wireless devices [CWP07].

2.2.1 Multicast Groups

Individual hosts are free to join or leave a multicast group at any time. There are no restrictions on the physical location or the number of members in a multicast group. A host can be a member of more than one multicast group at any given time and does not have to belong to a group to send messages to members of a group. The only difference between a multicast IP packet and a unicast IP packet is the presence of a “group address” in the Destination Address field of the IP header. Instead of a Class A, B, or C IP address, multicasting employs a Class D destination address format (224.0.0.0-239.255.255.255).

2.2.2 Group Membership Protocol

A group membership protocol is employed by routers to learn about the presence of group members on their directly attached sub-networks. When a host joins a multicast group, it transmits a group membership protocol message for the group(s) that it wishes to receive, and sets its IP process and network interface card to receive frames addressed to the multicast group. This receiver-initiated join process has excellent scaling properties since, as the multicast group increases in size; it becomes ever more likely that a new group member will be able to locate a nearby branch of the multicast distribution tree.

2.2.3 Multicast Routing Protocol

Multicast routers execute a multicast routing protocol to define delivery paths that enable the forwarding of multicast datagrams across an inter-network. The Distance Vector Multicast Routing Protocol (DVMRP) [WPD88] is a distance-vector routing protocol, and Multicast Open Shortest Path First (MOSPF) [Moy94] is an extension to the OSPF [Moy98] link-state unicast routing protocol.

Broadcast and multicast frames have the simplest frame exchanges because there is no acknowledgment. A generic 802.11 MAC frame is illustrated in Figure 2-7 . Framing and addressing are somewhat more complex in 802.11, so the types of frames that match this rule are the following:

- Broadcast data frames with a broadcast address in the Address1 field.
- Multicast data frames with a multicast address in the Address1 field.
- Broadcast management frames with a broadcast address in the Address1 field (Beacon, Probe Request, and IBSS ATIM frames).

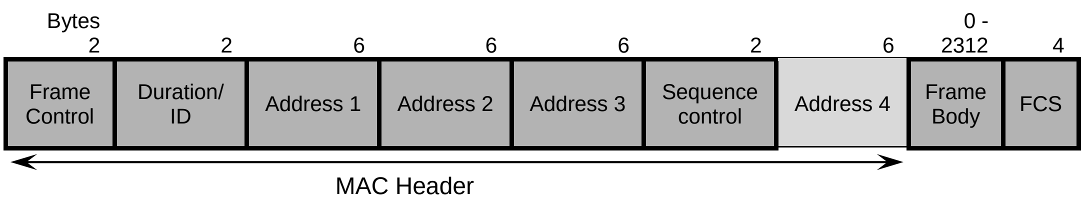


Figure 2-7: A generic 802.11 MAC frame.

Frames destined for group addresses cannot be fragmented and are not acknowledged. The entire atomic sequence is a single frame, sent according to the rules of the contention-based access control. After the previous transmission concludes, all stations wait for the time period DIFS (Distributed Inter-Frame Space) and begin counting down the random delay intervals in the contention window.

Because the frame exchange is a single-frame sequence, the network allocation vector (NAV) is set to 0. With no further frames to follow, there is no need to use the virtual carrier-sense mechanism to lock other stations out of using the medium. After the frame is transmitted, all stations wait through the DIFS and begin counting down through the contention window for any deferred frames. See Figure 2-8 below.

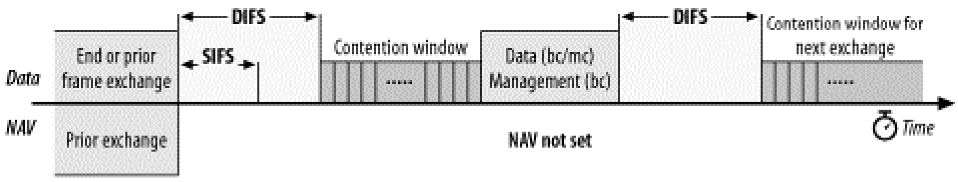


Figure 2-8: Broadcast/multicast management atomic frame exchange [Gas02].

Depending on the environment, frames sent to group addresses can have lower service quality because the frames are not acknowledged. Some stations will therefore miss broadcast or multicast traffic, but there is no facility built into the MAC for retransmitting broadcast or multicast frames [Gas02].

2.3 Multicasting over Wireless Networks

Multicast communications has been well supported for fixed wired networks for close to 20 years. Multicast protocols are used to generate a hierarchical tree containing hosts as part of a multicast group connected to multicast routers. The main functions of a multicast router are to forward multicast datagrams and to determine multicast group membership. Multicast group membership is determined by periodically broadcasting group membership probes (or query requests) which will be received by each attached device.

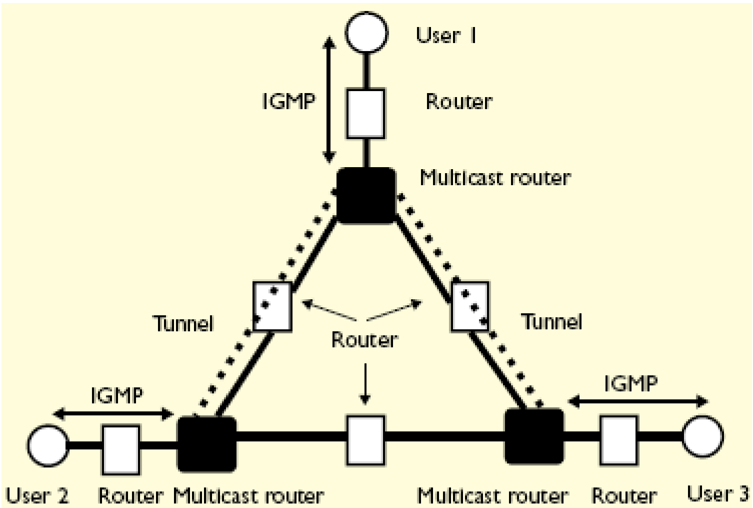


Figure 2-9: IP Multicast support using tunnelling [Var02].

The Internet Group Management Protocol (IGMP) [Dee89] handles host-to-router communication. IP tunnels are used to encapsulate multicast packets in a unicast packet so that multicast routers can communicate via non supporting IP routers. Figure 2-9 illustrates the basic operation of multicast host-to-router communication.

Extending existing multicast support to wireless networks is not a trivial task. For example, an asymmetrical wireless link [KNE03] allows will result in poor signalling or the use of a low bandwidth connection from/to the host. Table 2-2 outlines some of the issues concerning multicasting over wireless compared to wired networks.

Issue	Current "wired" multicast	Wireless and mobile multicast	Possible ways to support wireless and mobile multicast
Type of links	Symmetrical and fixed characteristics Broadcast links in LANs	Possibly asymmetrical and/or unidirectional links of varying performance and point-to-point links in cellular and PCS	Design of new protocols to handle route asymmetry and unidirectional links without reverse-path information (possible history and prediction-based schemes)
Bandwidth	Plentiful	Limited and variable amount	Protocols to adapt membership management and routing updates to the amount of bandwidth available and user mobility
Topology	Fixed	Fixed in infrastructure-based, dynamic in ad hoc networks	Protocols for both fixed and changing topology by "sensing" topological changes
Loss of packets	Infrequent (<1%)	Frequent and variable (1%–30% based on links)	Error control with possible retransmission from neighboring user(s)
Membership changes	Only when a user leaves or joins a group	Also when a user moves to another location	Protocols with reduced overhead for managing membership
Routing	Fixed routing structure throughout the multicast session	Routing structure subject to change due to user mobility	Protocols that could dynamically adapt the routing to current structure and available resources
Security Issues	Less complex due to fixed users and wired links	More complex due to wireless links and possible use of broadcasting	Encryption and security techniques in routing and membership management
Quality of service	Individual routes can use RSVP	Due to user mobility, RSVP may cause excessive overhead	Design of new protocols for "soft" QoS under varying link conditions and mobility
Reliability	Possible use of a transport-layer protocol (such as the Multicast File Transfer Protocol)	More complex due to wireless links and user mobility; possible unwanted interaction of protocols at transport and link layers	Design of new protocols that could allow one or more different retransmission schemes at one or more protocol layers

Table 2-2: Comparison of multicast issues over wired & wireless networks [Var02].

Existing multicast protocols are designed for fixed topologies and as such the problems increase for ad hoc networks which exhibit a high degree of mobility.

The complexity of the radio links in wireless networks makes it necessary to modify the existing IGMP for successful operation. Challenges include overcoming the unreliability of group queries/responses on wireless links, reducing the overhead generated from IGMP [KiH04] and managing leave latency [XyP97] (i.e. losses due to node mobility).

2.3.1 Multicast Routing for Wireless Networks

Routing protocols designed for infrastructure based wireless networks are not well suited to wireless mesh and ad-hoc wireless networks. This is largely due to mobility and in the case of ad-hoc, a lack of infrastructure. Figure 2-10 outlines several issues and possible solutions in multicast routing for ad hoc networks. The diagram highlights the increased activity due to mobile nodes which would in turn require additional management of group membership. Several multicast routing protocols have been adapted to operate in wireless networks. These include;

- Distance Vector Multicast Routing Protocol (DVMRP) [WPD88]
- Multicast Open Shortest Path First (MOSPF) [Moy94]
- Protocol Independent Multicast, Sparse Mode (PIM-SM) [EFH98], Dense Mode (PIM-DM) [ANS05]
- Core Based Tree (CBT) [BFC93]
- Ad-hoc On-demand Distance Vector (AODV) [RoP99]
- On-Demand Multicast Routing Protocol (ODMRP) [LGC99]
- Forwarding Group Multicast Protocol (FGMP) [LSG99]

A detailed and in-depth discussion on the operation of these protocols is given in [Var02]. We will consider the different aspects and latest developments of multicast routing in further detail in the next chapter.

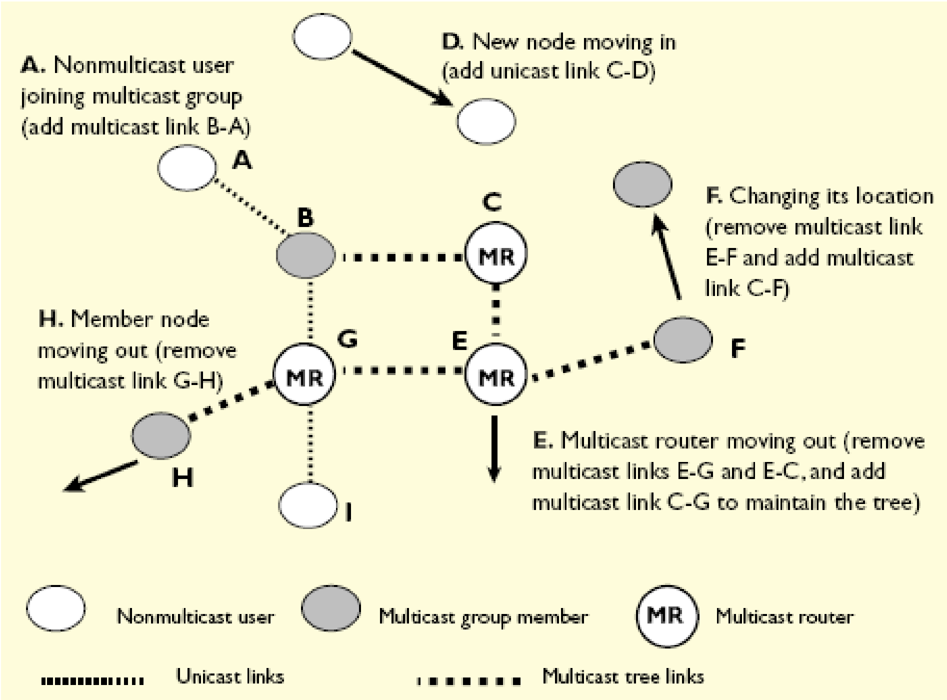


Figure 2-10: Multicast routing in ad-hoc networks [Var02].

2.4 The Multicasting Advantage

In [WNE00] the authors describe the multicast advantage by describing the operation of a network consisting of *N* nodes randomly distributed over a specified region. The authors clearly define the working parameters of all devices in the multicast tree (i.e. source, destination and relay nodes as well as antenna type and transmit power). The paper shows how a single transmission from a source node is sufficient to communicate with all neighbouring nodes if the transmit power is set to the maximum required to reach an individual node. This can be illustrated by the example as shown in Figure 2-11. In the diagram node i is the source transmitting to its neighbours, node j and k. The power required to reach node j is P_{ij} and the power required to reach node k is P_{ik}. A single transmission at power $P_{i,(j,k)} = \max\{P_{ij}, P_{ik}\}$ is sufficient to reach both node *j* and node *k* (based on the assumption of omni-directional antennas).

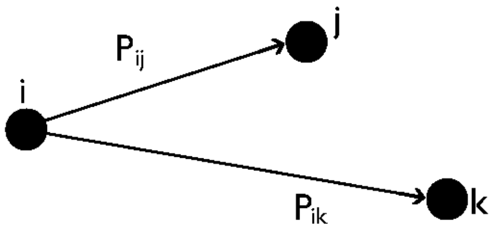


Figure 2-11: Example of multicast/broadcast advantage [WNE00].

The authors refer to the ability to exploit this property of wireless communication as the “wireless multicast advantage”

As a result, the wireless multicast advantage is characterised by [WNE00] by the following properties:

- “A node’s transmission is capable of reaching another node if the latter is within the communication range which in turn means that the received SINR exceeds a given threshold and that the receiving nodes have allocated (scheduled) receiver resources for this purpose.”
- “The total power required to reach a set of other nodes is simply the *maximum* required to reach any of them individually.”

2.5 Network Spanning Tree

In developing network broadcasting techniques a method known as flooding can be used to deliver information from a point of origin to all other nodes connected to a network. The basic principal behind flooding is for the origin to transmit information to all its neighbours. The neighbours in turn transmit this information on to their neighbours until all nodes have received the transmission. There are two additional basic rules to flooding; a node will not transmit a packet back to the node from which it received the transmission; a node will not forward the same transmission more than once to the same neighbour.

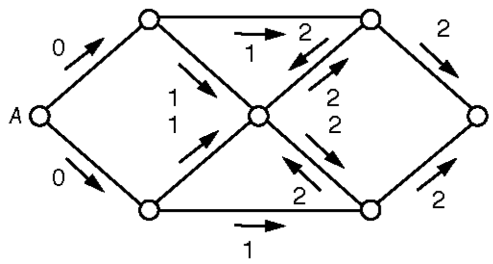


Figure 2-12: Example of broadcast flooding [BeG02].

A more communication efficient method of flooding is a technique based on the use of a *spanning tree* (see Figure 2-13). The task of designing a network with a minimal total length is called the *minimal spanning tree* problem (first published by Otakar Borůvka, 1926 [GrH85]). Minimal spanning trees are useful as one of the steps for solving problems on graphs, such as the Travelling Salesman Problem which tries to find the shortest route that visits every point in the network. There are efficient algorithms (methods) for solving minimal spanning tree problems. A simple method that gives an optimal solution is to start with no connections, and add them in increasing order of size, only adding connections that join up parts of the network that weren't previously connected. This is called Kruskal's algorithm after J.B. Kruskal, who published it in 1956 [GrH85], [BeW98].

A spanning tree is a connected sub-graph of the network which includes all the nodes without any unnecessary cycles (i.e. closed loops). Figure 2-13 shows an example of a spanning tree. We can see the lack of cycles in this diagram compared to broadcast flooding in Figure 2-12. Spanning trees require a total of *N-1* packet transmissions per packet broadcast, where *N* is the number of connected nodes. The trade-off is the need to maintain and update the spanning tree as the tree topology changes.

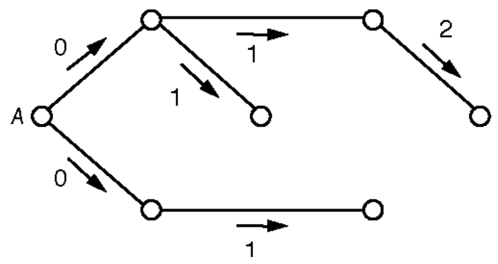


Figure 2-13: Example of a spanning tree [BeG02].

There are two fundamental approaches to multicast routing: *Shortest Path Trees* (SPTs) and *Minimum Cost Trees* (MCTs). The SPT algorithms minimise the distance (or link cost) from the sender to each receiver. MCT algorithms such as Minimum Steiner Trees (MSTs) minimise the overall edge cost of the multicast tree. Figure 2-14 illustrates the basic concept behind minimal link cost and minimal edge cost.

Figure 2-14: Four nodes connected using a SPT. MST connects the same four nodes by placing an additional Steiner node.

Assuming all four nodes are equally placed we can place a value of 1 on each of the square edges which would then give us an edge cost of √2 on the diagonal. For the SPT the minimum cost to a node would be 1 (along either of the square edges). The maximum cost to a single node would therefore be $1+\sqrt{2}$ with an overall tree cost of $2 + \sqrt{2}$. In the MST tree we attempt to lower the overall cost of the tree by introducing additional nodes known as Steiner nodes. In this case the minimum cost to one of the original nodes would be $\sqrt{2}$ which is also the maximum cost. The overall tree cost is $2\sqrt{2}$ which is less than the SPT. We will explore the significance and the consequences [BeW98], [CSU05] of this for wireless networks in the next chapter.

In wireless multi-hop networks, the tree cost can be redefined to exploit the wireless broadcast advantage: a minimum cost tree is one which connects sources and receivers by issuing a minimum number of transmissions (MNT). Among the different approaches, SPT is the more commonly used method for multicast routing in the Internet. The MNT approach was originally considered for energy-constrained wireless networks such as sensor and mobile ad-hoc networks [Ngu08].

There has been extensive research carried out in addressing multicast trees in wired networks. However, this does not necessarily translate directly over to WMNs. The problem of finding minimum cost trees based on Minimum Steiner trees has been shown to be NP-Complete [Kar75]. Minimum Steiner trees are shown to be complex to implement and will not always result in a minimum cost tree when used in a WMN [RuG05]. In [RuG05], Ruiz et al used the minimum number of transmissions as a link cost metric and demonstrate that the problem of finding a MNT tree in a WMN is also NP-Complete. In [RKD06] the authors acknowledge the fact that design goals in WMNs have shifted from maintaining connectivity to providing sufficient throughput. The authors use techniques taken from unicast routing and adapt them for multicasting and provide a comprehensive performance study. We will discuss some of this literature in more detail in the next chapter.

2.6 Shortest Path Problem

As briefly described in the previous section the shortest path problem is the problem of finding a path between two vertices (or nodes). Each communication link is assigned a positive number called its length. A link can have a different length in each direction. A sequence of links, known as a path, between two nodes has a length equal to the sum of the lengths of its links. A shortest path routing algorithm routes each packet along a minimum length (or shortest) path between the origin and destination nodes of the packet. The simplest possibility is for each link to have a unit length, in which case the shortest path is simply the path with the minimum number of links. This is also known as at *minimum hop path*. More generally the length of a link will depend on its transmission capacity and its projected traffic load. The idea is that a shortest path should contain relatively few and uncongested links, and therefore should be desirable for routing.

A more sophisticated alternative is to allow the length of each link to change over time and to depend on the prevailing congestion level of the link. Then a shortest path can adapt to temporary overloads and thus route packets around points of congestion. This idea is simple but also contains some hidden pitfalls, because by making link length dependent on congestion, a feedback effect is introduced between the routing algorithm and the traffic pattern within the network [BeG02]. We will now discuss in more detail three standard algorithms for solving the shortest path problem; the Bellman-Ford algorithm, the Floyd-Warshall algorithm and the Dijkstra algorithm, with the emphasis on the Dijkstra algorithm as it is the algorithm used in this thesis. All three algorithms iterate to find the final solution, but each iterates on something different. The Bellman-Ford algorithm iterates on the number of arcs in a path, the Floyd-Warshall algorithm, as described in [BeG02], iterates on the set of nodes that are allowed as intermediate nodes on the paths, and finally, the Dijkstra algorithm iterates on the length of the path.

2.6.1 Bellman-Ford Algorithm

In [BeG02] the authors define the operation of the Bellman-Ford algorithm as follows. Suppose that node 1 is the destination node then consider the problem of finding a shortest path from every node to node 1. Assume that there exists at least one path from every node to the destination. To simplify the presentation, $d_{ij} = \infty$ if *(i, j)* is not an arc on the graph. Using this convention it can be assumed without loss of generality that there is an arc between every pair of the nodes, since walks and paths consisting of a true network arcs are the only ones with length less than ∞.

A shortest walk from a given node *i* to node 1, subject to the constraint that the walk contains at most *h* arcs and goes through node 1 only once, is referred to as a *shortest (≤ h) walk* and its length is denoted by D_i^h .Note that such a walk will not be a path, that is, if it contains repeated nodes. By convention take

$$D_1^h = 0, \qquad \text{for all } h$$

The length D_i^h can be generated by using the Bellman-Ford algorithm;

$$D_i^{h+1} = \min_j \left[d_{ij} + D_j^h \right] \qquad \text{for all } i \neq 1 \qquad (2.1)$$

Starting from the initial conditions,

$$D_i^0 = \infty, \qquad \text{for all } i \neq 1$$

The algorithm is said to terminate after *h* iterations if,

$$D_i^h = D_i^{h-1}, \qquad \text{for all } i$$

Thus, the Bellman-Ford algorithm claims to first find the one-arc shortest walk lengths, then find the two-arc shortest walk lengths, and so forth. It can then be shown that the shortest walk lengths are equal to the shortest path lengths, under the additional assumption that all cycles not containing node 1 have negative length [BeG02].

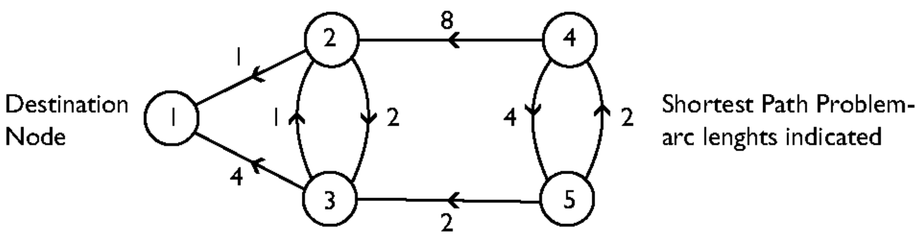


Figure 2-15 (a): Shortest path problem – arc lengths as indicated.

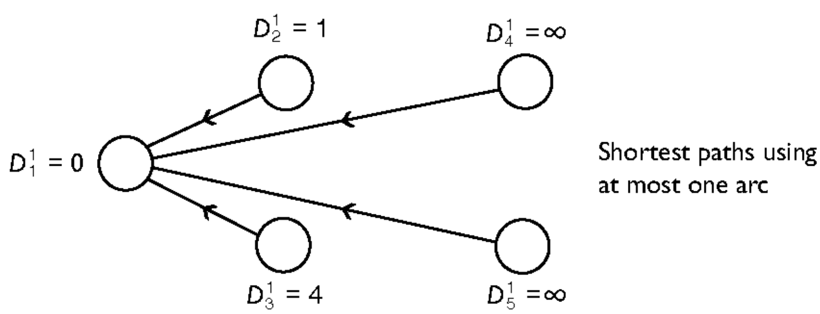


Figure 2-15 (b): Shortest path using at most 1 arc.

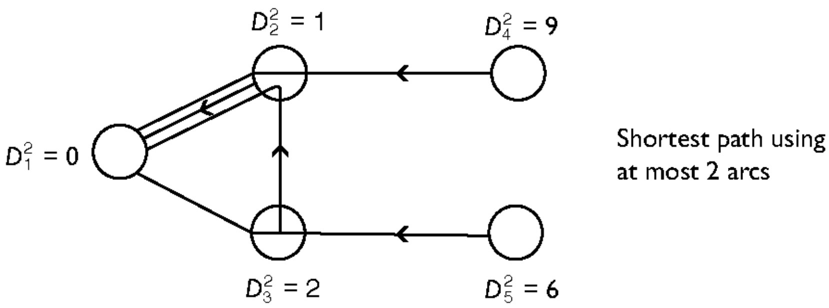


Figure 2-15 (c): Shortest path using at most 2 arcs.

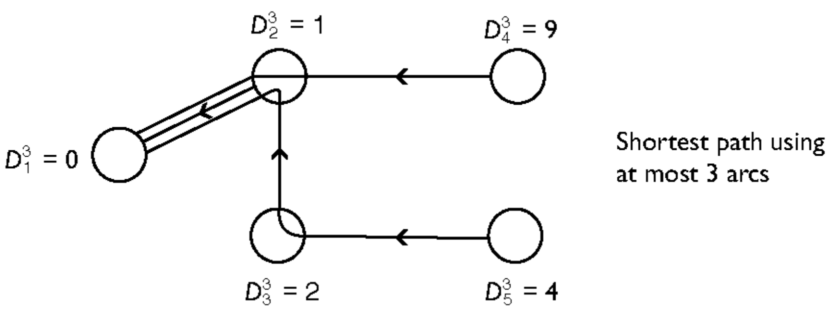


Figure 2-15 (d): Shortest path using at most 3 arcs.

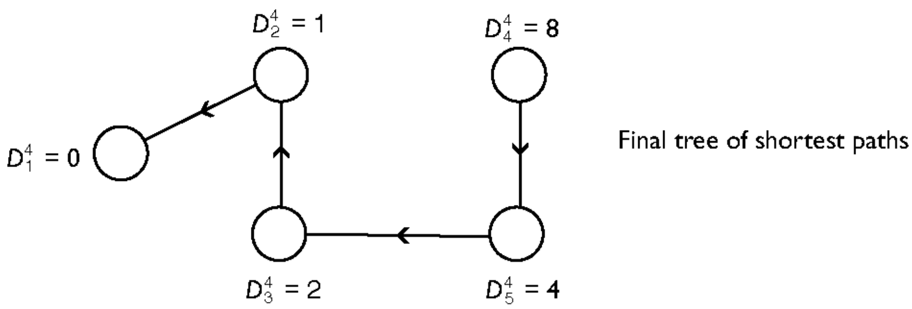


Figure 2-15 (e): Successive iterations of Bellman-Ford algorithm.
Final tree of shortest paths [BeG02].

2.6.2 Floyd-Warshall Algorithm

A well defined description of the Floyd-Warshall algorithm is given in [Beg02] and summarised in this section. This algorithm, unlike the Bellman-Ford and Dijkstra algorithms finds the shortest path between all pairs of nodes together. Like the Bellman-Ford algorithm, the arc distances can be positive or negative, but again there can be no negative-length cycles. The Floyd-

Warshall algorithm starts like both of the other algorithms with single arc distances (i.e. no intermediate nodes) as starting estimates of shortest path lengths. It then calculates the shortest paths under the constraint that only nodes 1 and 2 can be used, and so forth.

To state the algorithm more precisely, let D_{ij}^{n} be the shortest path length from node *i* to *j* with the constraint that only nodes 1, 2. . . . , *n* can be used as intermediate nodes on paths. The algorithm then is as follows:

Initially,

$$D_{ij}^{0} = d_{ij}, \qquad \text{for all } i, j, \qquad i \neq j$$

For *n* = 0, 1, . . . , N-1,

$$D_{ij}^{n+1} = \min\left[D_{ij}^{n}, D_{i(n+1)}^{n} + D_{(n+1)j}\right] \qquad \text{for all } i \neq j \tag{2.2}$$

To see why this works, induction is used. For *n* = 0, the initialisation gives the shortest path lengths subject to the constraint of no intermediate nodes on paths. Now, suppose that for a given *n*, D_{ij}^{n} in the algorithm above gives the shortest path lengths using nodes 1 to *n* as intermediate nodes. Then the shortest path length from *i* to *j*, allowing nodes 1 to *n*+1 as intermediate nodes, either contains node *n*+1 on the shortest path, or does not contain node *n*+1. For the first case the constrained shortest path from *i* to *j* goes first from *i* to *n*+1 and then from *n*+1 to *j*, giving the length in the final term of equation 2.2. For the second case, the constrained shortest path is the same as the one using nodes 1 to *n* as intermediate nodes, yielding the length of the first term in the minimisation of equation 2.2.

2.6.3 Dijkstra Algorithm

A detailed explanation of the operation of the Dijkstra algorithm and its comparative performance is once again found in [Beg02]. The algorithm requires that all arcs are non-negative (which will always be the case for wireless network applications). The general idea is to find the shortest path in order of increasing path length. Nodes are interconnected via a series of arcs such that the shortest of the shortest path to node 1 must be the single-arc path from the closest neighbour of node 1. Any multiple-arc path cannot be shorter than the first arc length because of the non-negative length assumption. The next shortest of the shortest paths must either be the single-arc path from the next closest neighbour of 1 or the shortest two-arc path through the previously chosen node (i.e. the closest neighbour), and so on. To formalise this procedure into an algorithm, we view each node i as being labelled with an estimate D_i of the shortest path length to node 1. When the estimate becomes certain (i.e. a shorter path cannot be found), we regard the node as being permanently labelled and keep track of this with a set P of permanently labelled nodes. The node added to P at each step will be the closest to node 1 out of those that are not yet in P. Figure 2-16 illustrates this concept.

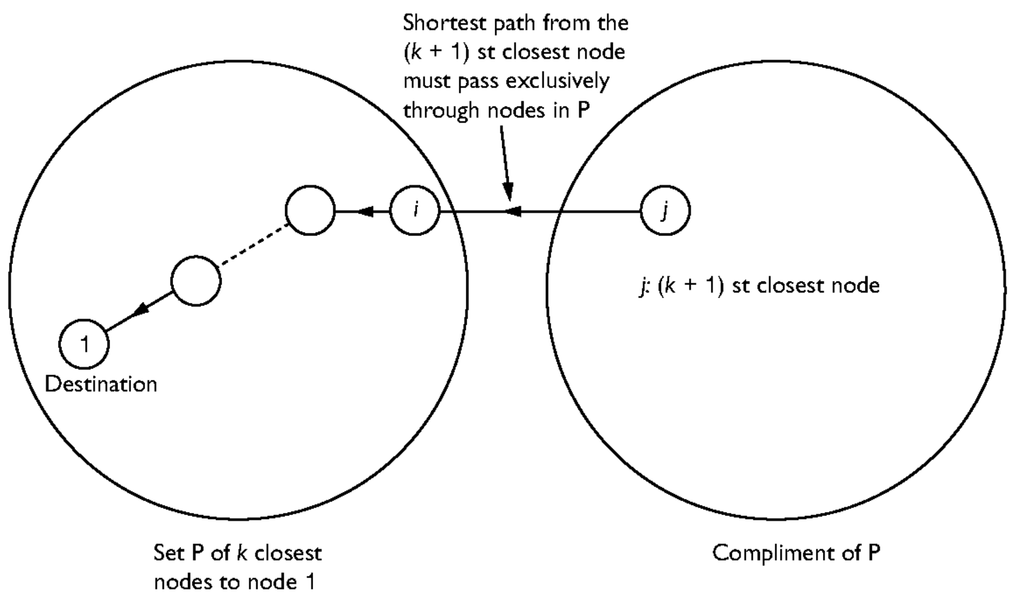


Figure 2-16: Basic idea of Dijkstra's algorithm [BeG02].

The figure above illustrates the basic idea of Dijkstra's algorithm with the set P of the k closest nodes to node 1 as well as the shortest distance D_i from each node i in the set P to node 1. Of all paths connecting some node not in P with node 1, there is a shortest one that passes exclusively through nodes in P (since $d_{ij} \geq 0$). Therefore the $(k+1)^{st}$ closest node and the corresponding shortest distance are obtained by minimising over $j \notin P$ the quantity $\min_{i \in P}\{d_{ji} + D_i\}$.

Dijkstra's algorithm can be formalised as follows, with the initial conditions;

$P = \{1\}$, $D_1 = 0$, and D_{j1} = for $j \neq 1$.

Step 1: Find the next closest node.

Find $i \notin P$ such that

$$D_i = \min_{j \notin P} D_j$$

Set $P := P \cup \{i\}$. If P contains all nodes then stop; the algorithm is complete.

Step 2: Update labels

For all $j \notin P$ set

$$D_j := \min\left[D_j, d_{ji} + D_i\right]$$

Go back to step 1.

To see why the algorithm works, interpret the estimates D_j. At the beginning of each step 1:

(a) $D_i \leq D_j$ for all $i \in P$ and $j \notin P$.

(b) D_j is, for each node j, the shortest distance from j to 1 using paths with all nodes except possibly j belonging to the set P.

Indeed, condition (a) is satisfied initially, and since $d_{ji} \geq 0$ and $D_i = \min_{j \notin P} D_j$, it is preserved by the formula $D_j := \min\left[D_j, d_{ji} + D_i\right]$ for all $j \notin P$, in step 2. Condition (b) is shown by induction. It holds initially. Suppose that it holds at the beginning of some step 1, let i be the node added to P at that step, and let D_k be the label of each node k at the beginning of that step. Then condition (b) holds for $j = i$ by the induction hypothesis. It is also seen to hold for all $j \in P$, in view of condition (a) and the induction hypothesis. Finally, for node $j \notin P \cup \{i\}$, consider a path from j to 1 which is shortest among those with all nodes except j belonging to the set $P \cup \{i\}$, and let $D_j^{'}$ be the corresponding shortest distance. Such a path must consist of an arc (j, k) for some $k \in P \cup \{i\}$, followed by a shortest path from k to 1 with nodes in $P \cup \{i\}$. The length of the path k to 1 is D_k, which then gives,

$$D_j^{'} = \min_{k \in P \cup \{i\}}\left[d_{jk} + D_k\right] = \min\left[\min_{k \in P}\left[d_{jk} + D_k\right], d_{jk} + D_i\right]$$

Similarly, the induction hypothesis implies that $D_j = \min_{k \in P}[d_{jk} + D_k]$, obtaining $D_j^{'} = \min[D_j, d_{ji} + D_i]$. Thus in step 2, D_j is set to the shortest distance $D_j^{'}$ from j to 1 using paths with all nodes except j belonging to $P \cup \{i\}$. The induction proof of condition (b) complete is therefore complete.

Note that a new node is added to P with each iteration, so the algorithm terminates after N-1 iterations, with P containing all nodes. By condition (b), D_j is then equal to the shortest distance from j to 1.

To estimate the computation required by Dijkstra's algorithm, note that there are N-1 iterations and the number of operations per iteration is proportional to N. Therefore, in the worst case the computation is $O(N^2)$, comparing favourably with the worst case estimate $O(N^3)$ of the Bellman-Ford algorithm. In fact, with proper implementation the worst case computational requirements for Dijkstra's algorithm can be reduced considerably [BeG02].

Dijkstra's algorithm is best described with an example. An illustrated example of the algorithm can be seen [Wal07] and is summarised as follows. The operation is shown in Figure 2-17, while the pseudo code for the algorithm is described below. For the following graph,

G = (V,E) where *V* is a set of vertices

E is a set of edges

Dijkstra's algorithm keeps two sets of vertices; *S,* set of vertices whose shortest paths from the source have already been determined and *Q,* the remaining vertices (undetermined).

Data structures needed:

d, array of best estimates of shortest path to each vertex.

p, an array of predecessors for each vertex.

With *a* as the source vertex, the pseudo code can be simplified as follows;

initialise *d* to infinity, *p*, *S* and *Q* to empty
$d = (\infty)$
$p = ()$
$S = Q = ()$

add *a* to *Q*
$d(a) = 0$

while *Q* is not empty
{
    u = extract-minimum(*Q*)
    add *u* to *S*
    relax-neighbours(*u*)
}

Listing 2.1: Pseudo code for basic idea of Dijkstra's algorithm.

Start at the source vertex, *a*

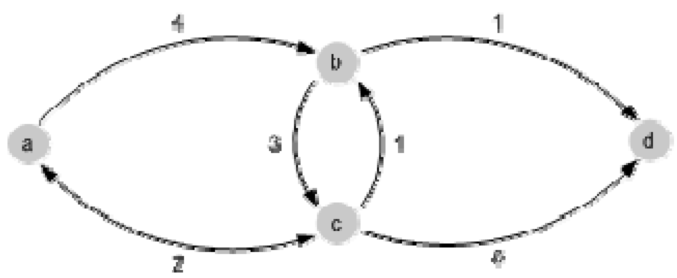


Figure 2-17 (a): Start with the source node *a*. Arc distances are all indicated [Wal07].

Start off by adding the source vertex *a* to the set *Q*. The set *Q* is not empty so extract its minimum, in this case *a* again. Add *a* to the set *S*, then relax its neighbours.

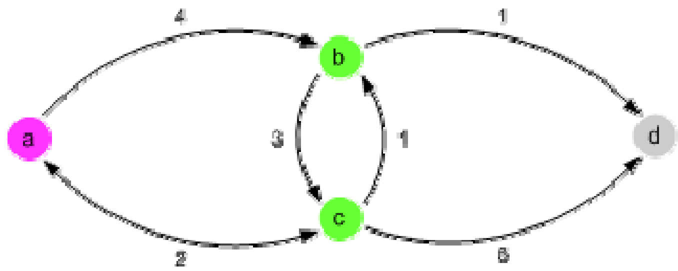


Figure 2-17 (b): Estimate distance to neighbours *b* and *c* [Wal07].

Node *a* has neighbours *b* and *c* to which a best distance is estimated. Taking node *b* first, determine the best distance from node *a* to *b* and set its predecessor

$d(b) = d(a) + [a, b] = 0 + 4 = 4$

$p(b) = a$

Similarly for node *c*, $d(c) = 2$, $p(c) = a$.

On the next pass *Q* will contain both *b* and *c* with *c* having the shortest distance to *a*. This shortest distance is removed from *Q* and placed in the list of sorted nodes *S*. The neighbours of node *c* are then relaxed, which are nodes *b*, *d* and *a*.

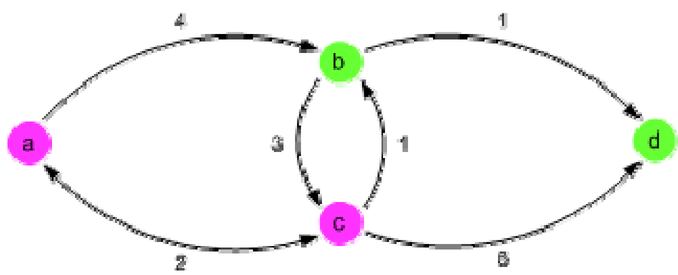


Figure 2-17 (c): Node *c* has shortest distance to source node *a* [Wal07].

Node *a* is not considered as it is in the sorted list *S*. The distance to nodes *b* and *d* will have to pass through node *c*. Therefore the distance to node *b* will now be;

$d(c) + [c, b] = 2 + 1 = 3 < d(b)$

A shorter path has been found to node *b* so update its distance;

$d(b) = 3$, $p(b) = c$ and add *b* again to *Q*.

The path for *d* has a distance of 7 with node *d* predecessor set to *c*. Node *d* is added to *Q*. Next extract node *b* from Q, as it has the shortest distance and add it to *S*.

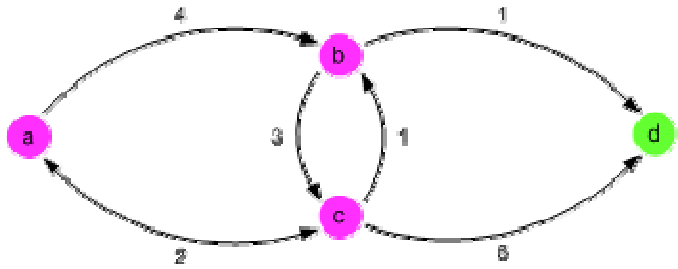


Figure 2-17 (d): Node *b* has a shorter path to the source through node *c* [Wal07].

Run another pass and find the neighbours of node *b* to be nodes *a*, *c* and *d*. Again, nodes *a* and *c* are not considered as they are in the sorted set *S*. The distance from node *d* through node *b* to *a* is now calculated as 4 < 7. The distance to *d* is updated as its predecessor. Node *d* is then added to *Q*.

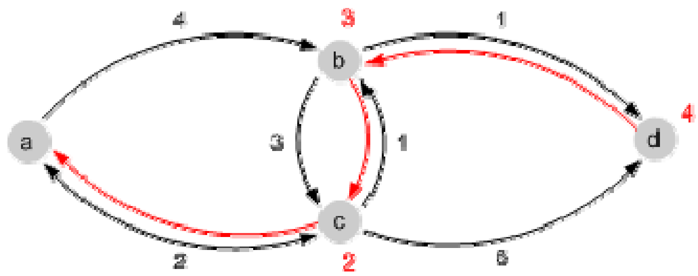


Figure 2-17 (e): All nodes have paths set t source node *a* [Wal07].

At this point the only node left in the unsorted set *Q* is *d*. Therefore, the process is complete and the shortest paths back to the source node *a,* is complete for all nodes.

2.7 Network Simulation

There are three main approaches generally adopted for the performance evaluation of networking protocols: analytical analysis, experimentation and simulation. Due to the high complexity of wireless communications, analytical analyses are often based on unrealistic assumptions (e.g., node synchronisation, ideal MAC layer, homogeneous location model, symmetric radio links, etc.) and inaccurate physical layer (PHY) models [INR09]. An example of inaccurate PHY modelling is the disk model which has been widely used to model the radio range of wireless nodes where the interference is generally not taken into account [RoE09]. In addition to this, theoretical analysis tends to focus on a given layer, ignoring or omitting the other network layers. The experimentation approach can provide valuable insight into the behaviour of protocols in wireless environments. However, setting up large scale test-beds is a tedious task and is not always feasible even more so in the case of large scale WMNs. Add to this the fact that the obtained results are strongly correlated to the surrounding environment and are difficult to reproduce [HCG08].

For these reasons, the use of simulations is generally considered to be the most convenient methodology to analyse the performance of protocols and distributed applications. According to [Sto08] the main contribution of simulation will then be to demonstrate a novel protocol, novel concepts, or theoretical analysis. Simulation should be used to provide minimal but necessary support for the claims made, provide justification for subsequent deeper simulation, emulation, or implementation, and/or indicate weaknesses to be addressed by further analysis and research.

Nonetheless, [INR09] states that the complexity of the physical phenomena constituting the radio medium introduces a trade-off between accuracy and computational cost in wireless network simulation. Selecting the correct level of detail (or level of abstraction) for a simulation is a difficult problem. The validity of the simulation results is deeply influenced by the amount of detail involved in the representation of the simulated system [PNY03]. Too little detail can produce simulations that are misleading or incorrect. In a performance evaluation, an inadequate amount of details in the model representation can lead to misleading or wrong results [CSS02]. However, the disadvantage to this is that adding detail requires time to implement, debug, and later change, it slows down simulation, and it can distract from the research problem at hand [HBE01]. According to [BrD07] an increased amount of detail in the simulated model translates into many factors:

- More computation is required to evolve the simulation.
- More memory is necessary to represent the modelled system.
- An increased amount of communication between the simulated entities.

The practical effect is a more complex simulation that requires more time to complete each run. Designing simulations to study a protocol inherently involves choosing which protocol details to implement and as such presents a trade-off between accuracy and computational cost, as discussed in [HBE01].

2.7.1 Off-The-Shelf Simulation Tools

A number of network simulation tools are available either commercially or under the GNU General Public License. The aim of these tools is to provide an advanced and complete simulation environment to investigate and evaluate networking protocols and wireless systems. Examples are;

NS-2:

The Network Simulator, NS [BEF00] also referred to as NS-2 because of its second generation, is a discrete event simulator targeted at networking research. NS provides support for simulation of TCP, routing, and multicast protocols over wired and wireless, including local and satellite, networks. The simulator is currently at version 2.34 with version 3 under development [NS210].

Network simulation has a very long history. NS itself is derived from REAL (Real and Large) [KeS97], which itself is derived from NEST (Network Simulator Testbed) [DSY90]. The NS-2 network simulator is one of the most widely used environments for wired and wireless network simulations. NS is basically a transport-level simulator that supports several variants of TCP (including SACK, Tahoe and Reno) and router scheduling algorithms. The simulator is developed in C++ while simulation models can be described using a variation of the Tool Command Language, TCL. The Virtual Inter-Network Testbed (VINT) project with its principal tool NS, aims to develop a comprehensive simulator for the Internet. However, the simulator is known to suffer from a limited scalability [NaG03] though some recent enhancements have been put forward to support simulations of a few thousand nodes

[NaG03]. Other issues noted are the use of a flat Earth model [KNE03] and it uses serial execution as opposed to parallel execution.

GloMoSim:

The Global Mobile Information Systems Simulator, GloMoSim [BTA99] is a scalable simulation environment for wireless and wired network systems. It employs the parallel discrete-event simulation capability provided by Parsec [BMT98]. GloMoSim currently supports protocols for a purely wireless network. In the future, the developers anticipate adding functionality to simulate a wired as well as a hybrid network with both wired and wireless capabilities. GloMoSim source and binary code can be downloaded only by academic institutions for research purposes. Commercial users must use QualNet, the commercial version of GloMoSim. An overview of the GloMoSim support features is given in Table 2-3.

Layers	**Protocols**
Mobility	Random waypoint, Random drunken, Trace based
Radio Propagation	Two ray and Free space
Radio Model	Noise Accumulating
Packet Reception Models	SNR bounded, BER based with BPSK/QPSK modulation
Data Link (MAC)	CSMA, IEEE 802.11 and MACA
Network (Routing)	IP with AODV, Bellman-Ford, DSR, Fisheye, LAR scheme 1, ODMRP, WRP
Transport	TCP and UDP
Application	CBR, FTP, HTTP and Telnet

Table 2-3: Overview of GloMoSim support features [GMS10]

JiST/SWANS:

Java in Simulation Time, JiST is described as a high-performance discrete event simulation engine that runs over a standard Java virtual machine.

Scalable Wireless Ad hoc Network Simulator, SWANS is a scalable wireless network simulator built on top of the JiST platform. SWANS is organised as independent software components that can be composed to form a complete wireless network or sensor network configurations. Its capabilities are similar to NS-2 and GloMoSim, however it is capable of simulating much larger networks. By taking advantage of the JiST design, SWANS is able to achieve high simulation throughput. In doing this the developers claim that users can save memory and run standard Java network applications over simulated networks. In addition, SWANS implements a data structure, called hierarchical binning, for efficient computation of signal propagation. It can be shown that JiST/SWANS outperforms NS-2 and GloMoSim in terms of scalability and memory usage. Table 2-4 below outlines the memory footprint of JiST when compared to GloMoSim and NS-2 for various event types.

	Memory Footprint		
	per entity	**per event**	**10K Node Sim**
JiST	36B	36B	21MB
GloMoSim	36B	64B	35MB
NS-2	544B	40B	74MB

Table 2-4: JiST space benchmark comparison [JiS10].

GTSNetS:

GTSNetS (The Georgia Tech Sensor Network Simulator) is an extension to the GTNetS project. GTSNetS is built on top of GTNetS as a simulation framework for sensor networks. As GTNetS is written entirely in C++ language using an object-oriented methodology, the developers claim to be able to take full advantage of the existing functionalities of GTNetS with moderate effort. By leveraging much of the existing capabilities of GTNetS, the developers state that it is capable of implementing GTSNetS in a modular and efficient manner. This allows GTSNetS to simulate large-scale wireless sensor networks. The developers state that their design technique eliminates the need to impose architectural or design decisions on the user who wants to simulate a particular sensor network [VRH05]. The main features of GTSNetS as outlined in [VRH05].

- A unifying framework of existing energy models for the different components of a sensor network.
- Providing two models of accuracy of sensed data and the ability to add new models helps in the understanding of the trade-off of quality versus lifetime.
- GTSNetS allows users to choose among established implementations of network protocols, applications, sensors, energy and accuracy models. Additional diversity is achieved by the ability to add new models.
- The simulator has excellent scalability features specifically designed for sensor networks with the ability to simulate networks of up to several hundred thousand nodes.
- Can be used to collect detailed statistics about a specific sensor network at the functional unit level, the node level as well as at the network level.

OMNeT++:

The Open-architecture Modular Network, OMNeT++ is a component-based, simulation environment with strong GUI support and an embeddable simulation kernel. The simulator can be used for modelling: communication protocols, computer networks and traffic modelling, multi-processors and distributed systems. OMNeT++ also supports animation and interactive execution. It is freely distributed under an academic public license. Its primary application area is the simulation of communication networks, but because of its generic and flexible architecture, is successfully used in other areas such as the simulation of complex IT systems, queuing networks and hardware architectures.

The simulator uses C++ modules before assembling into larger components and models using a high-level language. OMNeT++ boasts having an extensive GUI support, and due to its modular architecture, the simulation kernel (and models) can be embedded easily into users' applications. OMNeT++ IDE is based on the Eclipse platform and runs on almost all modern operating systems.

OPNET:

The OPNET Modeler [OPM10] is just one of the many tools from the OPNET Technologies suite. At the core of the OPNET Modeler is a finite state machine model in combination with an analytical model. OPNET Modeler can model protocols, devices and behaviours with a vast collection of special-purpose modelling functions. Similar to OMNeT, it makes use of a GUI and is supported by a considerable amount of documentation and study cases. A number of editors are provided to simplify the different levels of modelling that the network operator requires. Although OPNET Modeler is not open source software, model parameters can be altered which can have a significant effect on the simulation accuracy. OPNET Modeler has significantly large software overhead but provides diverse statistics modules at different levels. The need to enter into an OPNET Modeler license agreement and associated costs for additional modules can be seen to deter users [LFJ03].

OPNET Modeler is considered a high-end product used mostly by network R&D engineers. It can very precisely model protocols, devices, and behaviours using a finite-state machine paradigm, C/C++ language features, and about 400 special-purpose modelling functions. OPNET Modeler has optional add-on modules for radio and satellite modelling, multivendor import, and service-level prediction.

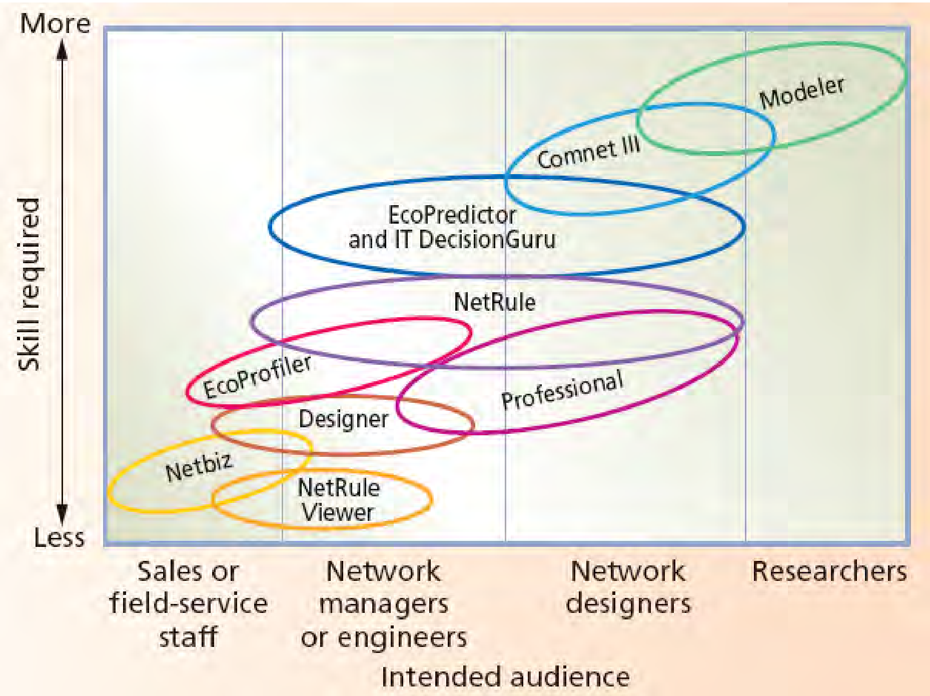


Figure 2-18: Skill requirement for various network tools [BrA00].

Originally a discrete-event simulator, OPNET Modeler now supports hybrid simulations, which combine discrete-event simulation and analytical modelling. It can also run a simulation in parallel over several CPUs. Both hybrid and parallel simulations can significantly reduce simulation runtimes [BrA00]. Figure 2-18 illustrates the intended audience and skill requirement for OPNET Modeler amongst various other network tools.

Parsec:

The Parallel Simulation Environment for Complex Systems, Parsec [UPC10] is a C-based simulation language, developed by the Parallel Computing Laboratory at UCLA, for sequential and parallel execution of discrete-event simulation models. It can also be used as a parallel programming language. It is available in binary form only for academic institutions.

Parsec is based on the Maisie simulation language and claims to use significant modifications such as [BMT98]:

- It uses simpler syntax.
- It uses modified language to facilitate porting code from the simulation model to the operational software.
- It has a robust and extensible runtime kernel that is considerably more efficient than its predecessor.
- It uses new protocols to predict parallel performance.

SSF:

The Scalable Simulation Framework, SSF [CNO99] provides a maximally compact interface for building discrete-event simulations. The simulator is developed using five core classes totalling a few dozen methods altogether. The five core classes are described as follows;

- *Entity* is the base class for all simulation components; it serves primarily as a container mechanism for defining alignment relations among a model's pieces. Entities that the modeler has co-aligned will presumably interact at close quarters through event exchange on channels with low or zero intrinsic minimal delay. The underlying simulator will take this assumption into account when mapping entities to processors.
- *Event* is the base class for the quantum of information exchange.
- *InChannel* and *OutChannel* are communication endpoints for event exchange; each instance of InChannel and OutChannel belongs to a specific Entity. SSF supports many-to-one communication as well as one-to-many and many-to-many. Each OutChannel will have an intrinsic minimal transmission delay (ascribable, for example, to device latencies or transmission delay on a simulated link) associated with it, which is automatically added to the per-write delays of individual events sent on it.
- *Process* is the base class for describing Entity behaviour. Each instance of Process is normally associated with a specific Entity; it might wait for input to arrive on the channels of that Entity, wait for some amount of simulation time to elapse, or do both in turn. The simplest Process waits for an event to arrive on a channel, responds to it, and then goes back to sleep. The binding of Process to Entity is not tight; a Process might wait on channels or access methods of all Entities that are co-aligned with the Process' nominal owner.

Commercial	Academic	Platform
	NS-2	Unix, Mac OS X, Windows via Cygwin
Qualnet	GloMoSim	Unix, Windows.
	JiST/SWANS	Linux, Windows
	GTSNetS	Linux, OSX, Solaris, Windows (Beta)
OMNEST	OMNeT++	Linux, Windows
OPNET	(Discount available)	Windows
	Parsec	Unix, Linux, Windows
	SSF	Solaris, Linux, Windows

Table 2-5: Comparison of network simulation tools.

2.7.2 Custom Simulation

Even with this array of sophisticated simulation tools, none of them can remove the burden from the designer on deciding which protocols to implement. In many cases such simulation tools are unnecessarily sophisticated and can be replaced with equally valid abstract models [Sto08]. In custom simulators, researchers typically include only the minimum possible details outside the immediate area of study. Existing simulators, as outlined previously, provide detailed protocol implementations. In [HBE01] the authors query what level of detail is required in developing new protocols, or in adapting existing protocols to model new hardware? The authors note that some simulators ease the cost of changing abstraction with multiple, selectable levels of detail.

Even with the large variety of network simulators available, the complexity of the wireless physical layer (PHY) leads to implementation choices during the simulators design. As a result of this, the PHY simulation accuracy varies dramatically from one simulator to another. In particular, interference management is probably the area where current simulators differ most significantly. The reason that a low accuracy is justified is generally due to performance [HBE01]. Further to this, it is not unusual for protocol and environment parameters to be tuned arbitrarily during the performance evaluation of high level protocols. As a result of this, it is difficult to achieve a representative and overall performance evaluation of protocols using such simulation design techniques. In fact, several previous publications ([CNO99], [BWG99], [FlJ95]), have shown that when comparing different simulation environments, the behaviour of certain wireless networking protocols can differ radically between environments. This tends to support the case for the development of custom network simulators using simple abstract models as suggested in [Sto08]. In the next chapter we will provide a comparative performance study of recent work in modelling and design of network simulation.

2.8 Optimisation Techniques

Algorithms for optimisation problems typically go through a sequence of steps, with a set of choices at each step. For many optimisation problems, using dynamic programming to determine the best choices is excessive; simpler, more efficient algorithms will usually suffice. A *greedy algorithm* always makes the choice that looks best at the moment. A key ingredient is the *greedy-choice property*: a globally optimal solution can be arrived at by making a locally optimal (greedy) choice. In other words, when we are considering which choice to make, we make the choice that looks best in the current problem, without considering results from sub-problems. Such a strategy is not generally guaranteed to find globally optimal solutions to problems. For the minimum-spanning-tree problems, however, it can be shown that certain greedy strategies do in fact yield globally optimum solutions [CLR02]. The following subsections describe the main optimisation techniques considered in this thesis. Ultimately an approach based on simulated annealing was used however, local and tabu search methods were also explored. An excellent review and description of optimisation techniques can be found in [Cop04] some of which are summarised in the following subsections.

2.8.1 Local Search

Local search is a metaheuristic which moves through a problem space by making small changes to an initial configuration in order to find a better solution. Local search techniques such as Hill Climbing are prone to finding local maxima that are not the best possible solution. It is possible to overcome this problem by repeating the optimisation procedure from different initial states. Sufficient iterations will improve the probability of finding a global maxima.

The trade-off with this type of method is between proving a satisfactory solution and running an exhaustive search of the entire problem space. An iterative search technique is often used to solve the travelling salesman problem where the search space grows extremely quickly as the number of cities increases. In certain cases where an optimal solution would be difficult to guarantee a few iterations would suffice in order to find a local maxima. Even one iteration of local search can happen upon the global maximum (albeit with a low probability).

2.8.2 Tabu Search

Tabu Search is a metaheuristic that uses a short term memory list of states that have already been visited to attempt to avoid repeating paths. The tabu search metaheuristic is used in conjunction with another heuristic in order to avoid local maxima. The tabu list keeps track of previously searched states and operates by searching unexplored paths that appear to be poor thus avoiding previously searched paths so that a better path can be found.

2.8.3 Simulated Annealing

Simulated annealing is a local search metaheuristic for solving global optimisation problems in a large problem space. The method is adopted from annealing in metallurgy and is an extension of a process called metropolis Monte-Carlo simulation. Simulated annealing is applied to a multi-value combinatorial problem where the aim is to minimise a particular value which is dependent on many variables. The minimising value is often referred to as the energy of the system and the operation can easily be transformed to find the maximum energy. Simple Monte Carlo simulation involves randomly selecting points within a search space in order to learn information about that search space. Metropolis Monte Carlo simulation adapts this method by making changes to the current state rather than choosing new search states at random from the search space. A new state is accepted if it yields a lower energy than the previous state. If the energy is higher than the previous state then a probability is applied to determine if the new state is accepted. This operation avoids local maxima by searching paths with a higher energy in anticipation of discovering a lower energy state. This probability is called a Boltzmann acceptance criterion and is calculated as follows:

$$e^{(-dE/T)}$$

Where T is the current temperature of the system, and dE is the increase in energy that has been produced by moving from the previous state to the new state. It should be noted that some systems use $e^{(-dE/kT)}$ where k is Boltzmann's constant. The temperature is used to determine the number of steps accepted that will lead to a rise in energy. A greater number of steps will be accepted at a high temperature than at a low temperature. This is known as the cooling schedule (or annealing schedule) and it determines the manner in which the temperature is lowered. Two popular cooling schedules are as follows:

$$T_{new} = T_{old} - dT$$

$$T_{new} = C \times T_{old} \qquad \text{where } C < 1.0$$

The simulated annealing process determines whether or not to move to a higher energy state by calculating the probability $e^{(-dE/T)}$ and comparing it to a random number between 0 and 1. If this random number is lower than the probability function, the new state is accepted. The probability $e^{(-dE/T)}$ is minimised when the increase in energy is high or the temperature is low resulting in less states being accepted.

2.9 Channel Model.

Thorough details of radio propagation and their impact on wireless networks can be found in [WMB06]. In the following sections a summary is given of the main factors which were considered when developing the wireless mesh simulator.

When developing simulation models using wireless networks it is essential to be familiar with radio propagation characteristics. Wireless networks differ from wired networks at the PHY layer by utilising electromagnetic signals transmitted in free space in order to communicate. Figure 2-19 illustrates the basic operation of a radio propagation path. For the purpose of design and simulation, stochastic models were developed which attempt to accurately describe the physical effects of the underlying environment. The network designer chooses the model based on how the system will be utilised. Factors influencing the choice of model include the frequency and range of the radio waves, the characteristics of the propagation medium and the antenna arrangement. For the purpose of our simulator a Free Space Loss (FSL) model combined with a path loss coefficient is used. However, more complex models exist which incorporate factors such as fading, shadowing and multi-path propagation. Examples of such sophisticated models are in described in [WMB06] and include;

- One Slope model (path loss determined by logarithmic distance)

- Hata-Okumura model (frequency dependent improvements on the One Slope model)
- Walfish-Ikegami model (propagation above roof tops)
- Berg Model (outdoor path loss along streets)

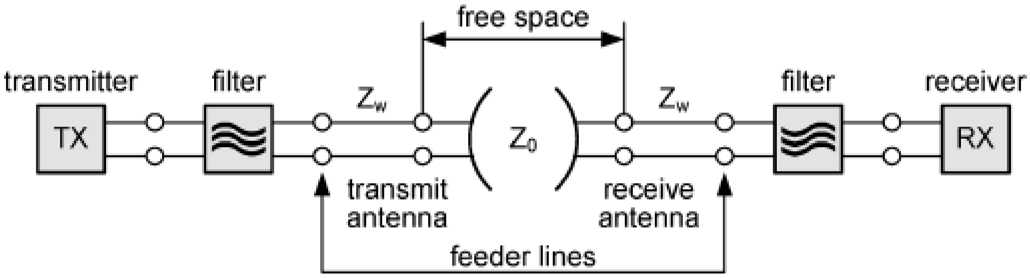


Figure 2-19: Radio transmission path [WMB06].

2.9.1 Free Space Propagation

A factor that influences the range of electromagnetic waves is the strength of the transmit power. The received power is inversely proportional to the distance and can be shown to decrease with the square of the distance. Consider an isotropic radiated signal of wavelength wavelength λ which transmits its power uniformly in all directions. For a transmit power of P_T and a gain of G_T and G_R (transmitter and receiver respectively), the received power P_T is given as;

$$P_R = P_T G_T G_R \left(\frac{\lambda}{4\pi d} \right)^2 \qquad (2.3)$$

The free space pathloss is described as the spatial diffusion of transmitted energy over the path of length d and is given by the term $(\lambda / 4\pi d)^2$.

Expressed using logarithmic representation, in the case of an isotropic antenna, the free-space loss L_F reduces to

$$L_F = -20\log\left(\frac{\lambda}{4\pi d}\right) = -20\log\left(\frac{P_R}{P_T}\right) \qquad (2.4)$$

2.9.2 Path loss Coefficient

Free-space propagation is considered to be unrealistic in mobile communications due to obstacles and reflective surfaces which will appear in the propagation path. In addition to attenuation caused by distance, propagated waves will also lose energy through reflection, transmission and diffraction due to such obstacles. A basic example of this is the two path propagation loss through reflection as illustrated in Figure 2-20 below.

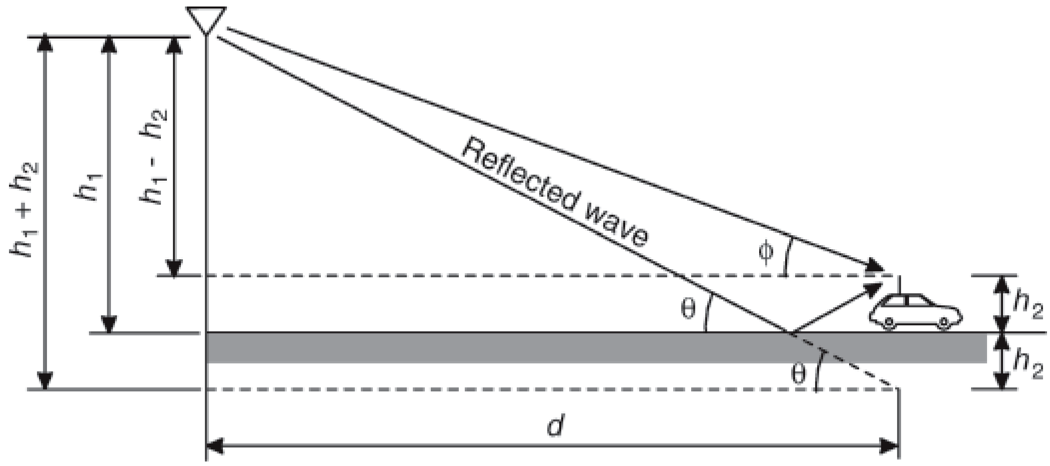


Figure 2-20: Two path propagation through reflection [WMB06].

Although the two path propagation model depicts a mobile radio environment more closely to reality it fails to take into account reflective properties of the ground surfaces. A rough ground surface will cause wave scattering in as well as reflection. In addition to this, the type of obstacles in the propagation path and their reflective properties need to be considered as they will cause additional attenuation.

In order to build a more realistic model which takes attenuation into account a path loss coefficient γ is introduced. So that, for an isotropic antenna,

$$P_R = P_T G_T G_R \left(\frac{\lambda}{4\pi} \right)^2 \frac{1}{d^{\gamma}} \tag{2.5}$$

Realistic values for γ are between 2 (free-space propagation) and 5.5 (strong attenuation, e.g., due to city buildings) [WMB06].

2.10 Chapter Summary

In this chapter we introduce the general concepts and architecture considerations concerning WMNs. As our research is concerned with improving multicast communications we present specific technical details on multicasting over WMNs. Furthermore we describe the multicast advantage whereby each node in communications range of a transmitter is capable of receiving the same data by way of a single source transmission. This is an inherent feature of all wireless networks, however, multicasting in wireless networks seeks to takes full advantage of this feature.

We discuss methods of creating spanning trees by solving shortest path trees (SPTs) in mesh network. We provide details of three similar techniques frequently used, namely Bellman-Ford, Floyd-Warshall and Dijkstras algorithm. We will present details in the next chapter how shortest path trees provide optimal performance in multicast WMNs. By analysing the construction of the multicast trees we aim to identify key areas which can benefit from enhancements to the topology (either by physically placing relay nodes or by tuning the power of each forwarding node).

In order to analyse the construction of the multicast trees we use network simulation tools. There are currently a large variety of open source and commercially “off-the-shelf” simulation tools available to the researcher. Alternatively custom simulation tools can be developed to tailor the exact needs of the researcher. Off-the-shelf simulation tools have the obvious advantage of being readily available and are generally well supported. However, the level of detail included in such simulators is often unnecessary and can actually provide misleading results. We will discuss this in more detail in the next chapter. As part of our simulations we will generate hundreds of thousands of simulated topologies. We will apply our algorithm to improve the performance of the multicast tree formed using these topologies. As we are effectively altering the topology we will use an optimisation technique to search for an optimal multicast tree. Therefore, we present in this chapter details of well known search optimisation techniques for comparison.

Finally, the last section of this chapter presents a brief overview of the channel model assumed in our simulation. Channel modelling in WMNs can be extremely complex. For simplicity of simulation we assume a free space loss (FSL) model with a fixed path loss coefficient. We will explain in chapter 4 the operation of our simulation model in detail. We will show how our algorithm to adapt the transmit power operates on a per node basis. In reality each node will experience different levels of interference due to a number of external sources. We use the FSL model simply to demonstrate the operation of the algorithm.

3 Literature Review

In the following sections we will discuss some of the issues being addressed regarding WMN research. In particular we will attempt to explain what research is being carried out in the development of multicasting over mesh. As will be seen, current approaches fail to take into account the specific problems which arise when multicasting is employed. The majority of current research attempts to adapt existing technologies taken from wired networks or from WLANs. Other studies concentrate on unicast traffic without considering the specific requirements for broadcast/multicast operation. However, this does not always translate well to WMNs. The following topics are reviewed based on current practices and are followed by a critical discussion;

- Routing Metrics
- Routing Protocols
- Rate Control
- Simulation Review
- Topology Optimisation Techniques
- Capacity of Wireless Mesh Networks
- Wireless MAC Anomaly
- Network Coding

3.1 Routing Metrics

For a routing algorithm to select better paths in a network it is necessary to explicitly take into account the quality of the wireless links between nodes. This is achieved by gathering information about the link and using it in such a way so as to minimise the link cost between nodes. There has been a number of different link quality performance metrics put forward with new ones being developed all the time. To gain a better understanding of link quality metrics we will first take a look at some of the more frequently used ones.

Possibly the simplest routing metric to implement is the minimum hop count. The idea is quite basic in that link quality is determined to be true or false depending on whether a link exists or not. Once the network topology is known it is then a matter of calculating the minimum number of hops between a source and destination node. The main advantage of this metric is its simplicity. No additional information is required after the hop count, therefore minimising any overhead. The simple nature of minimum hop count also happens to be the main disadvantage. The routing metric does not consider other factors such as lossy links, available bandwidth or congested nodes. In wireless data networks the transmission data-rate is a function of the received signal strength which is a function of the distance between the two wireless nodes (e.g. between an access point and mobile terminal or between two mesh nodes) [MBP06]. This type of metric leads to the discovery of slow links as it seeks to maximise the distance between hops, in order to minimise the hop count. This can be seen in [DAB03] and will be shown later in our analysis.

DeCouto et al [DAB03] introduced the Expected Transmission Count (ETX) metric which claims to find high-throughput paths on multi-hop wireless networks. ETX achieves this by minimising the expected total number of packet transmissions (including retransmissions) required to successfully deliver a packet to the final destination. The metric predicts the number of retransmissions required, using per-link measurements of packet loss ratios in both directions of each wireless link. The ETX metric incorporates the effects of link loss ratios, asymmetry in the loss ratios between the two directions of each link, and interference among the successive links of a path. The authors' primary goal for ETX was to design a metric capable of finding paths with high throughput, despite losses.

The ETX is calculated using the forward and reverse delivery ratios, d_f and d_r respectively (i.e. the probability that a packet successfully arrives and the probability that an acknowledgement (ACK) is successively received). The ETX is defined as follows;

$$ETX = \frac{1}{d_f \times d_r} \tag{3.1}$$

The expected probability that a packet is successfully received and acknowledged is given by d_f x d_r.

In [DPZ04a], Draves et al put forward a metric for use in multi-channel WMNs. The metric is based on the Expected Transmission Time (ETT) which the authors define as a bandwidth adjusted ETX. The ETT extends ETX not only by predicting the amount of time required for a packet to successfully traverse a route, but also by observing the highest usable bit rate of each link and the probability of successful delivery at that rate. If S and B denote the packet size and the link data rate respectively, then;

$$ETT = ETX \times \frac{S}{B} \tag{3.2}$$

ETT uses periodic broadcast packets of two sizes. Small packet sizes of 60 bytes are always transmitted at 1Mbps and correspond to ACKs. Large packet sizes of 1500 bytes are broadcast at various rates and correspond to data. This means that when using 802.11b large packets will be broadcast at 4 different rates (1, 2, 5.5, and 11 Mbps) whereas using 802.11g will mean broadcasting at an additional 8 rates (6, 9, 12, 18, 24, 36, 48, and 54 Mbps). Statistics gathered at each node are based on these broadcasts. Nodes then share this information with neighbouring nodes. The routing protocol determines that the best route is the one with the lowest ETT.

It is worth noting that there are alternative versions of ETT which incorporate various modifications. One such version is described by Bicket et al in [BAB05]. The main difference being that ETT described in [DPZ04a] utilises unicast packets to directly measure the ETT between each pair of nodes whereas the version in [BAB05] uses broadcast probes to predict the transmission time. The trade off between the two is associated overhead. A good example of the trade-off between active probing and passive monitoring can be seen in [KKD07a].

In [DPZ04b], Draves et al present an empirical evaluation of the routing metrics ETX [DAB03], Per Hop Round Trip Time (RTT) [ABP04] and Per-Hop Packet Pair Delay (PktPair) (based on [Kes91]). The performance of each is compared to the well known Minimum Hop Count metric. The authors use the routing protocol DSR [JoM96] over an indoor 23 node, stationary ad-hoc wireless network. The results show that with stationary nodes the ETX metric significantly outperforms minimum hop count. The RTT performs poorly due to load-sensitive issues and hence suffers from what the authors refer to as self-interference. Although PktPair attempts to overcome queuing delay issues suffered by RTT (as reported in [ABP04]) it too is load sensitive which results in its poor performance. However, in a mobile scenario hop-count is seen to perform better because it reacts more quickly to fast topology change.

In [RKD06], Roy et al present a study of high throughput performance metrics in multicast WMNs. The authors highlight the differences in unicast and multicast transmissions and identify how existing unicast routing metrics should be adapted to take advantage of multicasting. A combination of a 50 node simulation and an 8 node wireless test-bed using a modified On Demand Multicast Routing Protocol (ODMRP) [LGC99] is used to validate their work. Their study gives a performance analysis of PktPair (PP), ETX, ETT, Multicasting ETX (METX) [DBA05] and Success Probability Protocol (SPP) [BaM02]. Each of which is adapted to take advantage of the unique features of multicasting over wireless mesh. Both METX and SPP are adapted from energy efficient protocols.

The authors showed how their enhancements to ODMRP, using the adapted link metrics, yield better results in both simulation and experimental work. The authors found that heavily penalising lossy links is an effective way to avoid low-throughput paths. SPP (14% throughput gain) and PP (17.5% throughput gain) achieved the highest throughput performance because of their aggressive manner of penalising lossy links. Moreover, the study also noted that SPP has much less overhead (i.e. less bandwidth consumed from transmitting probe packets) than PP, which reduces the end-to-end delay. This also led to the observation of the trade-off between throughput gains achieved and the probing overhead incurred, i.e. a higher probing rate gives more recent information about the network but also causes interference for data packets.

Extending upon this, Liu et al [LHZ08] provide a comprehensive study of 10 Quality of Service (QoS) routing metrics for WMNs. The authors group metrics into 3 classifications, namely; ETX and the metrics based on it (ETX, Weighted Cumulative ETT [DPZ04a], Metric of Interference and Channel switching [YWK05a] [YWK05b], Multi Channel Routing Protocol [KyV05] [KyV06] and Interference Aware routing metric [SBM06]); modified ETX (mETX) [KoB06]; and other metrics (RTT [ABP04], PktPair [Kes91] and Contention Node and Aggregated Traffic Bandwidth (CN and B_{agg}) [KiB06]).
ETX based metrics were found to perform poorly under short term channel variations. This is due to the mean loss ratio failing to reflect high burst loss conditions. Modified ETX (mETX, not to be mistaken with METX given in [DBA05]) overcomes the shortcomings of ETX by considering channel time-varying conditions. The authors suggest that all ETX based metrics should be based on mETX. The disadvantages of RTT and PktPair are highlighted as described above in [DPZ04b]. CN and B_{agg} differ slightly from previous metrics in that they are node contention based measurements. Their main disadvantage is that they only reflect the medium during the broadcast of HELLO messages.

3.1.1 Discussion

Routing protocols developed for use with unicast are implemented using a variety of different routing metrics. Many multicast routing protocols [GuM03, GuM04, JeJ01, JiC01, LGC99, LEH03, RoP99, SSB99, XTM02] have favoured the use of minimum hop count routing metrics and have focused on scenarios of high mobility. Multicast transmissions differ from unicast in that they should take advantage of the wireless multicast advantage [WNE00]. For this reason alone it is clear that unicast routing metrics are not best suited for multicast. Link based quality metrics such as those proposed above require broadcast packets to probe the network. This type of technique is bandwidth consuming and can be seen as unreliable [ABP04, DPZ04a, DPZ04b, KKD07a]. A passive technique is put forward in [KKD07b] which is load independent and can provide additional measures regarding the quality of the link without an associated measurement penalty. Such metrics can be adapted to suit wireless multicasting over mesh in order to take advantage of the wireless multicast advantage.

3.2 Routing Protocols

One of the basic elements of a WMN is that it utilises a routing protocol which provides redundancy. In order to achieve this, the routing protocol must select a path set out by the designer. For nodes to successfully communicate with each other they must gather information regarding the network topology. This is generally achieved either reactively or proactively.

Routing protocols designed on the basis of the routing information maintenance approach are proactive or table-driven routing protocols.

Examples of such routing schemes are reactive or on-demand routing protocols and hybrid routing protocols. In the case of proactive or table-driven routing approach, every node exchanges its routing information periodically and maintains a routing table, which contains routing information to reach every node in the network. Examples of routing protocols that use this design approach are DSDV [PeB94], WRP [MuG96], and STAR [GaS99].

Reactive methods have proven to be more successful for WMNs if such networks are highly dynamic and nodes are allowed to roam. Among the most commonly used reactive protocols are AODV [PeR99] and DSR [JoM96].

Ad hoc On-Demand Distance Vector (AODV) routing protocol allows nodes to obtain routes, only when necessary, by broadcasting query request packets. Its principal concern is to discover a route with the minimum number of hops. AODV attempts to reduce the overhead by minimising the number of messages. This is achieved by making use of route sequence numbers thus avoiding loops. It also features a mechanism dealing with broken links and minimising the number of requests sent.

Dynamic Source Routing (DSR), like AODV, is a protocol which operates on demand. This method minimises the overhead by reacting only when route discovery is necessary. Route discovery probe packets are used to determine the route from source to destination. Routed packets contain the address of each node it will traverse in order to get to its destination.

In [Ngu08], Nguyen describes the two fundamental approaches to multicast routing; Shortest Path Trees (SPTs) and Minimum Cost Trees (MCTs). SPTs aim at minimising the path from sender to receiver while MCTs main goal is to minimise the overall cost of the tree. MCT algorithms for multicast routing are based on the Minimum Steiner Tree (MST) problem, which is NP-complete. As a result, heuristics are used to compute approximate Steiner trees.

Due to the complexity of computing Steiner trees in a distributed manner, the majority of the multicast routing protocols used in the Internet today are based on SPTs, such as Distance Vector Multicast Routing Protocol (DVMRP) [WPD88] and Multicast Open Shortest Path First (MOSPF) [Moy94]. The reason is that SPTs are easy to implement and offer minimum end-to-end delay, a desirable quality of service parameter for most real-life multicast applications. In [RuG05, RGJ05], Ruiz *et al* explore optimal multicast trees in WMNs. The authors redefine the cost of an MCT by taking advantage of the wireless multicast advantage (i.e. minimising the number of forwarding nodes).

On-Demand Multicast Routing Protocol (ODMRP) [LGC99] is a mesh-based, rather than a conventional tree based, multicast scheme and uses a forwarding group concept. Therefore, only a subset of nodes forward the multicast packets via what is termed as scoped flooding. It applies on-demand procedures to dynamically build routes and maintain multicast group membership. ODMRP is well suited to ad hoc wireless networks with mobile hosts where bandwidth is limited, topology changes frequently, and power is constrained. In [RKD06] a modified version of ODMRP is developed using various routing metrics in order to take full advantage of the multicasting over WMNs.

Nguyen [Ngu08] demonstrated through simulation that SPTs offer significantly better multicast performance than MCTs. The author points out that SPTs are best suited to low density low sending rate environments. When the group size is large and the sending rate is high, large numbers of forwarding nodes becomes a concern.

3.2.1 Discussion

Although routing protocols for multicasting have existed for some time for wired networks, multicasting for WMNs is still in its infancy. Currently there is no support for multicast routing over WMNs in the existing IEEE 802.11 standard. However, at present the IEEE 802.11s amendment to the standard is being developed to allow interoperability between heterogeneous mesh network devices. In the Internet Engineering Task Force (IETF), the Mobile Ad Hoc Network (MANET) [IET09] working group has standardised many multi-hop routing protocols such as Ad Hoc On Demand Distance Vector Routing (AODV), Dynamic Source Routing (DSR), and Optimised Link State Routing (OLSR) [ClJ03]. These routing protocols are mainly developed for the deployment of unicast traffic and take into consideration mobility but do not directly address multicasting [CaK08]. Furthermore, multicasting over WMNs does not support RTS/CTS nor does it support ACKs due to the high probability of collisions at the transmitter and can therefore be classed as an unreliable service [RKD06]. There are many open issues concerning multicasting over WMNs. For example, reliable service, efficient membership updates, multi-radio multi-channel networks and quality of service guarantees are amongst those not covered in our discussion. From the literature there is no one particular routing protocol or metric that outperforms all others. Any solution put forward would need to take into account how the network is to be used and the specifics of the topology. To this end a suite of protocols would need to be developed for efficient multicast communications.

3.3 Rate Control

There are two main kinds of PHY line rate adaptation schemes for IEEE 802.11. One is based on SNR (Signal to Noise Ratio) and the other is based on a history of Acknowledgment (ACK) frame reception. For SNR rate adaptation, channel condition estimates allow the transmitter to determine when to use a higher PHY rate. Examples of such techniques can be seen in [HVB01], [SKS02], however, such techniques usually require a feedback mechanism and are not compatible with current IEEE 802.11 standards.

In ACK frame reception history, the transmitter makes the link adaptation decision to alter the rate based on the local ACK frame reception history. Auto Rate Fallback (ARF) [KaM97], which switches rates between 1Mbps and 2Mbps if two consecutive ACK frames are not received. The rate is raised again if ten consecutive ACK frames are successfully received.
ARF has been adapted [CJB03] to change the success thresholds to deal with fast and slow changing wireless channels. The ARF3-10 algorithm [CJB03] is an adaptation of ARF which uses a small success threshold of 3 and a large success threshold of 10. A more sophisticated adaptation is presented in [QiC05] which guarantees a minimum number of rate increasing attempts. In [XKW06] the authors observe a shortcoming in these previous techniques, namely the assumption that all data rates should use the same thresholds.

Their technique modifies ARF and utilises different success thresholds for different data rates.

Choi et al [CNP07] present an adaptive rate adaptation algorithm which focuses on link layer collisions. The basic concept is to eliminate unnecessary rate downshifts which cab be wrongly triggered by link layer collisions. Rate increasing and decreasing parameters are adapted using simple link layer channel estimation (i.e. the number of consecutive idle slots). The algorithm is validated using the NS-2 simulator which demonstrates the algorithm's effectiveness in yielding significant performance gains when compared to fixed threshold solutions using ARF.

Chou, Misra & Qadir [CMQ06] address the issue of low latency concerning broadcast in WMNs. The concept of a multi-rate link layer multicast is introduced to provide low latency multimedia broadcasts in WMNs. They take advantage of the uniqueness of broadcasts in WMNs by taking advantage of the wireless multicast advantage [WNE00] and avoid the broadcast storm problem [NTC99]. Their proposal is to use multiple transmission rates to broadcast the same packet to child nodes, rather than a single broadcast at the lowest rate. Simulations of their rate aware heuristic show a 3 to 5 times improvement in latency. This method however requires additional broadcasts which will consume more bandwidth and reduce transmission opportunities.

In [VST06] and [VCO07], Villalón et al present a cross layer design for auto rate selection in multicast. The authors claim that none of the other proposals to date have come up with a structured set of control mechanisms which take into account the varying conditions characterising the channel conditions as well as application requirements.

Traditionally, a multicast group rate is set by the worst connected node in order to maximise coverage. This can result in a poor performance in a WLAN due to a MAC anomaly as shown in [HRB03]. The Auto Rate Selection Multicast Mechanism (ARSM) dynamically selects the multicast data rate based on the channel conditions perceived by mobile nodes. This is achieved by identifying the Access Point (AP) to wireless node channel exhibiting the worst conditions, expressed in terms of SNR. A cross layer communication is then adopted between the PHY and MAC layer.

ARSM uses 3 types of feedback mechanism to determine the rate selection:

- Explicit Feedback - The AP receives a multicast response frame from a mobile node within the multicast group. The AP then decides the rate based on the worst SNR value included in the multicast response frame.
- Implicit Feedback - The AP predicts the SNR value of the mobile node with the worst channel quality when a corrupt multicast response frame is received.
- No Feedback - If no multicast response frame is received, the AP retransmits a probe request. This is also used to determine if neighbouring nodes have left the multicast group.

3.3.1 Discussion

While many sophisticated approaches have been put forward for rate adaptation it would seem that there still remain open issues when addressing rate adaptation not just in the specific case of multicasting but also in the more general case of WMNs. The modifications to ARF do not adequately address the issue of fast changing fluctuations in the network. Other methods mentioned here incur additional overhead either by retransmissions (which in effect defeats the purpose of multicasting) or by generating additional traffic through probe responses. The cross layer approach presented by [VCO07] is the most interesting and can be adapted to include WMNs.

3.4 Simulation Review

In his review of simulation techniques, Stojmenovic [Sto08] offers advice on how to carry out what he terms a *proper and effective simulation activity for protocol design*. Stojmenovic challenges criticism on validation aspects and advocates simple models to provide “proof of concept”. After which the complexity of the model can be increased by introducing one parameter at a time. At each stage algorithms should be revised, adapting to new assumptions, metrics and the corresponding simulation environment.

When it comes to the use of network simulators the authors of [TMB01] reveal that different simulation tools yield different results even when configured with the same protocols. It is noted that this difference is derived from assumptions made at the PHY layer. Furthermore, it can be shown [PNY03] that previously published work cannot be replicated because the authors do not fully report the conditions in which the simulations were carried out. The situation becomes even more difficult as the number of simulation parameters is increased. The authors of [PNY03] also point out that it is not enough to simply collect data but also to present detailed statistics so that the correct conclusions can be drawn.

Details of the mismatching of modelisation in network simulators such as NS-2, OPNET and GloMoSim can be found in [CSS02]. Model parameters can have default hidden values within the simulator. It is unlikely that these parameters will provide the exact scenarios the designer had in mind. However, for the sake of reproducibility [Sto08] states that all parameters, including default parameters should be made clearer.

This “lack of independent repeatability” is also reported in [AnY06]. Andel & Yasince state that all settings should be made clear to the reader so that simulations or experiments can be reproduced accurately. Andel & Yasince go further by criticising the “lack of statistical validity” which is also reinforced in [KCC05]. They state the need to determine the number of independent runs and sources of randomness in a simulation. Kurkowski, Camp & Colagrosso [KCC05] present interesting statistics on a remarkable number of publications which fail to specify details such as the simulator used, number of simulation runs or whether or not nodes were mobile etc. As such, they state that none of these simulations are repeatable.

In [HCG08], Hamida et al argue that for the sake of realism and confidence in simulation results, using accurate and detailed physical layer models is a key issue. Their main criticism is the additional processing overhead necessary in using such models. [HBE01] expand upon this concept by describing the trade-offs associated with detailed simulation models over five case studies of wireless simulation protocol design. Interestingly, the authors also advocate the use of visualisations to evaluate simulations and pinpoint incorrect details. The authors state that too much detail results in slow, cumbersome simulators, whereas, simulators lacking detail can be misleading or incorrect.

In [KNG04], Kotz, Newport et al review six assumptions which remain part of many ad hoc simulation studies. Their work demonstrates the weakness of these assumptions and also indicates that making such assumptions will cause simulation results to differ greatly from experimental results. The authors then set out a series of recommendations for designers of simulation models and protocols.

In addressing large scale simulations in wireless networks [TBL99] state that the free space loss model is computationally efficient but ignores many losses. Advances in large scale simulators focus on reducing the computational complexity while trying to improve the accuracy.

3.4.1 Discussion

In his review, Stojmenovic [Sto08] is heavily critical of current simulation practices. Stojmenovic presents an interesting analysis of 45 articles presented at the 2008 ACM MobiHoc Conference. The table below sums up the main critical findings.

Model/metric in simulations does not match the one in problem statement	25%
Simulation does not study one variable at a time	39%
Article does not give a thorough and critical literature review	41%
Simulation does not compare with truly competing solution(s)	59%
Average degree (density) is not an independent variable	69%
Article has no simulation section	22%

Table 3-1: Observed practices from 45 articles presented at ACM MobiHoc ’08 [Sto08].

The recommendations and omissions reviewed in this section would seem like obvious advice which is often taken for granted. It would appear however, that many designers have neglected to observe such practices. After criticism of practices and giving recommendations for best practices (such as basic simulations for proof of concept, simplistic models for tractability, solving one problem at a time i.e. study one variable before adding another) the authors conceded that there is no single solution to solving all problems. Different problems require different approaches and it is difficult to give general recommendations. Simulation results, and hence comparison between different methods, can be unreliable and therefore, caution should be exercised when drawing any conclusions from these works. However, simulation can provide an excellent method for quickly analysing the operation of new protocols as well as the operation of large scale networks which would not normally be possible (due to practicality of deployment and hardware cost) using hardware. When using simulation all assumptions should be made clear so that meaningful observations and comparisons can be made.

3.5 Topology Optimisation Techniques

In topology optimisation techniques the designer attempts to influence the network performance by modifying or adjusting the layout of node placement. Extensive work has been carried out in this area for WLANs [MLR01, PKT02, LKC02, KaU02, PKT04, WSL04, GiK05, BaC05, KUK05].

An example of a further improvement is presented by Vanhatupa et al in [VHH07], where a genetic algorithm is used to explore the design space of a WLAN alongside an IEEE 802.11 rate adaptation aware quality of service (QoS) estimator to provide feedback to optimise node placement and configuration. This is essentially a design tool in which the algorithm selects the AP devices, antennas, locations etc. The tool is also capable of selecting the AP configuration including the transmit power and channel frequency.

Compared to manual configuration the authors claim their tool will allow for a higher capacity and a lower deployment cost with 98% coverage (i.e. the size of the physical are where a user has a connection). This is coupled with a design implantation taking approximately 15 minutes.

The genetic algorithm (GA) makes use of a “fitness function” which uses weighted QoS metrics, set by the designer, to output a non-negative indicator on how good the network plan is. A HexagonGA is employed which uses the fitness function in order to find sets and solutions. Only solutions found with high fitness are used for next generation solutions.

In [ABS08], Alotaibai et al point out that node placement is normally through ease of equipment placement and connectivity to targeted area. They state that the problem of adding nodes to an established network in a uniform manner for a given budget constraint is not advisable. The authors claim to have developed an algorithm to determine the best possible locations to place nodes in order to improve performance. By extending the work of Gupta & Kumar [GuK00] they show that the throughput T available to each user, for n randomly deployed nodes, each transmitting at W bits per second, is in the order Θ of,

$$T = \Theta\left(\frac{W}{\sqrt{n \log n}}\right) \tag{3.3}$$

By adding m relay nodes this now becomes,

$$T_{new} = \Theta\left(\frac{(n+m)W}{n\sqrt{(n+m)\log(n+m)}}\right) \tag{3.4}$$

This leads to the performance gain;

$$\rho = \frac{T_{new}}{T} \tag{3.5}$$

From their calculations they show that adding nodes in a uniform manner such as this is not beneficial as the cost of deployment would far out weigh any benefits. This can be clearly seen by the number of additional nodes required to meet specified performance gains (see Table 3-2).

Nodes	**Relays**	**Gain**
n = 10	m = 200 m = 700	ρ = 3.01 ρ = 5.01
n = 50	m = 750 m = 2500	ρ = 3.06 ρ = 5.04
n = 100	m = 1400 m = 4500	ρ = 3.07 ρ = 5.01

Table 3-2: Additional nodes required for specified gain

The work presented by [ABS08] uses OPNET to simulate an 802.11a WLAN fixed at 12 Mbps, using a single channel and fixed transmission power. 11 fixed nodes are used in the simulation with the introduction of 2 new nodes. The work presented does not clearly describe how the algorithm works to achieve optimal node placement. They do however state that best performance improvement was for inter cluster placement and that worst performance was for intra cluster placement.

Robinson et al [RUS08], present work on addressing the problem of adding new gateway nodes to an existing mesh in order to alleviate network traffic/load. The authors present a new technique for calculating gateway limited fair capacity as a function of the contention of each gateway. The authors define the gateway-limited fair capacity as a function of the airtime utilisation of the gateways, which depends on the routes used and amount of time the routes lead to a gateway deferring. Two separate gateway placement algorithms are used with local search operations to maximise that capacity gain on an existing network. The two placement algorithms are adapted solutions based on Minimum Hop Count and Minimum Contention techniques. *MinHopCount* minimises the average hop count for all paths in the network. *MinContention* minimises the average contention size and gives better performance guarantees than *MinHopCount*. The authors compare their technique to an exhaustive search of all gateway node placements. Local search is comparable to optimal solutions and considerably better than near optimal solutions (64% performance improvement in network capacity over greedy heuristic).

In [GMR08], Gomes, Molle & Reyes note that the capacity of wireless mesh is under utilised as the size of the network is increased. Therefore maximising the capacity requires optimising the gateway placement and routing while taking interference into account. A mixed integer linear programming (MILP) technique for computing 802.11a and 802.16 WMNs which provide maximum bandwidth guarantee is presented.

Grid and random mesh topologies are used during simulations. The authors claim that small networks (4 x 4 grid) generate MILPs with 1000s of variables and constraints and stating that large instances cannot be solved.

The MILP either gives the maximum throughput guarantee for every router in the network or the minimum number of gateways required satisfying a given traffic demand. The main objectives for MILP presented are to solve the gateway placement problem (fixed number of gateways to reach maximum throughput); to solve the optimal gateway placement problem; to solve the fair routing scheduling routing problem. The work presented allows MILP to find near optimal solutions for small sized problems.

3.5.1 Discussion

By expanding on the work carried out by Gupta & Kumar [GuK00], it can be seen [ABS08] that simply adding new nodes to an existing network in a uniform manner is an inefficient method for improving network performance. It is clear that a more considered approach is necessary. Much of the work carried out has considered the boundary limits of WLANs or ad hoc networks as opposed to fixed node mesh networks. Furthermore, much of the research has centred attention on providing additional gateway nodes without fully exploring methods of optimising the single gateway network (or indeed in the case of networks not connected to the Internet, no gateway at all). Furthermore, such research has also put the focus on considering unicast traffic when considering optimal networks. Sophisticated techniques have been employed in discovering the optimal positioning of such nodes and have largely ignored the optimal positioning of mesh nodes. This can be due to a design defect, where nodes are initially placed through ease of deployment rather than optimal positioning. Optimal positioning is possible through simulation but in practical networks this will not be feasible due to the associated labour costs and unpredictability of the wireless medium.

3.6 Capacity of Wireless Mesh Networks

As mentioned in the previous section, Gupta & Kumar, in their much cited work on the capacity of wireless networks [GuK08], have derived a set of lower and upper boundaries for ad hoc network capacity. Guidelines are provided on how to improve the capacity of ad hoc networks by simply stating that, “a node should only communicate with nearby nodes” and that nodes should be grouped into clusters. They further state that throughput capacity can be increased by deploying relay nodes. In other words, communication of a node with another node, that is not its close neighbour, must be conducted via relay nodes or clusters.

This is a challenging task considering the distributed nature of ad hoc and WMNs, as stated by [AkW05]. In their survey of WMNs [AkW05] states that analytical approaches such as [GuK00] and [GrT02] have significantly driven much of the research progress in wireless network capacity. However, [AkW05] also criticises these two approaches for failing to adequately capture more sophisticated networking protocols by use of over simplified analytical models.

Such analytical models fail to determine the exact capacity of a network with a given number of nodes. Moreover, due to differences in ad hoc and WMNs such analytical models will not hold and therefore a new model is needed.

Jun and Sichitiu [JuS03] try to overcome these shortcomings by attempting to determine the exact capacity of a WMN. They achieve this by introducing a concept termed as the bottleneck collision domain in order to calculate the capacity of a WMN. The authors define this as “the geographical area of the network that bounds from above the amount of data that can be transmitted in the network”. In other words, a bottleneck collision domain (BCD) is one that has to transfer the most traffic in the network. There can be more than one BCD, all transferring the same amount of traffic. The BCD will throttle the throughput of the entire network. It is worth noting that [JuS03] use a single gateway and assume that all nodes receive an equal share of the available bandwidth through some fairness scheme (not described).

In a heavily mathematical approach, Yang-Li [LiX09] addresses the capacity of wireless ad hoc networks for the specific case of when multicasting/broadcasting is used. Yang-Li presents a detailed analytical model in order to derive the upper and lower bounds on the multicast capacity of a random wireless network. The derivation presents the nominal capacity for the upper and lower bound conditions such that the multicast capacity $\Lambda_k(n)$ is;

$$\Lambda_k(n) = \begin{cases} \Theta\left(\sqrt{\dfrac{n}{\log n}} \bullet \left(\dfrac{W}{\sqrt{k}}\right)\right), & \text{when} \quad k = O\left(\dfrac{n}{\log n}\right) \\ \Theta(W), & \text{when} \quad k = \Omega\left(\dfrac{n}{\log n}\right) \end{cases} \tag{3.6}$$

This essentially describes order Θ, of the capacity of the upper and lower boundaries for a uniform random deployment of n nodes in a square region connected to a group of k nodes (i.e. when the multicast group is large, $k = O\left(\frac{n}{\log n}\right)$ or small, $k = \Omega\left(\frac{n}{\log n}\right)$). Each node is assumed to have a uniform transmission and interference range and will transmit at W bits/second. Yang-Li further describes how simultaneously transmitting nodes can be separated by a minimum distance in order to eliminate interference.

3.6.1 Discussion

Determining the capacity of a wireless network is a non trivial task. As we have seen there have been several attempts to provide upper and lower boundary limits. Subtle differences in ad hoc and WMNs mean different models should be tailored to deal with the specifics of each. The majority of work to date has considered unicast traffic and not fully explored the capacity of a network for broadcast/multicast. Yang-Li's detailed description [LiX09] falls into the same trap as previous work criticised by [AkW05] in that the analytical model fails to determine the exact capacity of a network for a given number of nodes nor does the method take account of protocol advancements which can considerably affect the capacity. The performance of mesh networks is largely dependent on the topology and as such, boundary limits will only give the general capacity under a very broad range.

3.7 Wireless MAC Anomaly

We will briefly discuss an anomaly which was first presented in [HRB03]. The anomaly was discovered in IEEE 802.11b WLANs when stations connected to the same access point (AP) operate at different PHY rates. As a result, when some mobile hosts use a lower bit rate than others, the performance of all hosts connected to the same AP is considerably degraded.

This is a common enough occurrence due to nodes situated far from an AP which are then be subjected to a poor Signal to Noise Ratio (SNR). When this occurs a node will switch its modulation technique and reduce its bit rate to a lower value. This results in all nodes connected to the AP having a degraded throughput due to the low bit rate of the node with the poor SNR.

In [HRB03], Heusse et al demonstrate how in the case of 2 stations, connected at 1Mbps and 11Mbps respectively, the station connected at 11Mbps will have its throughput reduced to below 1Mbps.

The anomaly is further discussed through a simple example given in [BRC05]. Assuming that each station has an equal probability of transmission and is observed over a long period of time, it can be said that the two stations will have an average transmit proportion of the time C. Therefore,

$$C_{11Mbps} = \frac{1}{12} \qquad \text{and} \qquad C_{1Mbps} = \frac{11}{12}$$

If the transmission rate efficiency factor ρ is considered [Uni05], then the throughput of each station is given as;

$$R_{Tn} = \rho_n R_n C_n \tag{3.7}$$

This simple calculation shows how a lower throughput can be achieved when using a higher transmission rate. This is largely due to transmission efficiency and the CSMA/CA mechanism guaranteeing an equal long term channel access to all nodes and penalising hosts using higher rates.

The authors of [BRC05] present work which attempts to counter the effects of the anomaly by implementing traffic priority on the MAC in order to favour stations with a better SNR.

3.7.1 Discussion

Although the anomaly is discussed in terms of WLANs the same issues hold for WMNs where the CSMA/CA mechanism is in operation. This degradation of throughput would be alleviated in multicast mesh due to the fact that parent nodes will transmit to all children at the same rate. Nevertheless, the situation would still arise when neighbouring parent nodes are forwarding to their respective child nodes at different rates (e.g. parent node A transmitting at 11Mbps, parent node B transmitting at 1Mbps). Therefore, we must take the interference range into careful consideration when placing relay nodes or when changing the topology in general.

3.8 Network Coding

In this section we will discuss a more recent approach to solving network communication issues which has gained popularity in wireless network applications. In [ACL00] Ahlswede et al introduced a new class of problems called *network information flow* which took its inspiration from computer network applications. The authors redefine information flow in a computer network by applying a technique called network coding which essentially increases the information content of each transmission. The traditional approach for multicasting uses the notion of replicating data at nodes so that each multicast group member eventually receives a copy of all the data. The authors state that in classical information theory this type of approach holds for point-to-point communications. However, for multicast communications this type of technique is not optimal because information needs to be coded at each of the nodes. Furthermore the problem becomes more complicated when there is one source. The authors observe improvements in the network when techniques such as random block codes and convolutional codes are used in the single source point-to-multipoint case. Such coding techniques enable intermediate nodes inside the network to code and decode the information carried by different flows.

Since the ground-breaking work of Ahslwede et al [ACL00] there has been growing popularity in the area of multicasting point-to-multipoint communications in both wired networks [CWJ03, JLC04, ZLG04] and wireless networks [DEH05, KKH05]. As a consequence of much of this work, coding is carried out inside the network by forwarding nodes while decoding is now carried out by each of the multicast group receivers. In [HBT06], Hamara et al investigate through simulation the specific application of file sharing over wireless mesh networks using network coding. The authors highlight the difference between wireless and wired networks in that wireless nodes cannot listen to more than one neighbour simultaneously (i.e. for the case of multiple sources). However, the broadcast nature of wireless networks has the obvious advantage of allowing multiple neighbours to receive a single coded transmission. The neighbouring nodes will then decode the required parts only. The authors state that the main parameters influencing the performance of file sharing in wireless networks when network coding is used are: the number of nodes, how fragmented the data is, the number of source nodes and their location, and the cooperation between nodes. The authors conclude that the main benefits of network coding can be observed in networks which would normally experience heavy losses.

In [KRH08], Katti et al state that much of the research carried out on network coding has been theoretical to date. The authors present experimental results using their coding scheme on a 20 node wireless test-bed. In this paper Katti et al specify that a minimum resource requirement is necessary for static WMNs for the successful operation of their network coding technique. For example a minimum amount of memory storage is required as well as the use of omni-directional antennas for opportunistic exploitation of the broadcast property. The authors also note that their method does not optimise the power usage based on the assumption that nodes are not energy limited. Their results show how the aggregate throughput can be increased by maximising the amount of data delivered by the use of network coding.

More recently in [YLL09], Yang et al investigate the problem of providing a reliable broadcast mechanism in WMNs by implementing network coding. The aim of the authors' coding scheme is to provide a reliable wireless broadcast service while simultaneously reducing the broadcast overhead and delay. The simulation used in this paper allows for broadcast trees to be constructed using minimum number of transmissions spanning trees. The authors claim to achieve through simulation 100% packet delivery ratio (i.e. the reliable delivery of an entire broadcasting file to all nodes) by ensuring each node is covered by the most efficient neighbour. The authors make the assumption that a local optimal coding solution can achieve an improved global performance. The coding technique presented, R-Code has shown through simulation to reduce the average number of transmissions and delay by 14% and 50% respectively.

3.8.1 Discussion

The research in network coding in recent years is quickly becoming an exciting and popular method for enhancing the performance of wireless networks. The broadcast/multicast advantage offers benefits that are well suited to network coding. Until recently much of the work carried out in network coding has been purely theoretical however, recent advancements have changed this. The most obvious disadvantage to network coding is the additional resources and processing power requirements necessary for its implementation. In light weight nodes this is less likely to be an option as production costs will ultimately dictate the development. However, network coding presents a novel approach to increasing the performance of wireless networks which can be coupled with other techniques to yield performance gains which cannot be ignored by hardware manufacturers.

3.9 Chapter Summary

In this chapter we have given an extensive overview of the current methods for obtaining routing metrics and the protocols used for routing traffic over wireless networks. Many of these methods are taken from unicast and are simply expected to work in multicast. The Minimum Hop metric for example is often used due to its simplicity and minimal overhead. This type of metric does not consider the performance of the network, however it does work well in mobile environments which require a high degree of adaptability. ETT on the other hand utilises the highest available bandwidth of each link by considering influential network factors. Many multicast solutions are modifications to existing unicast routing metrics coupled with modifications to the routing protocol. Shortest Path Trees (SPT) offer significant performance improvement to multicasting over alternative methods. It is however, recommended that the density and sending rate be kept low to maximise such benefits. When the group size is large and the sending rate is high, a large number of forwarding nodes becomes a concern. This is due to increased interference and contention from neighbouring nodes. Furthermore, rate adaptation should be utilised in order to take advantage of multicasting. Rate adaptation has many open issues such as overhead generated from probe packets and the use of retransmissions which effectively defeats the purpose of multicast. As such, a suite of protocols is necessary to develop efficient multicast communications which take into account the network topology as well as its intended use.

Our review of network simulation methods exposes some of its short-comings. However, simulation can be used as a valuable tool when used correctly. The level of detail used in simulation should be carefully considered and all assumptions should be clearly made. Simulation offers the obvious advantage of requiring little cost compared to hardware solutions. Simulation should be used as proof of concept and caution should be exercised when comparing simulation results using different tools. There is a trade-off between software and hardware solutions when used in academic research and industry. Using hardware in academic research is no guarantee of a practical solution due to hardware platforms and licensing used [RMC08]. This is due to industrial solutions motivated by cost of production and revenue potential.

Determining the capacity of a wireless network is a non-trivial task with the focus being mainly on providing upper and lower boundary limits. Such limits do not provide information on the capacity of a network when a specific number of nodes are used. Furthermore, optimisation of the network topology has mainly provided solutions through increasing the capacity of the network by placing gateways in strategic positions and has largely ignored optimising the network itself. The reason for this is that much of the research has been carried out on ad-hoc networks were there is a high degree of mobility rather than a fixed mesh network (were nodes are repositioned in order to provide a network gain). The much cited work of [GuK00] suggests that relay nodes should be used between clusters. However, using their calculations to determine the number of relay nodes necessary to provide a useful network gain, yields unrealistic results. Clearly a more considered and fully thought through approach is necessary to provide a more practical solution.

In this chapter we also discuss the implications of the MAC anomaly which can be observed when nodes are transmitting at different rates. Although we have stated that many multicast solutions use a fixed transmission rate across all forwarding nodes we must be aware of the effects of the anomaly if a method using different transmission rates is used.

Finally, we introduce the relatively new research area of network coding in wireless networks. Network coding essentially increases the information content of each transmission allowing the receivers to decode the necessary parts and hence improve the efficiency of each transmission. Many practical issues still remain with network coding, however, it is hoped in the near future that many of these will be resolved.

4 Framework for Analysis

In this chapter we will outline our development of a custom wireless network simulator. We will describe the progressive stages of the development which will justify our starting assumptions for a basic multicast simulator before further development to provide performance enhancements to the network. Our methodology allows us to develop the simulator by making small changes to the functionality and adding necessary detail as required. After careful analysis of the Basic Model we identified potential performance bottlenecks on the network specific to wireless multicasting. Furthermore, we pose the question of how we can reduce or eliminate the effects of such bottlenecks?

Our goal is to fully describe the steps taken in developing the simulator and to clearly state any assumptions made. The objective is to develop an abstract model to evaluate cause and effect of poor performance and to provide a proof of concept solution through simulation. Design requirements and development of the simulator is defined using recommendations outlined in [Sto08]. The model is defined by the working parameters and the constraints placed upon it. The simulator was developed over a period of approximately 24 months (80% developed in the first 12 months and 20% in the second 12 months)

Large data files of network statistics were generated during each simulation. Perl is ideally suited through the use of regular expression for pattern matching and processing such files. The primary function of the Perl simulator is for evaluation purposes. However, due to the modular nature of the simulator, the software would require minimal effort for further reuse.

.

We use Dijkstra's shortest path algorithm to construct network spanning trees using various link cost metrics. Each link cost metric is used independently and a performance analysis is carried out upon successful completion of each set of simulations. Source code for all simulation models described can be found in Appendix E.

4.1 Simulation – Basic Model

Based on the discussions covered in Chapter 3 we have developed a custom simulator. It was determined that a custom simulation model allows us greater freedom and flexibility in implementing a multicast wireless mesh simulator. Although it is possible to undertake this type of simulation using commercially available or academic based simulators, a custom simulator would allow us to modify and build the simulator in a specific manner without the need for unnecessary detail. Developing a custom simulator affords us the ability to modify and add new features as required with the flexibility to only include the mechanisms which are of interest to us.

In order to develop our simulator a number of initial assumptions needed to be made. Following the recommendations outlined in [Sto08] we undertook to develop a simulation tool by constructing a simple model, adding one new parameter, testing it and then progressing on the next level of complexity. The overall design has resulted in an abstract simulation model which includes a level of detail [HBE01] that we feel is appropriate to yield useful results.

The model was fully developed using the Perl programming language [PPL10] and all results are logged and graphed using Gnuplot [Gpt10]. We have also made use of the GD library [LGD07] to display the topological information. The working plane dimensions are based on a five hop path across the diagonal using the maximum transmission range. The initial model had the following characteristics.

- The nodes will be randomly placed in a 2D plane of specified dimensions i.e. 650m x 650m, giving an area of 422.5 x 10^3 m^2.
- We will allow for various node placement patterns through random distributions with various node densities.
- Each node will be allowed to transmit a given packet only once (this is to conform to current 802.11 MAC broadcasting methods).
- The transmission rate for a point-to-multipoint is dependent on the lowest available PHY rate of an IEEE 802.11b network (the number of line rates can easily be adapted to suit 802.11a/g).
- We will use Dijkstra's Shortest Path Algorithm to calculate the minimum path and construct network spanning trees. Link cost performance will be assessed by analysis of network delay and throughput performance.
- For comparative purposes we will use the link cost metrics; minimum hop (*MinHop*), minimum distance (*MinDist*), minimum path contention (*MinCont*), minimum transmit power (*MinPower*) and modified ETT.
- Performance metrics measured will be, maximum path delay, average network throughput, average path length, number of forwarding nodes, number of nodes used and percentage coverage.

The purpose of the Basic Model is to enable a performance analysis of a basic multicast network. The graphical representation of the network allows us to identify potential problem areas which can then be addressed to enhance the network performance.

4.1.1 Perl Simulator – A Basic Model

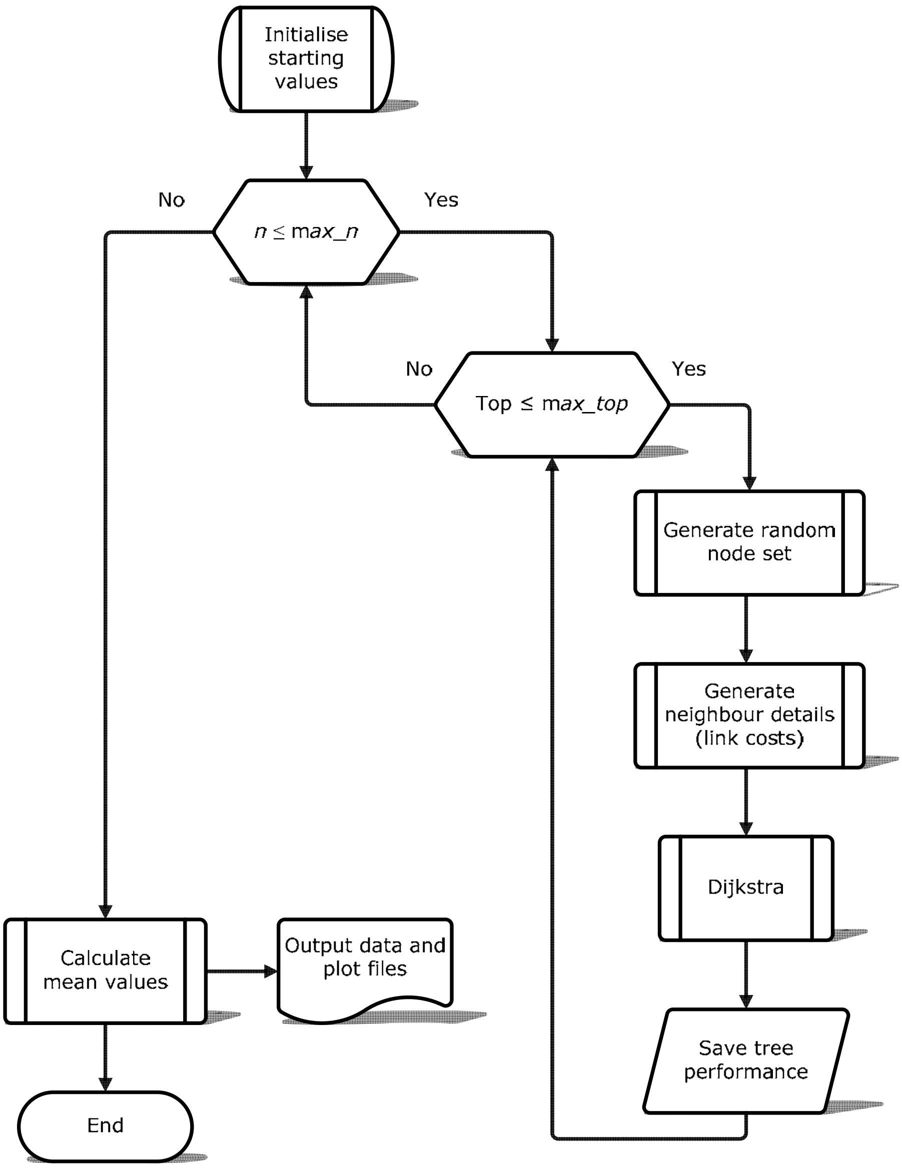


Figure 4-1: Basic simulation flow chart.

In Figure 4-1 we illustrate the steps taken in our Basic Model. In this example we allow for the simulation to increase the number of nodes, n and stop at some maximum, max_n. For each set of nodes a new random topology, Top is generated up to a maximum of max_top times. During this cycle the link costs of every node, in relation to every other node, is calculated. A link cost is chosen and used by our implementation of Dijkstra's shortest path algorithm. The performance of each tree is recorded and later used to calculate performance statistics.

Random node placement was implemented using a Mersenne Twister random number generator [HeJ08]. We will see later how we developed this Basic Model to include network performance enhancements and also a method to escape from getting trapped in local maxima with regard to the tree performance. Tree performance is based on the maximum path delay, mean network throughput and percentage node coverage.

It can be shown that the interference range is not static but instead is a function of the distance between transmitter and receiver [LZL06] [XGB03]. However, in order to maintain the simplicity of our model we will assume a fixed interference range. We define the interference range as the range within which other transmitters can interfere with ongoing neighbouring communications and will therefore contribute to the contention. The transmission range is the range within which a neighbour can successfully receive data at a certain rate. Each rate will have a set transmission range depending on the transmit power (see section 4.3 for range versus rate curves, Figure 4-14 and receiver sensitivity values, Table 4-2). Our simulation assumes the following starting conditions;

- Each node is identical with homogenous settings.
- The simulator operates using a single fixed channel with omni-directional antennas.
- A node can communicate with any other node within the maximum communications range (assuming a circular radio transmission area).
- A node can interfere with any other node within its interference range.
- A fixed working plane is used with boundary edges (i.e. not an infinite plane).
- A single fixed source node is used, termed the *Root*.

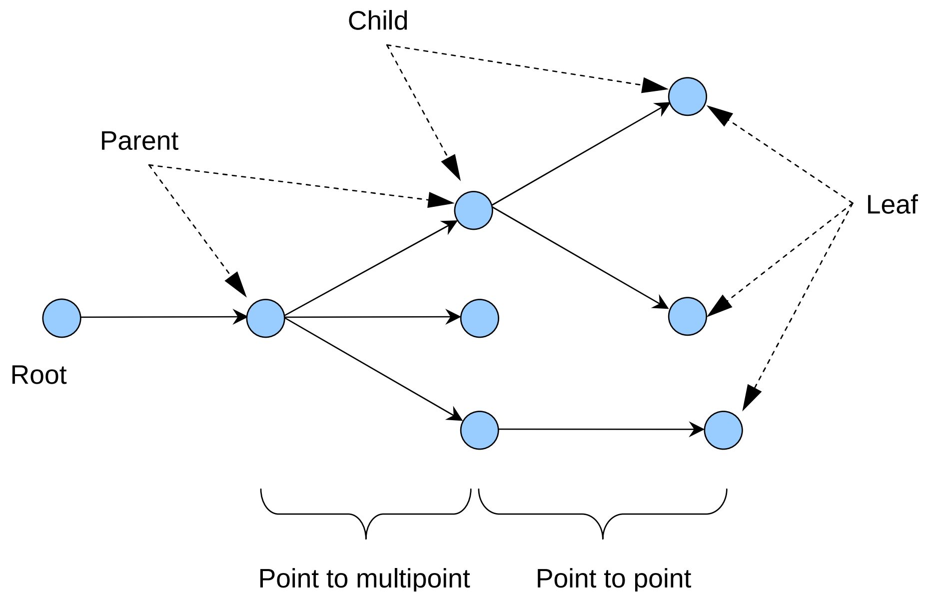


Figure 4-2: Example of network hierarchy.

We define the *Root* node as the multicast source node. We further define a *Parent* as a forwarding node to one or more neighbouring *Child* nodes. A

Child node can also be a *Parent* if it forwards traffic. A *Leaf* node is last hop *Child* and does not forward traffic.

4.1.2 Node Generation & Neighbour Discovery

A random distribution of fixed nodes is generated for each new topology. Each distribution is generated using a Mersenne Twister random number generator and is generated from an allocation of nodes. Nested hash tables are used to create structured records of each node. Structured records of neighbour details are generated for each node in relation to every other node in the topology. The structured records store data pertaining to the link cost associated with each node. A node will either be;

- A neighbouring node - A node within the communication range of another node.
- An interfering neighbour - A node outside the communication range but within the interference range of another node.
- Not a neighbour - A node outside the interference range of another node.

Once the neighbour discovery process has taken place the structures are populated with link costs. This type of hierarchy makes it relatively straight forward to include new link cost metrics as needed. Once the structures of nested hash tables have been updated for all nodes they are made available to the next stage in the process, the Dijkstra shortest path algorithm.

4.1.3 Implementing Dijkstra's Shortest Path

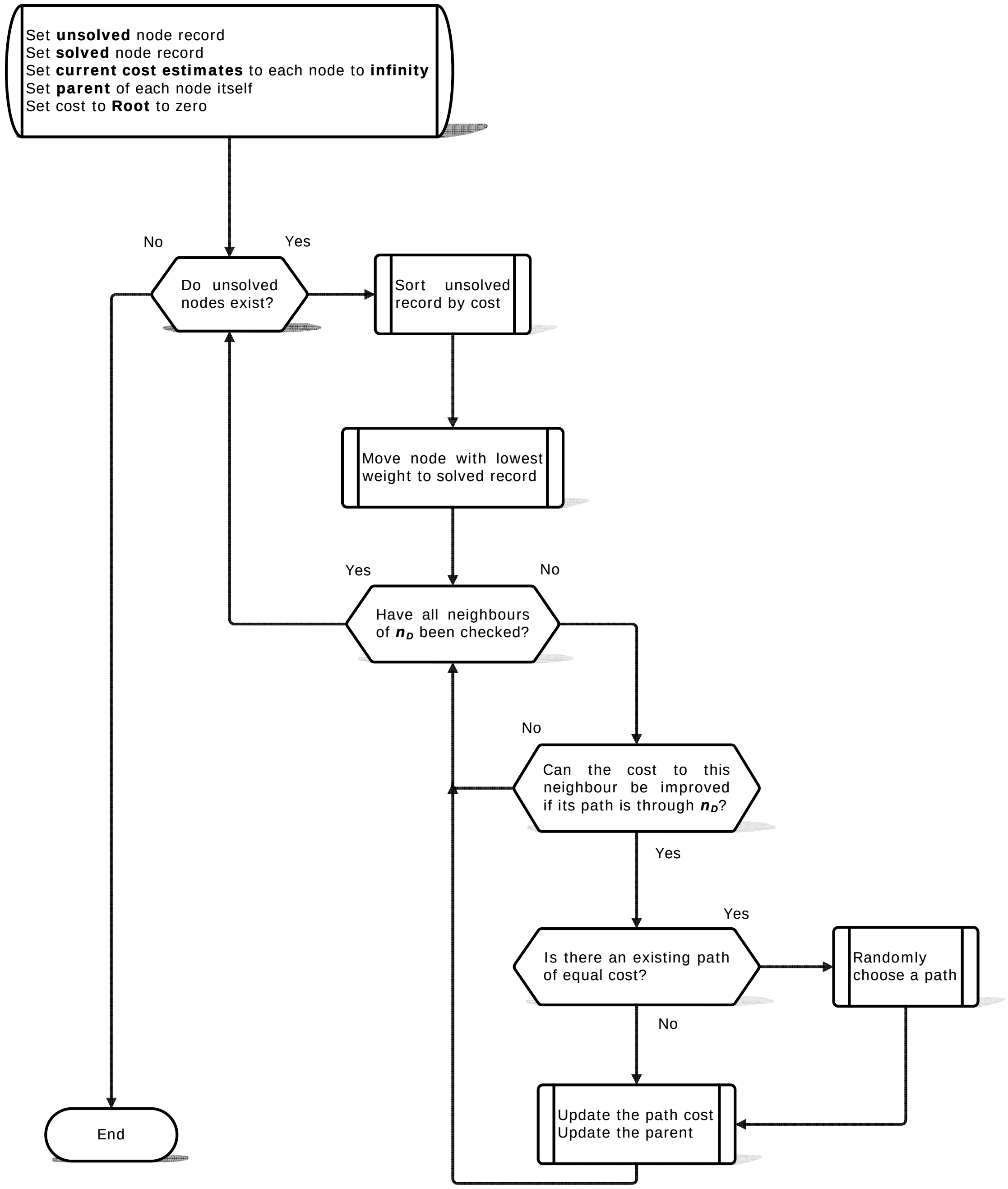


Figure 4-3: Operation of our implementation of Dijkstra's shortest path algorithm.

There is one parameter passed to our Dijkstra procedure, the link cost metric. The Dijkstra algorithm will use this to access the relevant neighbour link costs from the neighbour structures. It should also be noted that the basic implementation of Dijkstra does not take into consideration paths of equal cost. Once Dijkstra finds a shortest path it does not discriminate in any way against two or more paths with the same link cost. Therefore, in our implementation when paths of equal weight are discovered we randomly choose one of the paths each time. We illustrate the general operation of our implementation of Dijkstra in Figure 4-3.

4.1.4 Performance Metrics

In order to assess the performance of a network topology a set of performance metrics are required. The performance metrics are necessary for evaluation and will depend on the design goals of the network. We define a destination node as $n_D \in S$, where S is the set of nodes in the multicast group. The multicast source node is termed $Root$ and the $Path\ Rate$ is the lowest transmission rate of a neighbour in a multicast branch. We define three core performance metrics for our simulations, relative delay, mean network throughput and transmission contention. The relative path delay is the delay associated with transmitting data from one node to another at a particular transmission rate and is a function of the maximum line rate. The assumption is that transmissions taking place at lower path rates will have a higher relative delay (e.g. for path rates of 11 Mbps and 5.5 Mbps, relative delay will be 1 unit and 2 units respectively). As each simulation will be repeated with a random deployment of nodes a statistical mean is calculated over all topologies considered, so that;

Delay:

$$Relative\ Path\ Delay, d_R = \sum_{Root}^{n_D} \left(\frac{Line\ Rate}{Path\ Rate} \right) \qquad (4.1)$$

where $Line\ Rate$ is the physical layer transmission rate in Mbps. The average worst case delay for all paths in the network is then calculated using the maximum relative path delay ($max\ path\ delay$) on each topology. The $max\ path\ delay$ is the worst case path delay in a particular topology. Hence this will provide a conservative performance metric.

$$Mean\ Delay,\ d = \frac{\sum_{top}^{\max\_top} max\ path\ delay}{max\_top} \qquad (4.2)$$

As the relative delay is a function of the line rate and path rate it will have a dimensionless unit.

Throughput:

The mean network throughput TP is derived from a conservative calculation based on the minimum throughput, TP_{min} along a path. The throughput at each node, TP_n depends on the available link rate and the number of contending neighbours.

$$Throughput, TP_n = \frac{link\ rate}{No.\ of\ \ Neighbours + 1} \qquad (4.3)$$

$$TP_{min} = \frac{\sum_{n}^{n_D} min\ path\, TP}{number\ of\ \ paths} \qquad (4.4)$$

$$Mean\ \ Network\ Throughput, TP = \frac{\sum_{top}^{\max\_top} TP_{min}}{max\_top} \qquad (4.5)$$

where a neighbour is defined as any node within the interference range. The mean network throughput is measured in Mbps.

Transmission Contention:

The transmission contention is dimensionless and is based on the number of competing neighbouring nodes and the rate at which they will transmit. As with the throughput calculation, the mean network contention, C is derived from the sum of the individual path values in each topology.

$$Contention,\ C_{Tx} = \left(No.of\ \ Neighbours + 1\right) \times \left(\frac{Line\, Rate}{Path\, Rate}\right) \qquad (4.6)$$

$$Path\, Contention,\ C_p = \sum_{root}^{n_D} C_{Tx} \qquad (4.7)$$

$$Mean\ Network\ \ Contention,\ C = \frac{\sum_{top}^{\max\_top} \max\ C_p}{max\_top} \qquad (4.8)$$

In our results we will also provide additional performance analysis on such aspects as node coverage, transmission power, and number of forwarding nodes, amongst others.

4.1.5 Simulation Assessment

In order to develop our simulator it was necessary to carry out a number of performance assessments with the intention of establishing a well defined Basic Model. The Basic Model does not provide any enhancements or optimisation techniques. However, it does contain all of the core functionality of the multicast simulation process. Our simulation is intended to operate as an abstract model implementing core functionality and is then used to develop new enhancements and optimisation techniques. To help achieve this we analyse the performance of the Basic Model under various initial conditions/assumptions which will then be refined to create a basis for further simulations and development. Each stage of development is logged and validated in accordance with the recommendations outlined in Chapter 3 so that we have a well defined methodology.

Link Cost Evaluation

A number of link cost metrics were assessed on their performance when implemented in the Dijkstra shortest path algorithm. Among these were minimum hop count (*MinHop*), minimum contention (*MinCont*, based on contending neighbours) and minimum distance (*MinDist*, the minimum distance in this case is the minimum Euclidian distance. For wireless networks the use of this link cost is not entirely practical. This is due to multi-path propagation and the practicalities of determining a precise location of wireless nodes. However, it is useful for evaluation purposes). In Figure 4-4 and Figure 4-5 we give examples of the performance characteristics when using different link cost metrics in Dijkstra's shortest path algorithm. In the example we use between 10 to 150 nodes incremented in steps of 10 nodes. The dimensions of the working plane were chosen in order to ensure 5 hops could be obtained across the main diagonal at the maximum transmission distance (i.e. at 1 Mbps). The nodes are randomly deployed over an area of 650m x 650m (422.5 x 10^3 $m^{2)}$ and simulations are repeated 1000 times for each node density. Using 1000 samples is generally accepted to be sufficiently large to provide a statistical mean. The distribution of nodes can be seen to be uniform over the working plane when 1000 topologies are used. This is illustrated in Appendix F. The transmit power is held fixed at 9 dBm. Table 4-1 gives the approximate node density conversions.

Nodes, *N*	**Node Density, Nodes/m^2 x 10^{-6}**
10	24
20	47
30	71
40	95
50	118
60	142
70	166
80	189
90	213
100	237
110	260
120	284
130	308
140	331
150	355

Table 4-1: Approximate node density values for fixed nodes on an area of 650m x 650m.

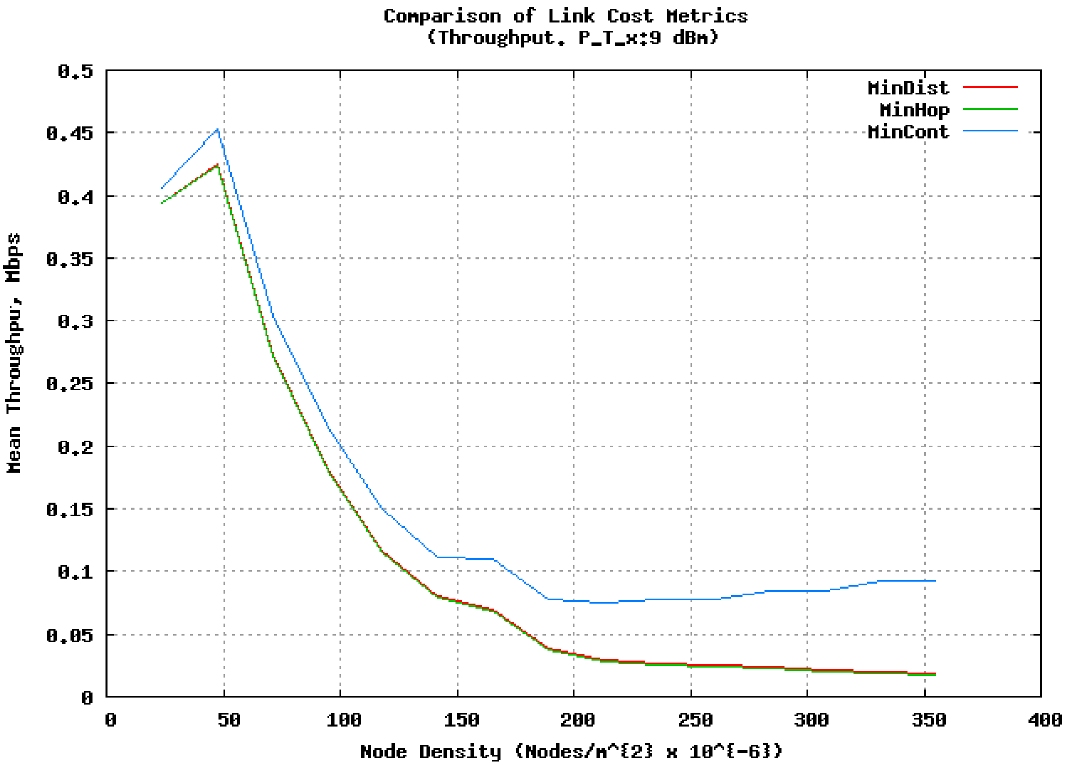


Figure 4-4: Link cost metric evaluation for mean network throughput in the Basic Model.

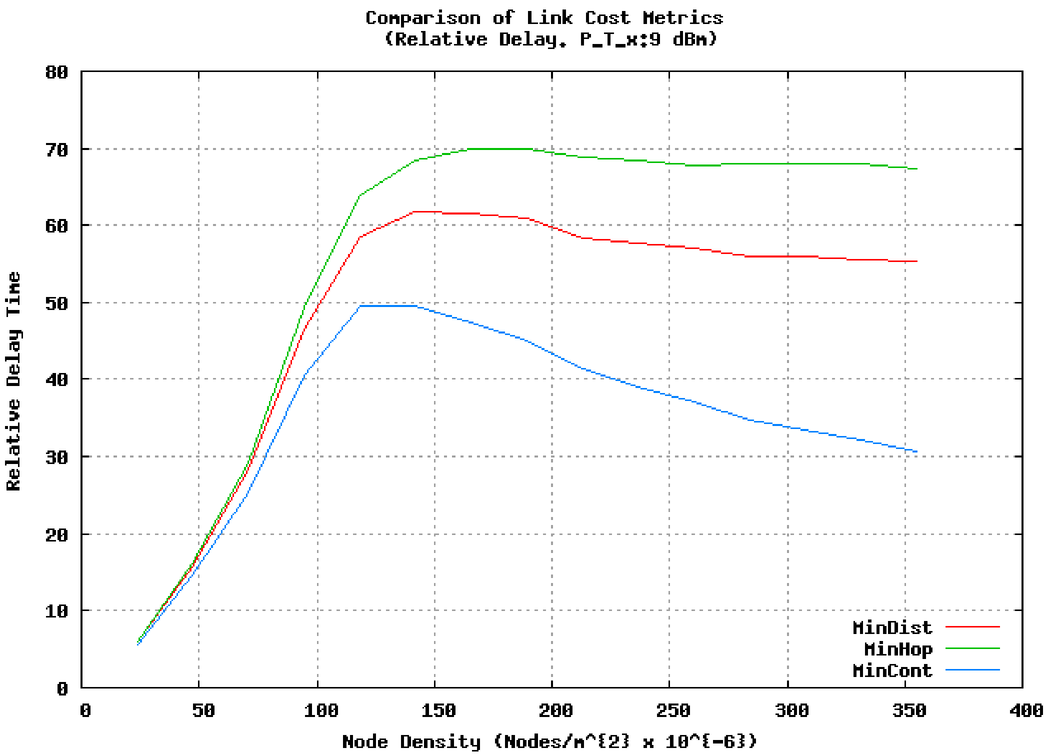


Figure 4-5: Link cost metric evaluation for network delay in the Basic Model

The figures show how different link costs metrics perform while increasing the number of nodes in the network. When used on the same network, the link cost metric will show different results, as would be expected. From the throughput performance in Figure 4-4 we can see that *MinCont* performs better than *MinHop* or *MinDist*, resulting in a higher mean network throughput. When we look at the delay performance in Figure 4-5 we can see that the *MinCont* outperforms the other metrics again. These results are due not just to the characteristics of the link cost metric but also to the topology of the network itself. We will see in the next section how the placement of the multicast source node *Root*, affects the performance of the network.

In our simulations we will also include the use of a modified Expected Transmission Time (ETT) link cost. Our version of ETT takes into account the transmission efficiency factor ρ as described in [Uni05]. The efficiency factor is based on the associated overhead required to transmit frames at each of the available PHY rates. The overhead increases as the rate increases resulting in a lower efficiency, thus giving us;

$$ETT = \sum_{i=1}^{n} ETX_i \times \left(\frac{S}{B_i} \right) \times \rho \qquad (4.9)$$

$$= \sum_{i=1}^{n} \left(\frac{1}{1 - P_i} \right) \times \left(\frac{S}{B_i} \right) \times \rho \qquad (4.10)$$

where S is the packet size and B is the actual data rate and

$$P = 1 - (1 - P_f)(1 - P_r) \qquad (4.11)$$

where P_f and P_r are the forward and reverse packet loss respectively.

As multicasting does not support ACKs and if we assume no forward packet loss this can be reduced to:

$$ETT = \left(\frac{S}{B}\right) \times \rho \qquad (4.12)$$

Multicast Source Node Placement, *Root*

We mentioned previously how the performance results can be affected by the network topology. As an example, we take the *MinCont* delay performance in Figure 4-5. Initially the mean delay increases while the number of nodes remains below 20. As the network becomes more densely populated the delay begins to decrease. When this link cost metric is used, routes tend to avoid densely packed regions in favour of less populated areas. Our model uses boundary edges and as such nodes cannot exist outside this boundary. In an infinite plane model, nodes outside this boundary which are not considered to be part of the network can interact and interfere with nodes close to the boundary edge. Because we do not use an infinite plane model paths will form towards the edge of the working plane, close to the boundary regions. As a result of this, the path from the multicast source to each destination node will be formed by multiple short links with a high link rate. From this, we can observe that the tree characterisation was a result of the link cost and the position of the multicast source node. We then identified four locations for the multicast source node which influenced the path characteristics.

In Figure 4-6 we illustrate the four key positions for the *Root* node on the working plane. The *Central* position paths will tend to follow a radial distribution regardless of the link cost metric used. The *Mid Central* position produces similar paths with shorter paths biased toward the near corner. The *Single Edge Boundary* position presents a different scenario due to our bounded working plane in that nothing exists directly behind this position. Therefore, paths will form toward the edge if minimal contention is a requirement or in a radial pattern for minimum hop count. From our analysis of these four starting locations we found the *Double Edge Boundary* placement to exhibit more challenging and interesting characteristics than the other three.

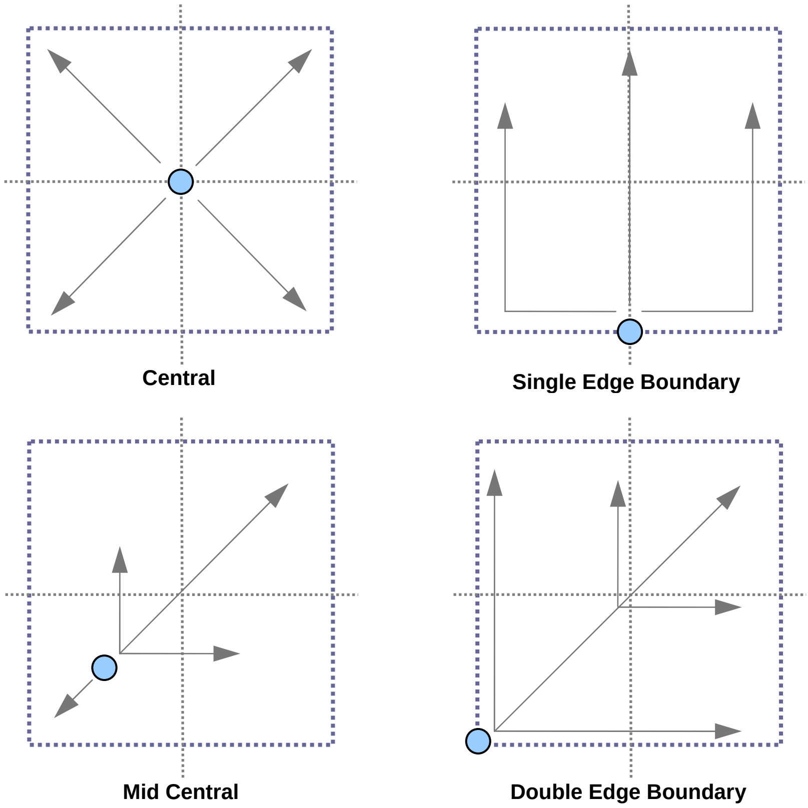


Figure 4-6: Four key locations of multicast source node, *Root*.

With two boundary edges paths we are presented with three main paths across the working plane; each of the edges and through the central diagonal. The central diagonal also presented us with the longest route across the working plane, (i.e. $\sqrt{2l^2}$, where l is the length of one side of the working plane).

Graphical Display of Network

So far we have concentrated on examples of analysing performance data of the network. As part of our simulations we have enabled a graphical visualisation of the network. By outputting a network constellation diagram we can quickly observe the configuration of the network tree. On closer inspection we can also observe how paths are formed from the *Root* node to each of the *Parent* and *Child* nodes in the hierarchy. In Figure 4-8 we can see the *Root* positioned on the double edge boundary. In the diagram link rates are colour coded as per the legend in Figure 4-7.

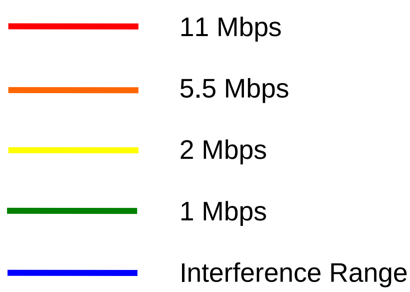


Figure 4-7: Legend for link rates as illustrated in Figure 4-8.

By positioning of the multicast source node like this, we can see a number of general routes taking place. This diagram illustrates the case when the minimum hop metric is used. The majority of the paths spread out diagonally away from the *Root* node and then work toward the edges. A network constellation diagram illustrating the use of a minimum contention link cost would show the opposite effect (i.e. in general, paths will form toward the edges and work inwards).

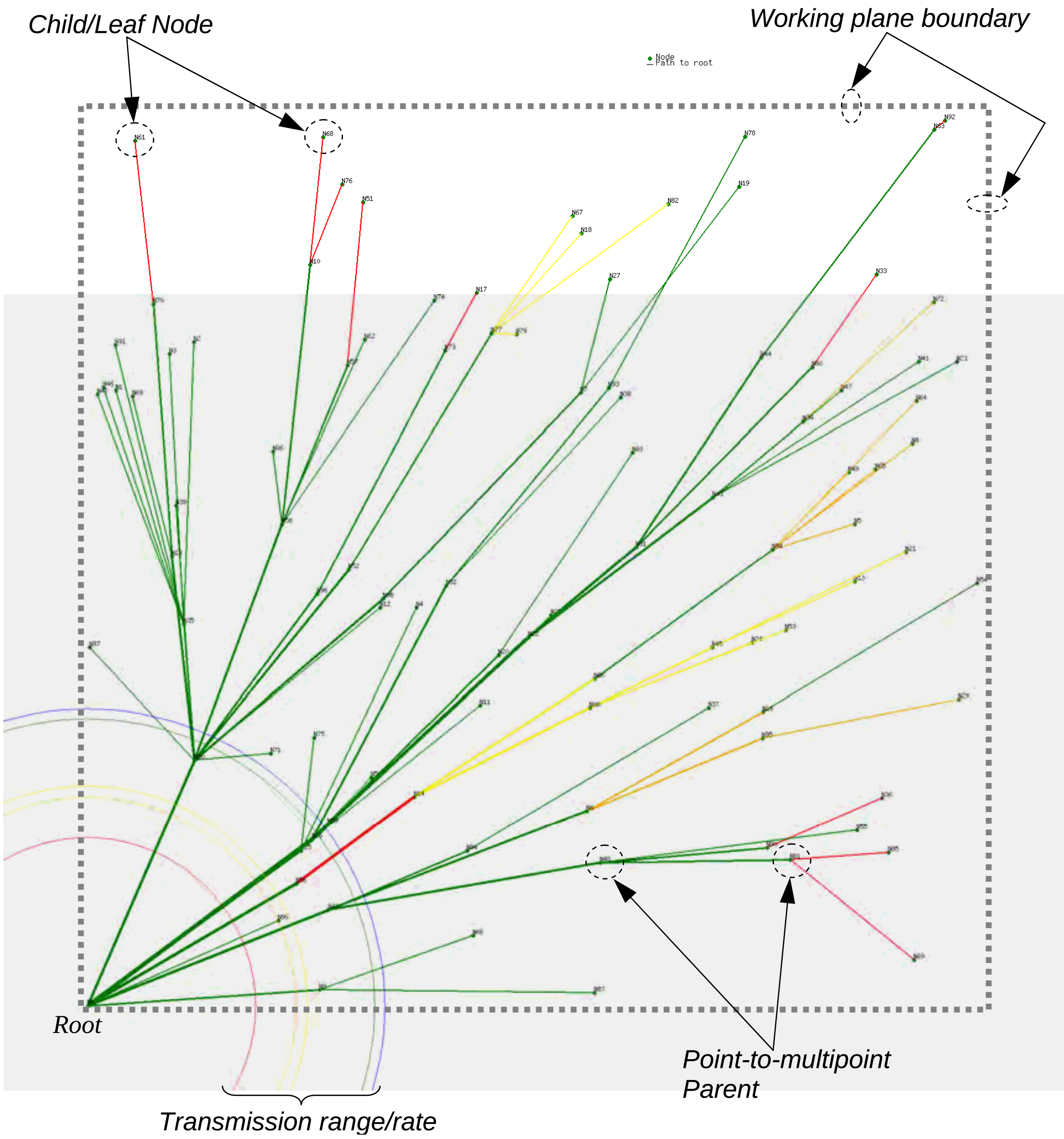


Figure 4-8: Graphical representation of the multicast network. The *Root* node is fixed at the double edge boundary position.

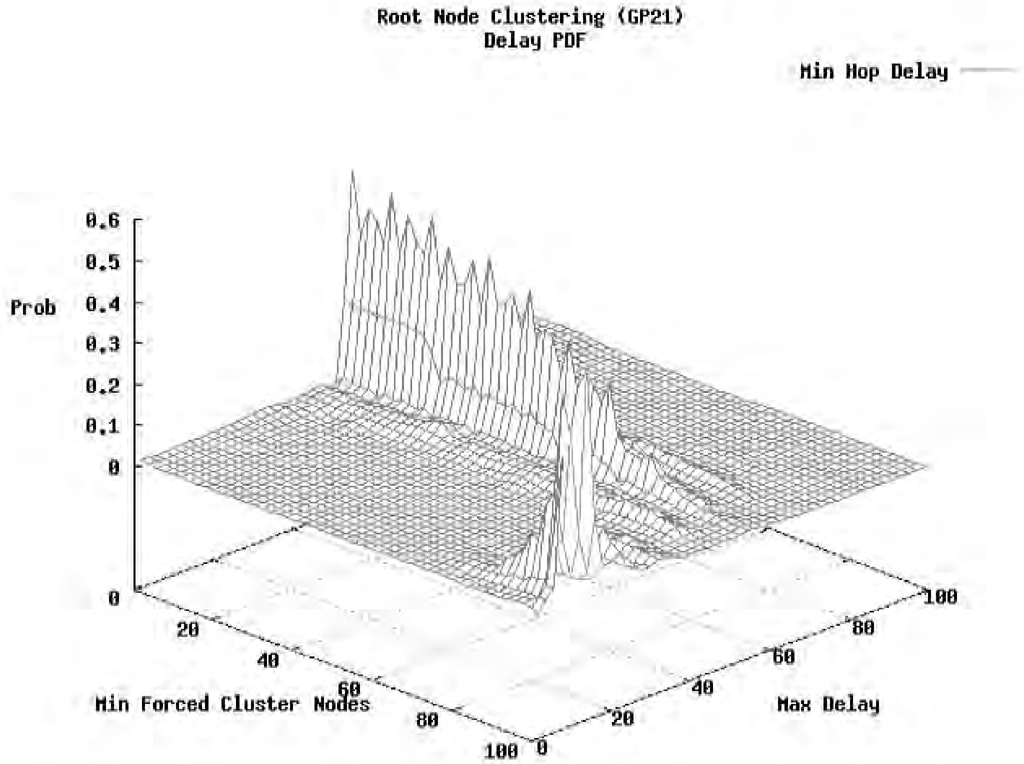
We can clearly see multicasting in operation through point-to-multipoint links all operating at the same rate (indicated by the same link rates/colours at each branch point). From Figure 4-8 we can identify two main general characteristics; that the majority of the link rates are 1 Mbps (coloured green) and that long point-to-point links are formed before branching into point-to-multipoint links. As a result of the multicast effect, forwarding nodes can only transmit once and will transmit at the lowest PHY rate. This means if one *Child* node is far from the source all other children will have to receive the transmission at this low rate. The long links are as a result of the link metric and are common in all our link metrics. Generally speaking, Dijkstra's shortest path algorithm will attempt to minimise the number of forwarding nodes which will (when using minimum hop for example) create long point-to-point links. These will also appear when using different metrics such as minimum contention. Although minimum contention will initially create links toward the edge, as stated previously, when the network becomes densely populated paths will form through the centre of the working plane to avoid traffic from edge nodes and will again result in long point-to-point links.

The use of constellation network diagrams afforded us valuable insights into the network operation when using the basic simulation model. By using such diagrams we were able to identify problematic regions of the network and thus presents us with interesting questions. For example: How do we eliminate or reduce poorly performing point-to-point links? How do we improve the performance of a *Child* node that suffers due to a poorly performing *Child* node of the same *Parent*? We will discuss these questions in further detail in the following sections 4.2 and 4.3.

Clustering

In addition to the previous questions, our simulation model presented us with many other interesting features regarding network behaviour. For instance, what are the effects of clustering in the network? By clustering we refer to a localised dense collective of nodes. We identified two conditions when this can be of significance. Random clustering on the working plane and clustering within interference range of the *Root* node. We give examples of clustering around the *Root* in Figure 4-9 and Figure 4-10. In order to analyse the effects of clustering the working plane was divided in a 5 x 5 equally spaced grid. Grid positions were numbered from left to right and top to bottom such that grid position 1 (GP1) is located at the top left corner. Nodes were distributed randomly across the working plane as before however grid positions were now biased toward taking a fixed number of nodes.

Results from all grid positions can be found in Appendix A. The figures show a 3-Dimensional plot of the probability density functions of delay when using the minimum hop and minimum distance link cost metrics. Below each plot is a contour map which helps illustrate the effects of clustering. We can see that initially trees constructed using minimum distance will have lower mean delays with a wider distribution of values compared to minimum hop. However, as the number of nodes within the communication range of the *Root* increases, both cases show similar results. This result is as expected and essentially shows us that once nodes lie within the communications range of the *Root* node, the weighting effects of different link costs have little effect as the node density increases.



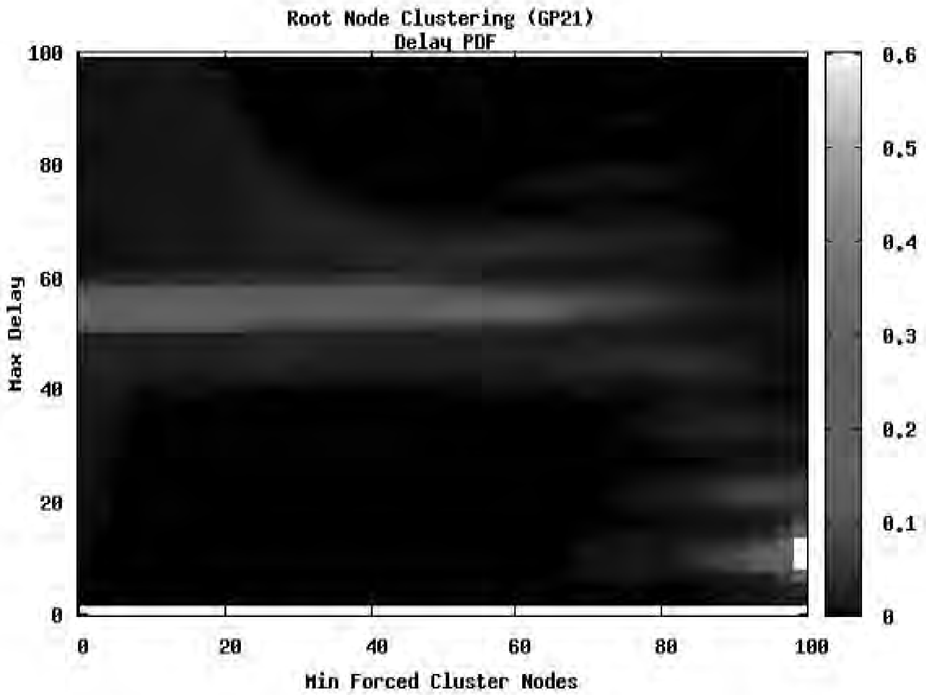
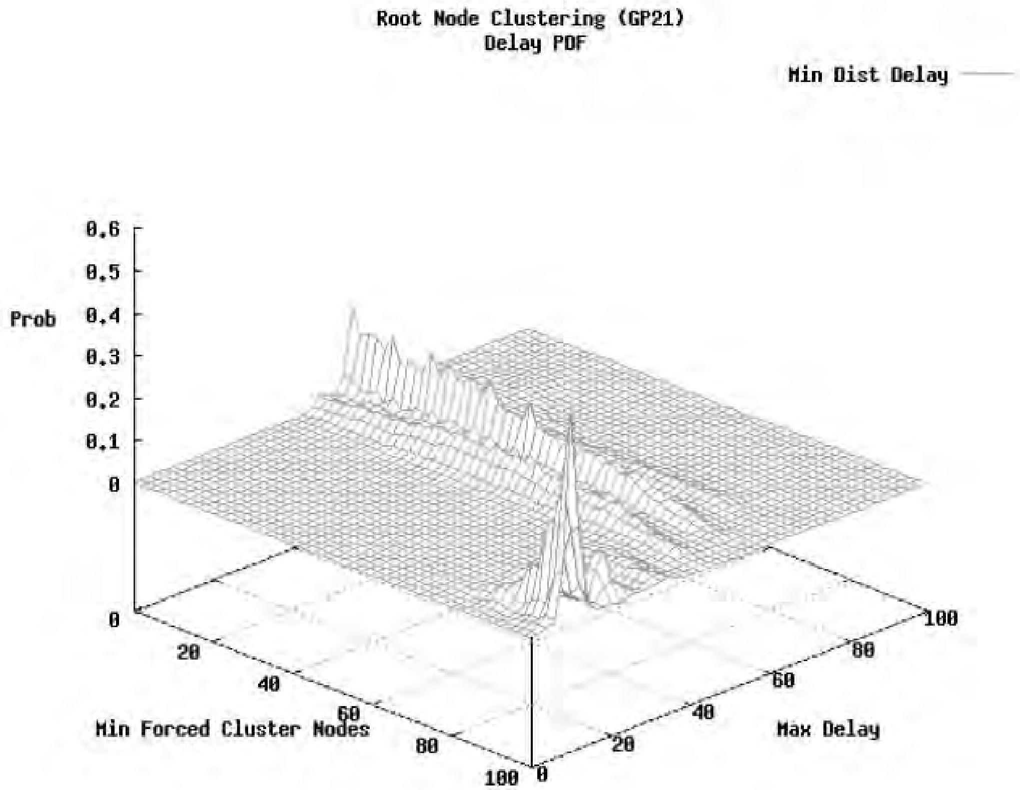


Figure 4-9: Delay characteristic of *MinHop* when clustering takes place at the *Root*.



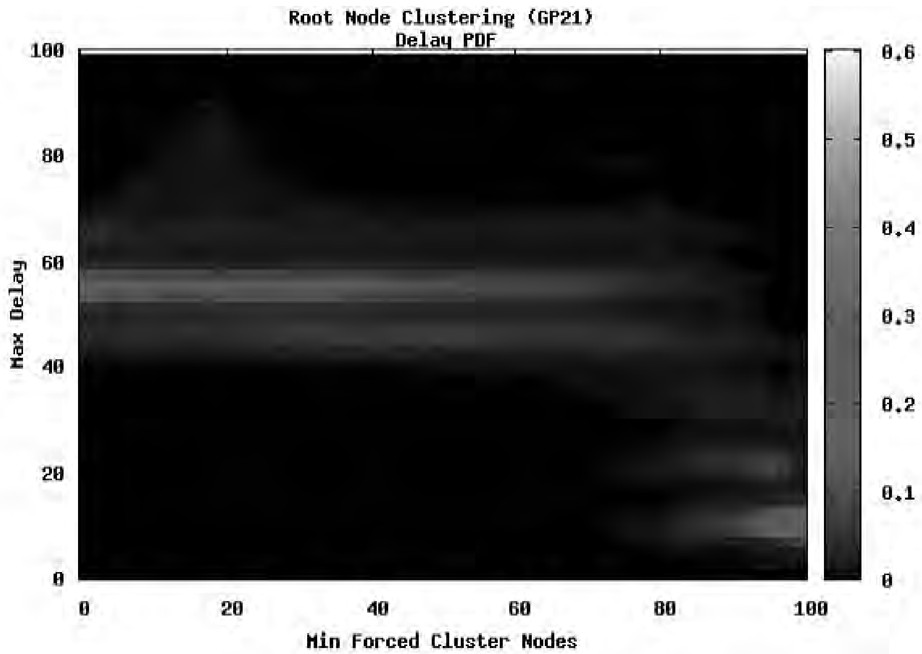


Figure 4-10: Delay characteristic of *MinDist* when clustering takes place at the *Root*

4.2 Midpoint Node Optimisation.

In the previous section (4.1) we discussed the operation of our basic simulation model. Through analysis of our simulations output we ask the questions:

- *How do we eliminate or reduce poorly performing point-to-point links?*
- *How do we improve the performance of a Child node who suffers due to a poorly performing Child node of the same Parent?*

We have shown that the existence of long links can cause poorly performing point-to-point and point-to-multipoint links through low transmission rates. This occurs as a result of at least one neighbouring node operating at the edge of the communication range and/or due to the multicast operation. Furthermore, we noted in section 3.7 how nodes operating at lower transmission rates than their neighbours can have an adverse effect on neighbours operating at higher rates. Therefore, we recognise that there is a requirement to either eliminate or reduce the effects of such nodes. From our study of spanning trees in section 2.5 we introduce the concept of minimising the overall cost of a spanning tree by introducing additional nodes. In wireless networks this essentially translates into adding relay nodes such that each relay node will only forward traffic to a specific location and will not take part in the same duties as other nodes (e.g. it will not operate as a multicast *Parent*).

In Figure 4-11 we illustrate two of the problematic instances. Links *A* and *B* present us with situations where we observe low transmission rates. In both instances we say that the link rate of *A* and *B* is lower than the link rate between neighbouring nodes along the same path. In the first instance the overall path rate will be affected due to link *A*. In the second instance, the effects are increased due to the *Parent* node transmitting to all *Child* nodes at the lower rate of link *B.*

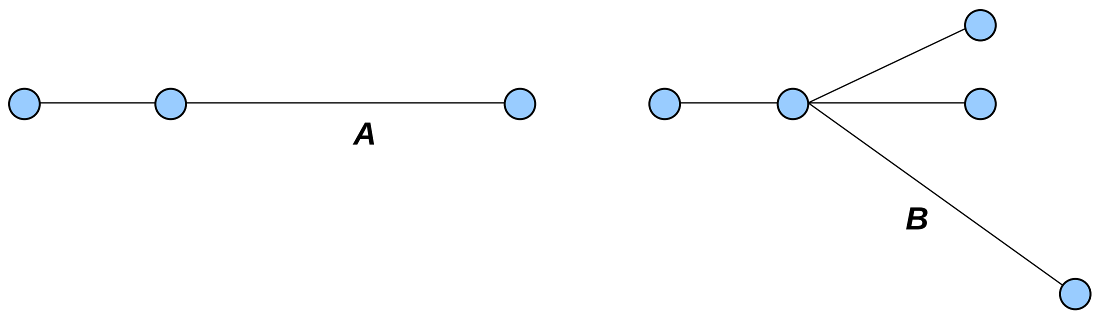


Figure 4-11: Example of influential point-to-point/multipoint links.

The first solution we put forward is to clearly identify such instances, as illustrated in Figure 4-11, and to minimise their effects by introducing relay nodes in order to improve the path rate and overall tree performance. In Figure 4-12 we illustrate the basic idea of our technique.

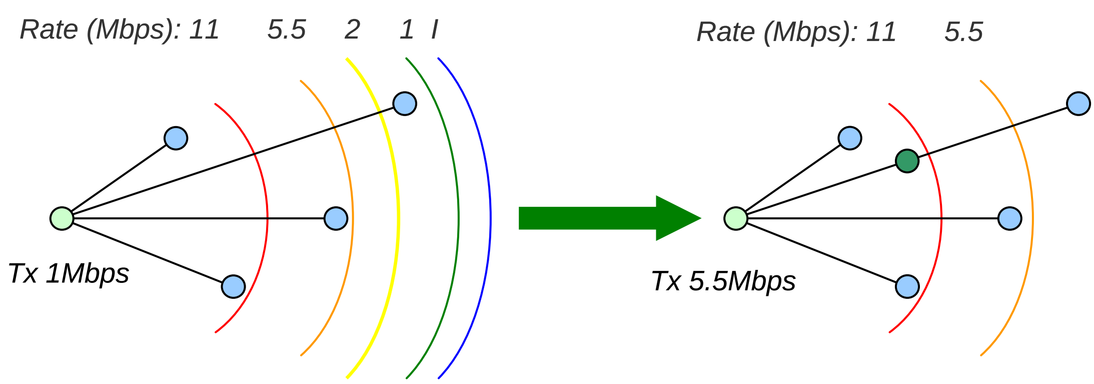


Figure 4-12: Introducing a relay node to improve multicast path rate.

4.2.1 Optimisation Criteria

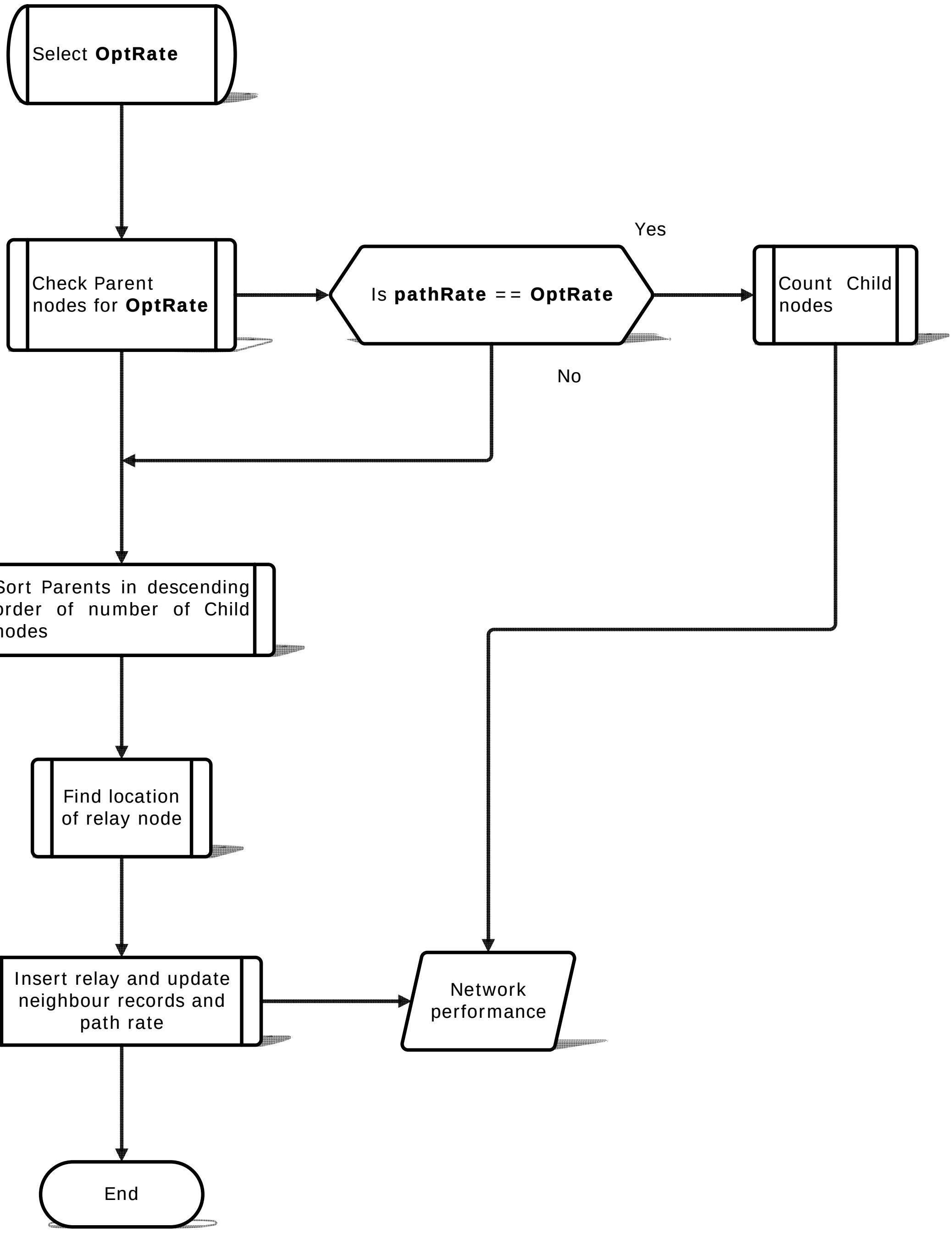


Figure 4-13: Criteria for relay node placement.

The first step in introducing relay nodes is to identify the links which require additional nodes. We assume that the locations of all nodes in the original budget are fixed and known after deployment. After the neighbour discovery has taken place for each topology we set the path to the *Root* node and set the multicast path rates. In order to set the multicast path rate we must also determine the maximum link rate of each node. A record of the maximum link rate is stored along with link costs in our node structures. We are then able to search through the structures in order to determine which links have low maximum link rate. At the same time our record structures allow us to determine if the node is a *Parent* or a *Leaf* node and how many *Child* nodes it has. Figure 4-13 illustrates the criteria for relay node placement.

With this information at hand we have a number of factors to consider in order to determine how we should proceed;

- How many relays should we add in total?
- What is considered a poor link rate? (i.e. do we stop at 1Mbps or should we continue up until the maximum PHY rate)
- How do we decide which links will receive relay nodes?

The first two questions are related to the available allocation of relay nodes. Although through simulation this does not pose any difficulty, it is an important consideration for practical deployments. In relation to the last question we will assume we have an infinite allocation of relay nodes and can therefore attempt to place relay nodes on all links. The order in which we will do this is as follows;

- *Parents* with a minimum multicast rate will be considered first.
- *Parents* with a minimum multicast rate will be dealt with in descending order of the number of *Child* nodes.

4.2.2 Updating the Node and Neighbour Lists

After each new relay node is placed we will update the neighbour records. Although each new relay node will appear in the neighbour list it will not be allowed to perform multicast functions. The primary function of the relay nodes is to act as an intermediary node and forward point-to-point traffic. A record of the network performance is taken after each new node is inserted into the network. We stop the algorithm when there are no more links remaining which require a relay node. The performance of the network is then analysed for all additional relay nodes.

In keeping with the simplicity and generality of the model we assume that all relay nodes will use omni directional antennas. We acknowledge that for point-to-point communications the use of directional antennas would be better suited. It should be noted that we can adapt our technique to allow for a fixed budget of relay nodes or to set the limiting multicast rate.

4.3 Power Optimisation.

In the previous section we discussed a method of physically adding nodes to an existing network in order to alleviate problematic aspects on the network. The source of the problems is due to low link rate bottlenecks or a low link rate *Child* node in a multicast branch. In our discussion we assume that the distribution of nodes is homogenous with the exception of relay nodes which have only point-to-point forwarding abilities. We now present a different technique for altering the topology of the network through adjusting the transmit power of each node.

The basic concept here is to adjust the transmit power of a node so as to operate within the receiver sensitivity thresholds. Through the use of this power adaptation mechanism it is our intention is to increase the PHY link rate through an improved link performance. The obvious advantage of this method is that no additional equipment is required for a given budget. However, fine tuning a network in such a manner requires a feedback mechanism which can prove to be quite difficult due to a number of external factors impacting on the ambient noise level. In keeping with our design methodology we will show how to develop a simple model and build upon its complexity to allow further development and analysis.

Each manufacturer of wireless networking equipment designs the radio receiver with sensitivity threshold levels. The receiver sensitivity thresholds will determine the minimum signal power that will result in a wireless frame being successfully received. Typical receiver sensitivity thresholds can be seen in [LiJ07], [NWa10] and [WMB06]. Table 4-2 indicates the receiver (Rx) sensitivity settings used in our simulator. The table also indicates the typical transmitter (Tx) ranges associated with each rate when a transmit power, P_{Tx} = 9 dBm and an attenuation factor, γ = 3 is used (typical values for γ are generally between 2 and 5 with 5 indicating strong attenuation). Receivers will have a specified sensitivity range which at one end will allow them to receive transmissions at the highest rate (based on the modulation scheme) whilst at the other end will allow them to receive interference from neighbouring transmissions. The transmission rate is based on Packet Error Rate (PER) which is related to the signal strength which in turn is related to the distance between nodes, resource utilisation and channel error model. By adjusting the signal strength we can increase the transmission range as well as the transmission rate through an improved link performance.

Rx Sensitivity	**Tx Rate**	**Tx Range (P_{Tx} = 9 dBm, γ = 3)**
- 95 dBm	-	134m
- 94 dBm	1 Mbps	124m
- 91 dBm	2 Mbps	99m
- 87 dBm	5.5 Mbps	73m
- 82 dBm	11 Mbps	49m

Table 4-2: Simulation settings for receiver sensitivity levels.

Using a free space loss (FSL) model and the same attenuation factor as before, we obtain the range versus rate characteristics as shown in Figure 4-14. The diagram shows the transmit power P_{Tx} characteristic for a repeated doubling of the power from 0 dBm to 18 dBm (i.e. 1 mW to 63 mW). We assume a maximum transmit power of 18 dBm and the ability to adjust the power in increments of 0.5 dBm according to [KBK08a] and [KBK08b]. For the purpose of our simulation we will limit tuning granularity to steps of 1 dBm in order to reduce the number of iterations.

In Figure 4-14 the vertical lines indicate the receiver sensitivity thresholds. By adjusting the transmit power we intend to change the range/rate characteristic of our network topology. We can for example simply increase the power level of all nodes or a selection of nodes (e.g. *Leaf* nodes) in order to maximise coverage.

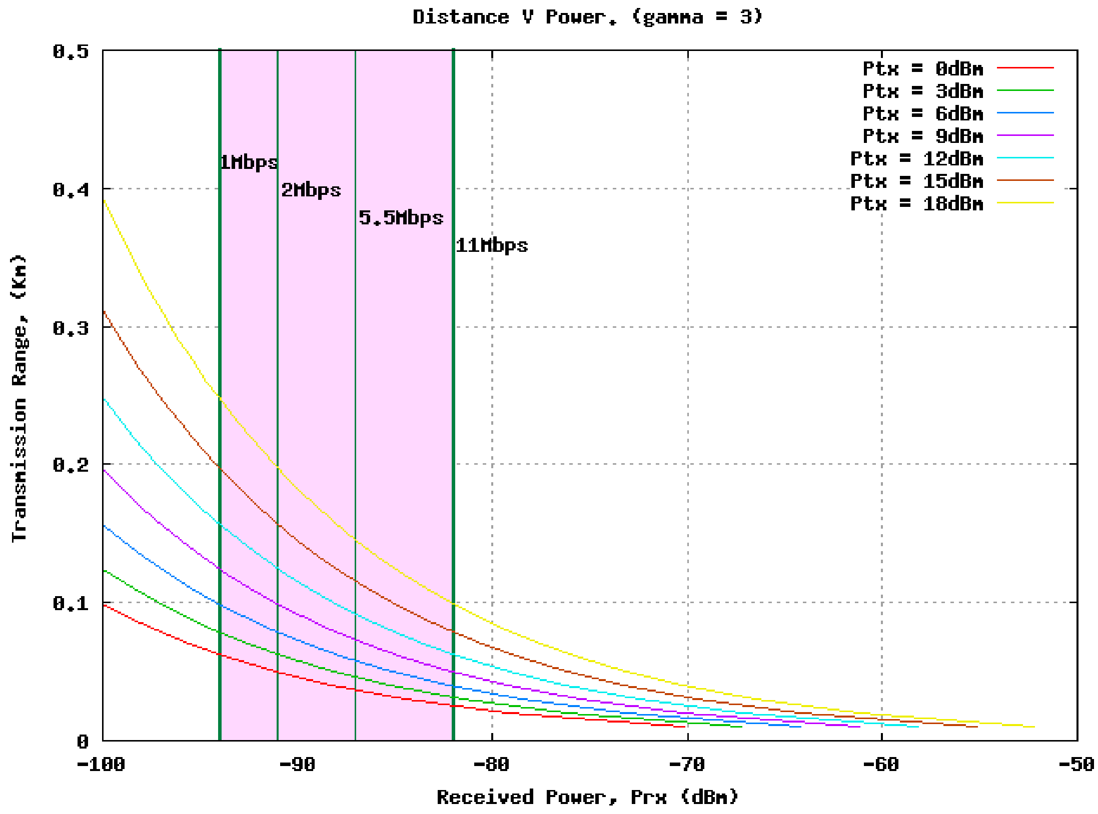


Figure 4-14: Range vs Power plot for 802.11b using FSL model.

However, we now focus on adjusting the power to eliminate slow transmission links as outlined in the previous section. For example, if a *Child* node is positioned a fixed distance from its *Parent*, we take advantage of the relationship between the distance and received power as described in section 2.9. On identifying a slow link we can then increase the transmit power of the *Parent* sufficiently to allow for a change in the modulation scheme and hence the transmission rate. Conversely we can also reduce the transmit power where necessary to reduce the effects of interference with neighbouring nodes.

In Figure 4-15 we illustrate how the multicast rate to a group of *Child* nodes is initially 1 Mbps due to the proximity of the furthest *Child* node. By increasing P_{Tx} we increase the transmit range and allow a higher rate modulation scheme to be used for this node. This will enable a higher multicast rate to all *Child* nodes connected to the *Parent*.

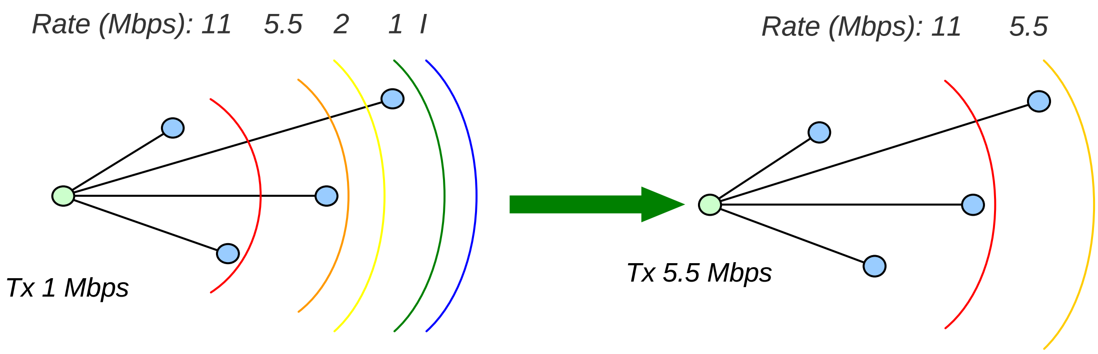


Figure 4-15: Increasing P_{Tx} to increase the Tx rate. A multicast rate is increased from 1 Mbps to 5.5 Mbps.

4.3.1 Power Control Algorithm

We adapted the basic simulator model to incorporate our power control algorithm. In Figure 4-16 we illustrate the operation of the power adaptation. After a topology has been generated and shortest paths have been set we invoke the power control scheme.

We first initialise our starting parameters, such as the receiver sensitivity values, attenuation factor, channel frequency, transmit power and transmission range. We then define a maximum transmission rate *optRate* that we wish to achieve before getting the path transmission rates, *pathRate* of all *Parents*. The *pathRate* will initially not be set and will only be defined after initial paths have been selected.

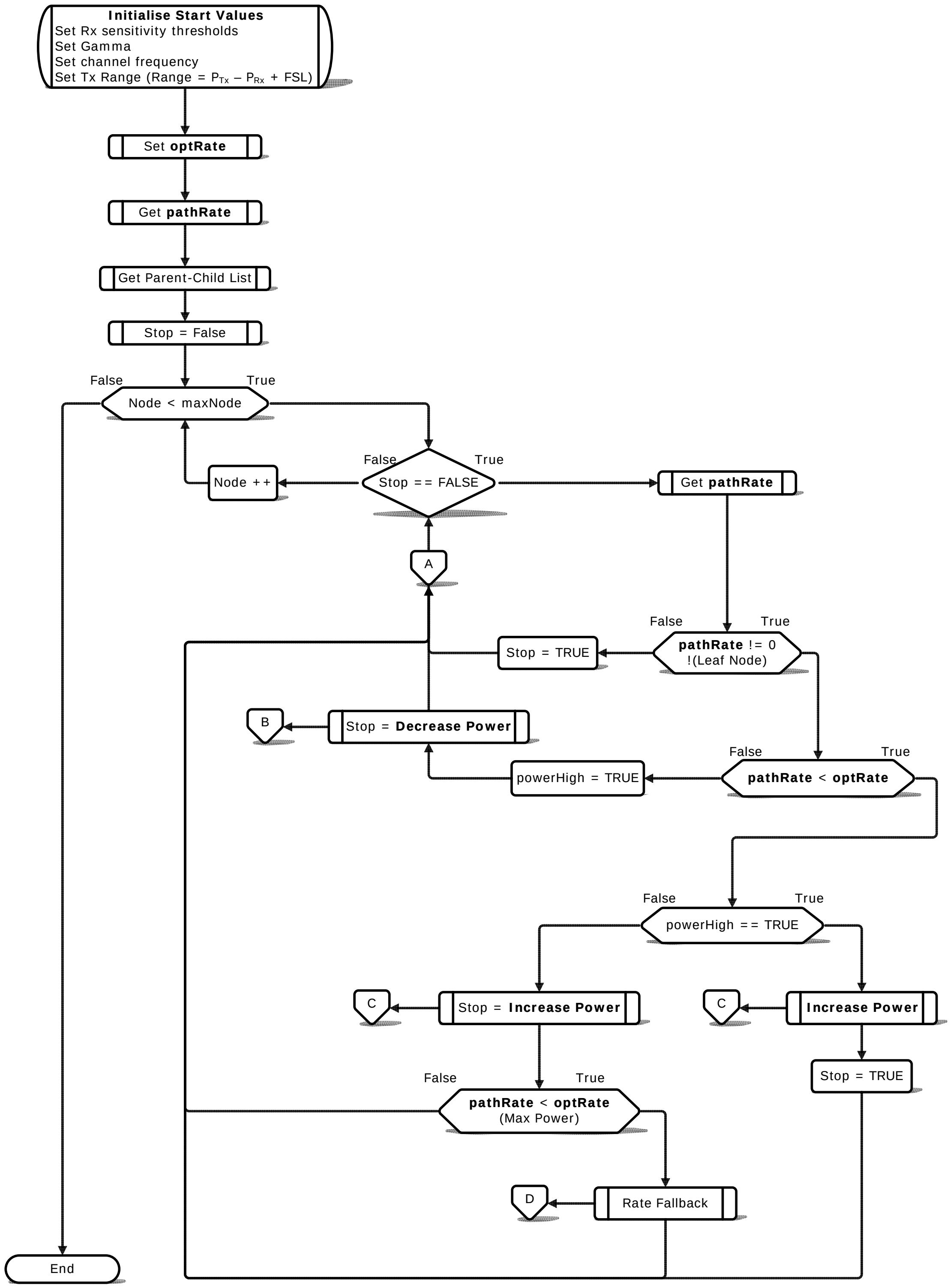


Figure 4-16: Flow chart for power adaptation algorithm.

A condition we have set on the algorithm is to maintain the minimum tree hierarchy between *Parent* and *Child* nodes. This means that the same *Parent-Child* relationship must be maintained after the power has been decreased to ensure the same level of connectivity. This constraint enables a comparative analysis of the network before and after the algorithm has been applied while maintaining a minimum level of connectivity.

With this condition set we must then create a list of *Parent-Child* node relationships. Once this is in place we search through each node to establish if it is a *Parent* node and determine how to best adjust its transmit power. Our stopping condition will be set when a maximum or minimum power level has been reached.

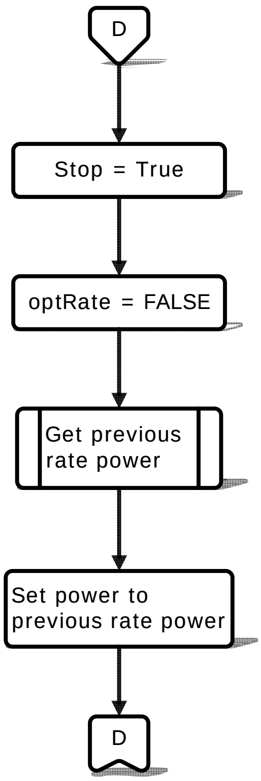


Figure 4-17: Rate Fallback ensures the previous working state is maintained if connectivity is lost or maximum power is reached

After determining that a node is a *Parent* (i.e. it is not a non-forwarding *Leaf* node) we check to see if its multicast transmission rate, *pathRate* is set to the *optRate*. If the transmission rate is already at the maximum rate optRate then we allow the *Parent* node to decrease its power in order to reduce the interference to neighbouring nodes. If after decreasing the transmission power, connectivity is lost to one or more *Child* nodes, we implement a power fallback scheme. The power fallback scheme ensures that the previous working state is re-applied. If the rate is not at the maximum transmission rate *optRate*, then we increase the transmit power. We continue to increase the transmit power until either *optRate* is achieved or until the maximum power level has been reached. If we find that the *optRate* has not been reached after the maximum power has been set then we implement a rate fallback (see Figure 4-17). The rate fallback mechanism allows us to reduce the power to the setting for the next highest transmission rate.

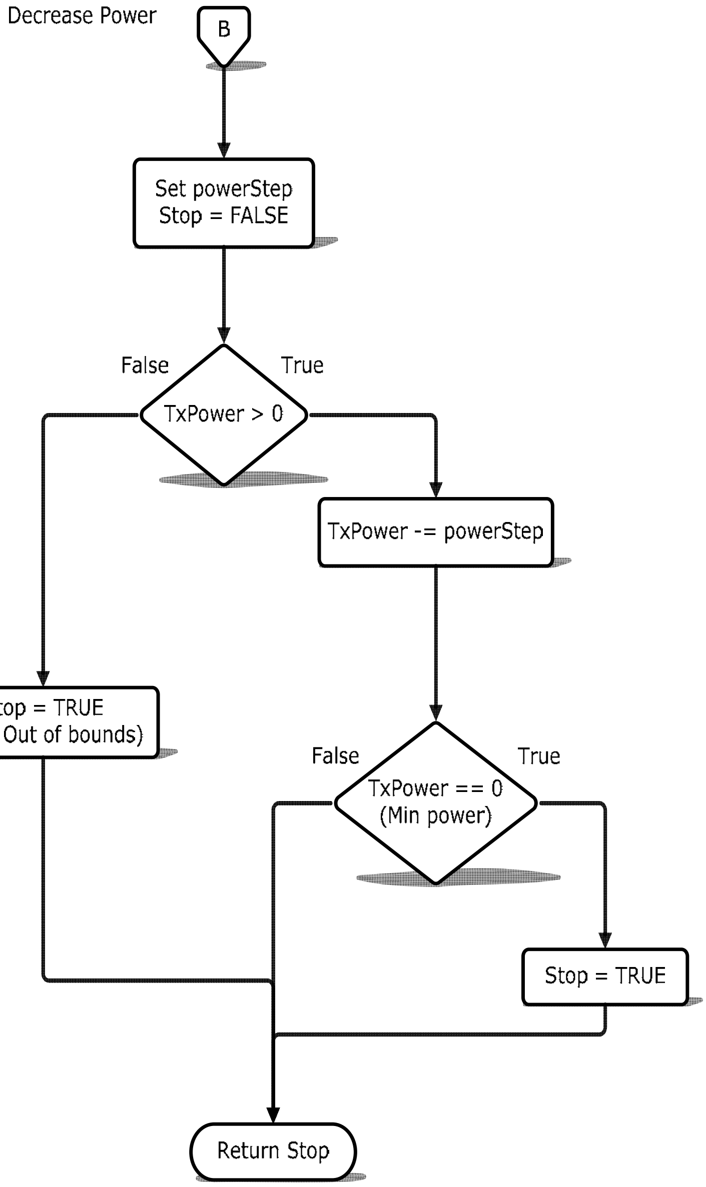


Figure 4-18: Decrease Power function with out of bounds error checks.

In Figure 4-18 we illustrate the operation of decreased power control. We include error checking in the flow chart to ensure out of bounds operation is not possible (i.e. the transmit power will not operate outside the power settings of 0 dBm – 18 dBm). The operation of increased power control works similarly and can be seen in Figure 4-19.

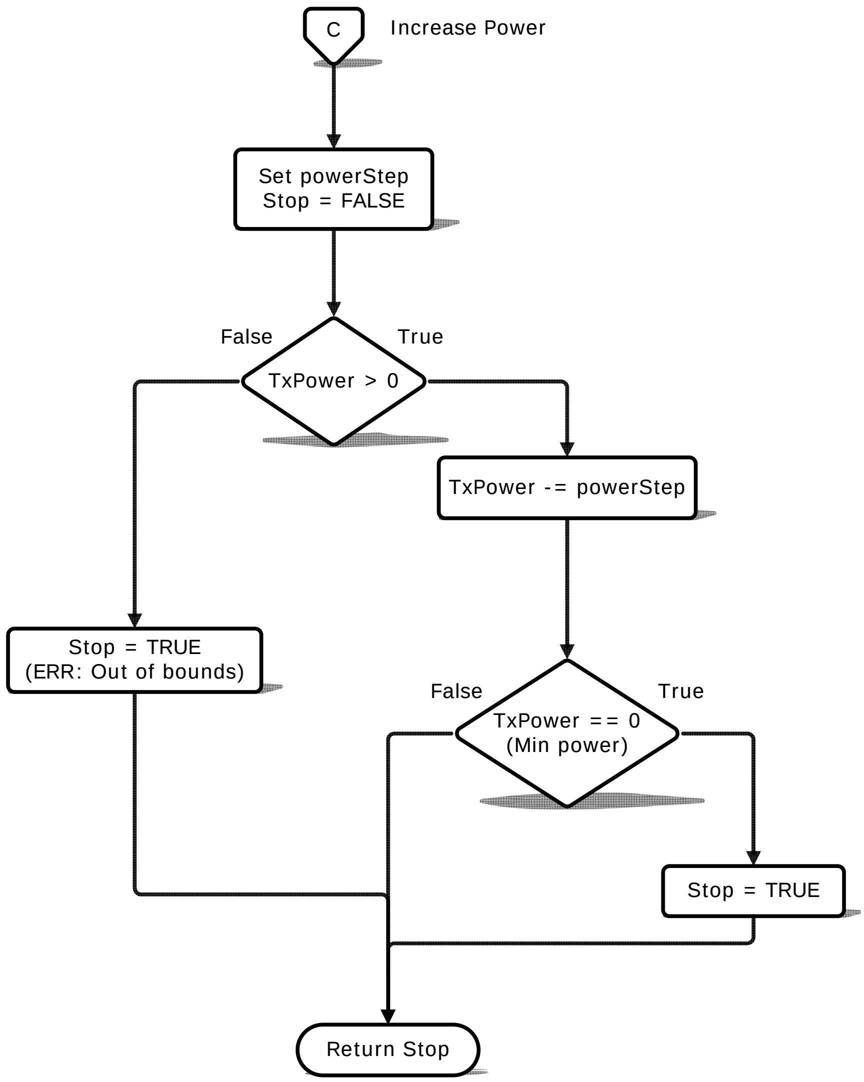


Figure 4-19: Increase power control flow chart.

Once the power has been changed (through either an increase or a decrease) it is necessary to update the neighbour list information again due to changes in the received power levels. At this stage we also check to see if the *Parent-Child* relationship has been maintained and implement the power fallback scheme if necessary. Figure 4-20 illustrates the operation of this procedure.

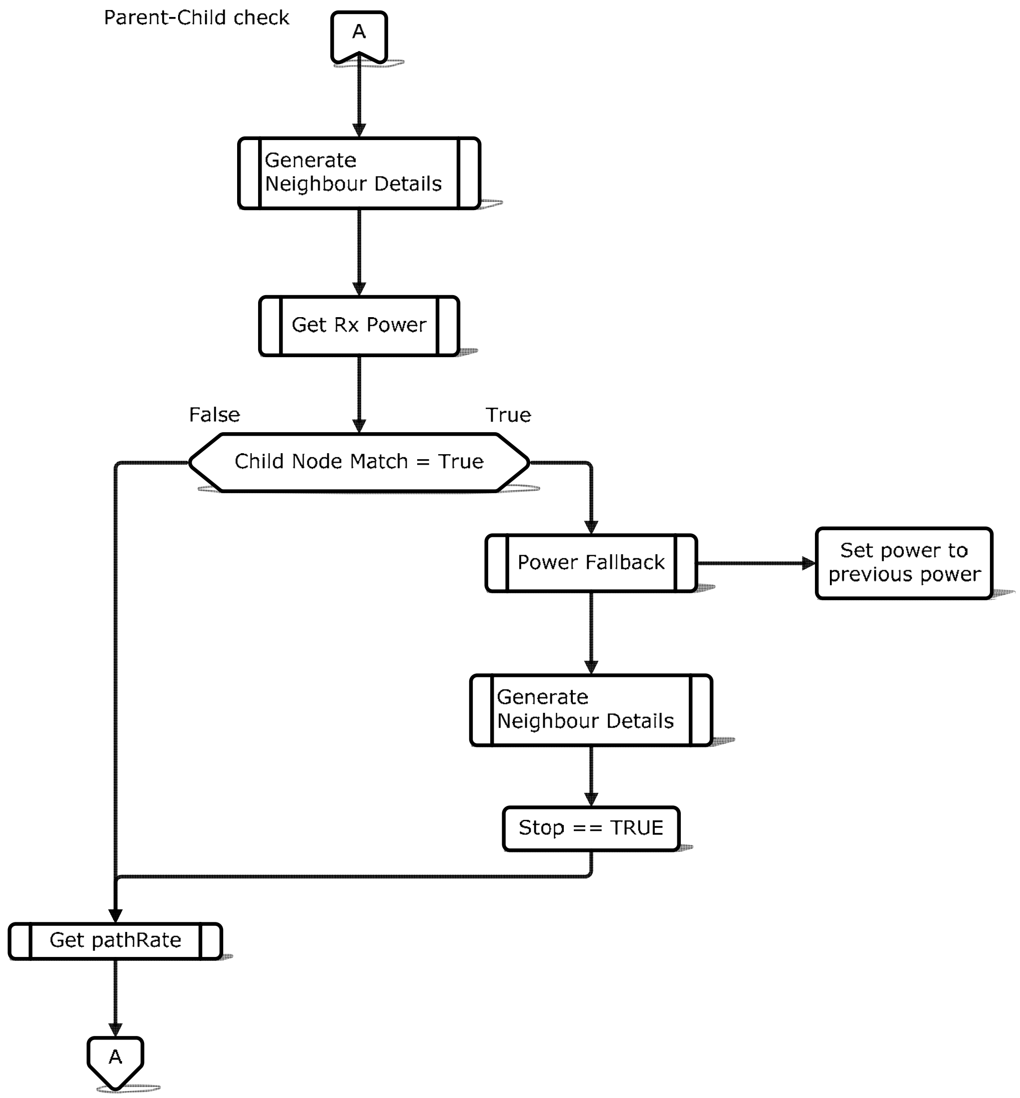


Figure 4-20: Power fallback operates if *Parent-Child* node relationship is not maintained.

Our power adaptation algorithm attempts to adjust the power of each *Parent* in order to increase the transmission rate or to reduce the neighbour interference. By searching through each node our technique is essentially a greedy path optimisation. The algorithm does not explicitly look to improve the transmission rate of the entire tree. However, it does implicitly attempt to achieve an overall tree improvement by decreasing the transmit power if an increase does not result in a path gain.

4.4 Optimal Spanning Tree

In the previous section we discussed our method of adjusting the power of each *Parent* node. The implementation of the algorithm maintains the tree hierarchy and attempts to achieve performance gains on a path by path basis. However, it is possible that within a given topology, after adjusting the power of each *Parent*, there exists a better performing tree. In other words, if Dijkstra's shortest path algorithm was run again, after the power was adjusted, would new routes be formed with a better tree performance? Furthermore, could this new tree be fine tuned with another pass of the power control algorithm? Running Dijkstra's shortest path algorithm will find the optimal paths in the network. However, because new paths are formed the neighbour dynamics will have changed. Nodes which previously could not receive transmissions now come into interference or communications range. When a distant node initially comes into communications range it will have a poor received power (depending on how much the power was increased) and as such will be the cause of a low multicast rate to the *Child* nodes within its group. Any gains from the new paths formed by running Dijkstra's shortest path algorithm again will then be negated by these effects. However, running the power adaptation algorithm again enables us to provide performance enhancements (e.g. a node which was previously a *Leaf* node with a fixed power will benefit from a power adjustment).

In order to determine this outcome, an exhaustive enumeration method (i.e. brute force) can be performed to search for all possible outcomes. This type of search technique can be time intensive and will depend on the number of simulations. In section 2.8 we discussed several search techniques for global searches. One such technique is known as Simulated Annealing. We have modified the approach taken by the simulated annealing technique to search for high performing multicast trees after a power control mechanism and Dijkstra's shortest path algorithm is applied.

We will run successive implementations of our power adaptation algorithm followed by Dijkstra's shortest path algorithm. With each iteration of both algorithms we calculate the mean throughput for the tree. Successive iterations will search for a maximum tree throughput. If the tree performance is greater than the maximum throughput, the tree is accepted and a copy of the topology details (i.e. each node remembers its power settings and *Child* nodes) is stored. If the tree performance is less than the maximum then our search algorithm will decide, based on an exponential probability function (Figure 4-21), whether or not to accept this new state and continue, or to move to a new state.

The probability function is related to the change in throughput and the number of searches already performed:

$$Probability\ Function > rand\,(0..1) \tag{4.13}$$

We define the $Probability\ Function = e^{-\left(\Delta TP/T\right)}$ (4.14)

where ΔTP is the change in throughput from moving from one state to a new state. T is the cooling process to determine the number of steps taken leading to a higher mean tree throughput. If T is large, more states leading to a fall in throughput will be accepted.

$$T = {}^{k}\!/_{k\max} \tag{4.15}$$

where k is the current iteration and $kmax$ is the upper limit. If a better throughput is found both k and $kmax$ are reset to initial values. If the probability function is successful (i.e. it produces a value greater than *rand (0..1)*) then the value $kmax$ is increased in order to favour continued searches. We can invert the ratio $k/kmax$ in order to favour less searches (i.e. faster but less accurate search). The basic idea being that, if the difference in throughput is relatively small then a maximum is close and the number of potential iterations will be reduced. If the difference is relatively large then we favour additional searches. As the search progresses we also increase the chance of carrying out an additional search by increasing the value of k.

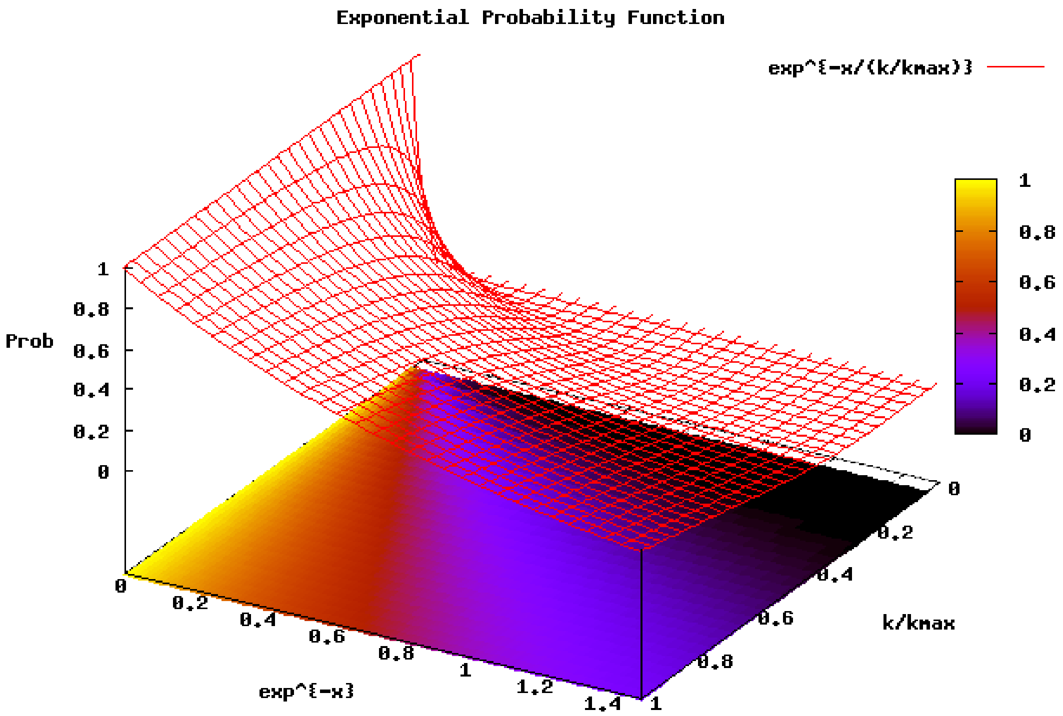


Figure 4-21: Probability function used to decide if successive searches will be carried out.

In Figure 4-21 we can see that if the difference in throughput is relatively small then we allow a high probability of acceptance (i.e. a maximum throughput is close). As the search algorithm moves through successive iterations the ratio *k/kmax* increases which allows for further states to be explored (i.e. the search did not quickly find a maximum but instead has progressively moved to better states).

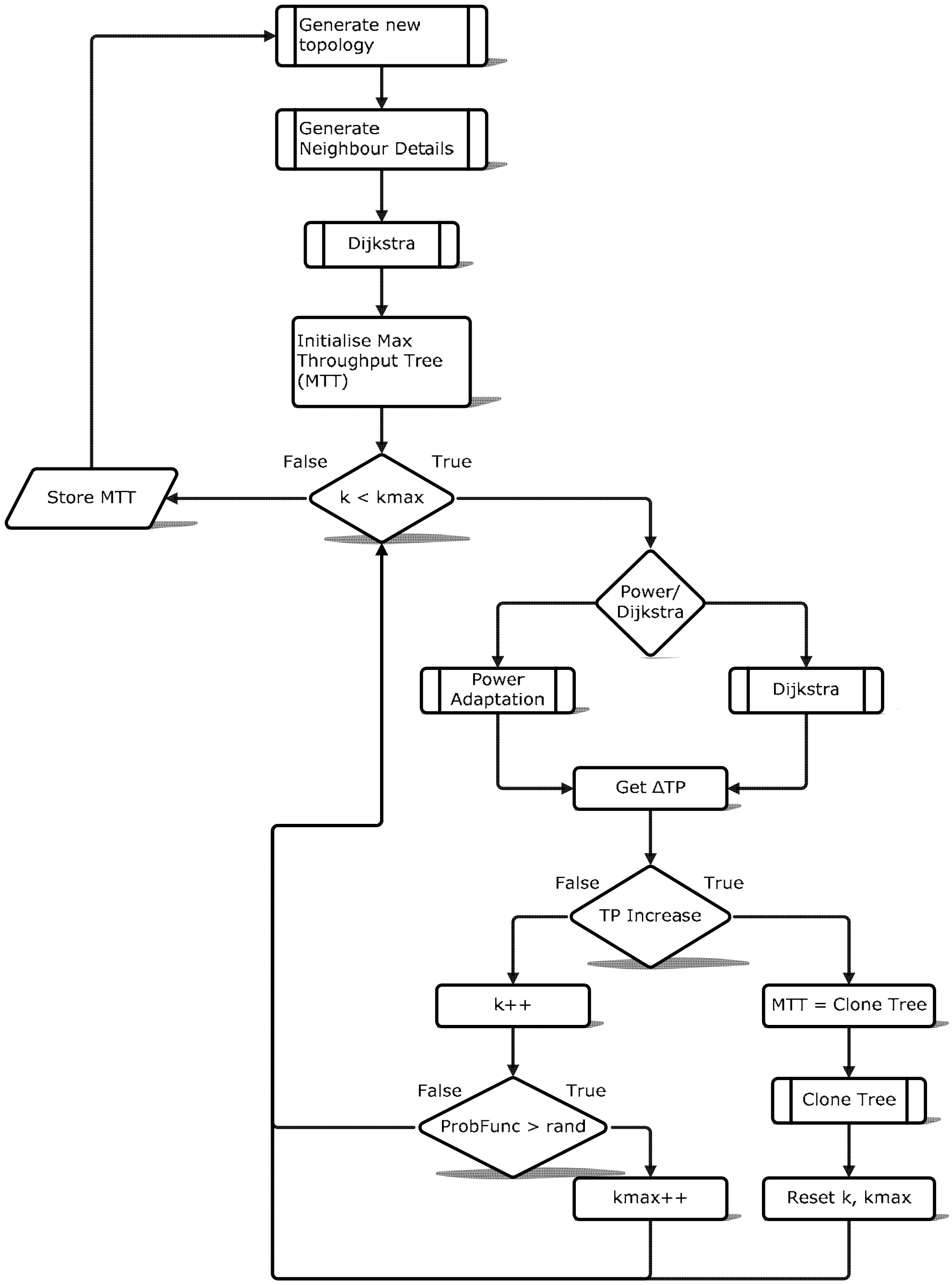


Figure 4-22: Flow chart illustrating search for maximum throughput tree.

Our search technique operates by assuming it is worthwhile to search a path of lower gain with the expectation of eventually finding a better solution. We also reduce the possibility of getting stuck in a local maximum by ensuring we do not search the same solution twice. Figure 4-22 illustrates the operation of the search algorithm. The simulation takes place as before except this time we loop through multiple iterations of the power adaptation and Dijkstra's shortest path. We can see from the diagram that if an increase in throughput is detected a cloned copy (or deep copy) of the structured node records is taken. This cloned copy is now designated as the best solution found in the search. We then reset the variables, k and $kmax$ in favour of prolonged search. If a lower throughput is detected we increase k to reduce the search steps before using the probability function to determine if additional lower states should be accepted.

4.5 Post Capture Analysis

A set of simulations run over 19 power settings (i.e. from 0 dBm – 18 dBm in steps of 1 dBm), with the maximum tree throughput search in operation, can amount to approximately 2 – 7 million unique network configurations. This is based on 1000 topologies for 15 node densities employing between 7 to 25 iterations of the optimisation technique. The output from our simulations is processed after completion in order to generate statistical data. We compile a number of plots based on the performance of the network.

As we are considering changes to the network topology we must consider topological effects such as the coverage. We measure the coverage of the network by carrying out a Monte Carlo sampling for each of our PHY rates. The percentage coverage is carried out as illustrated in Figure 4-23. In addition to this our results will include probability density functions of key performance metrics.

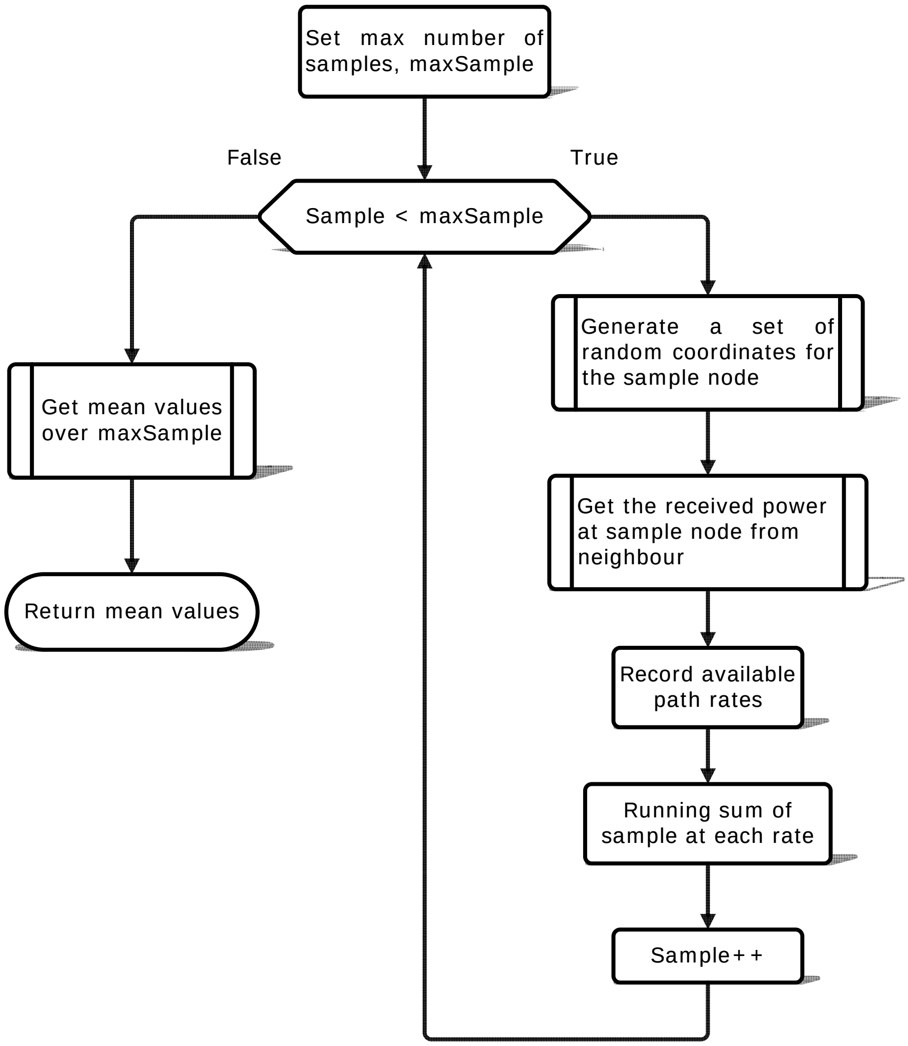


Figure 4-23: Calculating the percentage coverage for each rate using sampling.

4.6 Chapter Summary

In this chapter we have provided details of the simulation models used starting with the Basic Model. The Basic Model carries out all of the core operations of multicasting in WMNs. As part of our analysis we generate a graphical representation of the mesh network in order to identify key areas which can be improved upon. We use Dijkstra’s shortest path algorithm using various link cost metrics for comparison. Furthermore we thoroughly investigated the placement of the *Root* node and examined its effects on the performance of the network. By placing the *Root* node in a *Double Edge Boundary* position we present a more challenging network scenario.

This chapter provides a detailed description of the operation of two algorithms intended to improve the performance of multicasting communications over WMNs. The first method aims to identify positions in the network topology which can benefit from the placement of additional relay nodes. The second method describes how we can tune the transmit power on a per node basis in order to improve the transmission rate (and hence improve the mean network throughput). In order to find an optimal tree we describe a search technique to identify a network with maximum mean network throughput.

Our simulation provides a well defined characterisation of the network topology under various node densities and transmit powers. We clearly state the network and simulation assumptions made at each stage. As our method provides abstract simulations our aim is to provide a proof of concept. We use a Free Space Loss channel model for simplicity of simulation. However, the simulation model can be adapted to include more sophisticated channel models.

5 Results and Analysis

In the following sections we present the main findings of our simulation results for multicasting over WMNs. We first describe results from the Basic Model by comparing the performance of various link cost metrics. We present a selection of results which best describe the general performance characteristics of the network simulations. It should be noted that each set of results involved running hundreds of thousands of simulations (over 500,000 for each link cost metric) carried out using Perl v5.8.8 running Fedora Core 10 (kernel version 2.6.27.15) on a Dell E2200 Dual Core 2.2GHz desktop. We present mean values (for the performance metrics such as relative max delay, throughput, coverage etc) to describe the performance and provide additional detail in the form of probability density function (PDF) distributions where necessary. A complete set of results for all relevant simulations is provided in Appendix A.

After the Basic Model results have been presented we provide results for two optimisation techniques by using the algorithms described in sections 4.2 and 4.3. We will describe the outcome of our results for both techniques using performance metrics as described in section 4.1.4. Again, hundreds of thousands of simulations are performed (between 2 and 7 million in the case of the power optimisation algorithm). We also again present mean value plots (for performance metrics such as relative max delay, throughput, coverage etc) to describe the performance of the network with PDF plots used to provide additional detail as necessary. A complete set of results for all relevant simulations is provided in Appendix C and D.

5.1 Un-optimised Results

In section 4.1 we discussed the operation of what we termed our basic simulation model. We refer to the model as basic as it does not introduce any optimisation techniques. However, it does handle all of the core functionality of the multicast simulation process. As part of our design process we look at the effects of introducing new parameters or changing existing ones in our simulation. This is a key point in the development of our model. One of our design goals was to carry out a performance evaluation at various stages and to ascertain how best to proceed. By working with the Basic Model we are able to evaluate the performance of our network as the node density increases.

In this section we will evaluate the performance of the network for various link cost metrics. We will see in later sections that although a particular link cost metric performs well in the Basic Model it will not necessarily be suitable for use when an optimisation technique is introduced. In the following sections our simulation set-up is as follows;

Working Plane: The working plane for the each simulation was fixed at 650m x 650m (i.e. an area of 422.5 x 10^3 m^2). The boundary edges of the plane were open (i.e. no reflections) and all nodes were randomly distributed within the working plane.

Channel Model: The Free Space Loss (FSL) channel model was used for each simulation as described in section 2.9.1 with a path loss coefficient, $\gamma = 3$.

Path Selection: Path selection was carried out using Dijkstra's Shortest Path algorithm as described in section 4.1.3.

Multicast Rate: The multicast rate was set after all paths had been selected. The multicast rate is the rate of the slowest link in a multicast branch.

Tx Power: The transmit (Tx) power across all nodes was homogenous over all node densities and all topologies. This was regarded as a complete set of simulations. The Tx power was then incremented for the next set of simulations. Tx power was in the range 0 dBm – 18 dBm and incremented in 1 dBm steps.

Node Density: The fixed node density was increased from 12×10^{-6} Nodes/m^2 to 355×10^{-6} Nodes/m^2 (this amounts to 150 nodes distributed over the 422.5×10^3 m^2 area).

Topologies: For each node density 1000 random topologies were generated.

Link Cost Metrics: A full set of simulations over all Tx power settings were carried out for the following link cost metrics, minimum contention, minimum hop, minimum received power, minimum distance and modified ETT.

5.1.1 Mean Delay Characteristics

In the following diagrams we plot the relative delay (as outlined in section 4.1.4) for each fixed transmission power setting as the node density increases. In Figure 5-1, Figure 5-2 and Figure 5-3 we present the network delay characteristics when using the link cost metrics minimum contention, minimum hop and minimum received power respectively. We can see from the plots that the characteristic is similar for all three link cost metrics. The dark shaded area on the contour map indicates instances when the delay is relatively low. We can see that for low power and low node density the delay is at a minimum. If the node density and transmit power is too low, then few paths will be formed which will result in poor connectivity and hence lower delays (i.e. not all of the nodes in the network are connected back to the *Root*). We would expect the network delay to be relatively low when the node density is low due to shorter paths with fewer hops.

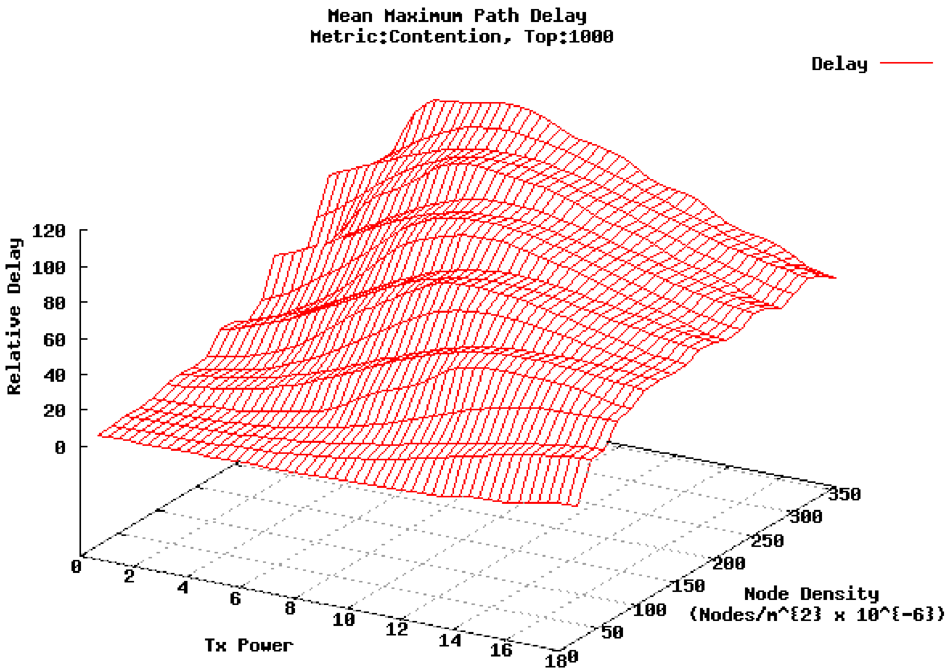


Figure 5-1 (a): Mean delay plot for minimum contention. Delay is at a maximum when the node density is high and Tx power is at 6 dBm.

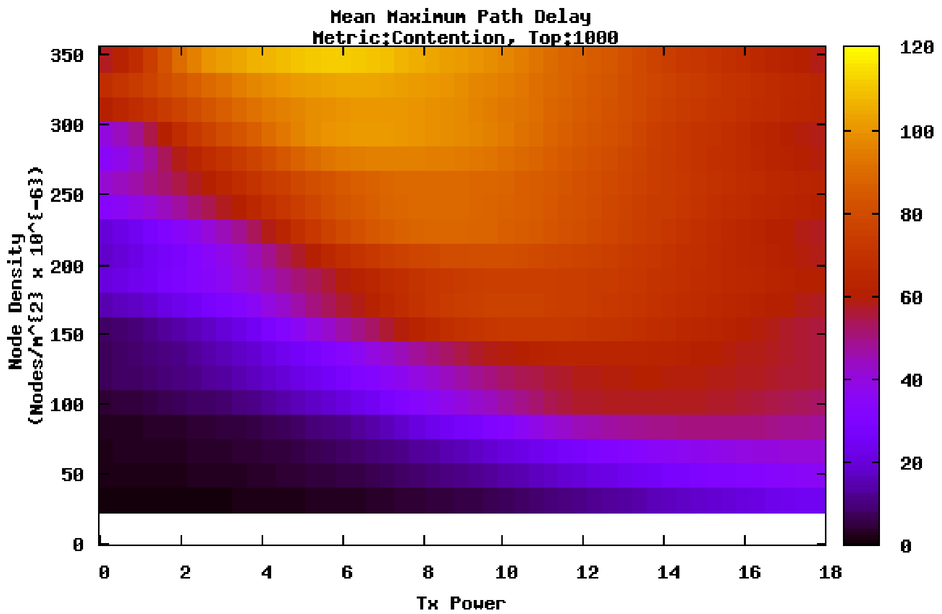


Figure 5-1 (b): Contour map of mean delay for minimum contention. Dark shading indicates low delays

When minimum hop (Figure 5-2) and minimum received power (Figure 5-3) are used we observe a small improvement at higher transmission powers. As the power increases the delay initially increases and reaches a maximum when the power is between 2 dBm and 10 dBm. As we increase the power further the delay begins to decrease. The relatively low delay characteristic is almost constant across all node density levels when the transmit power is above 16 dBm. The reduced delay at higher transmit powers is due to an increase in the selection of higher rate PHY links and as a consequence higher multicast rates are used which will reduce the delay.

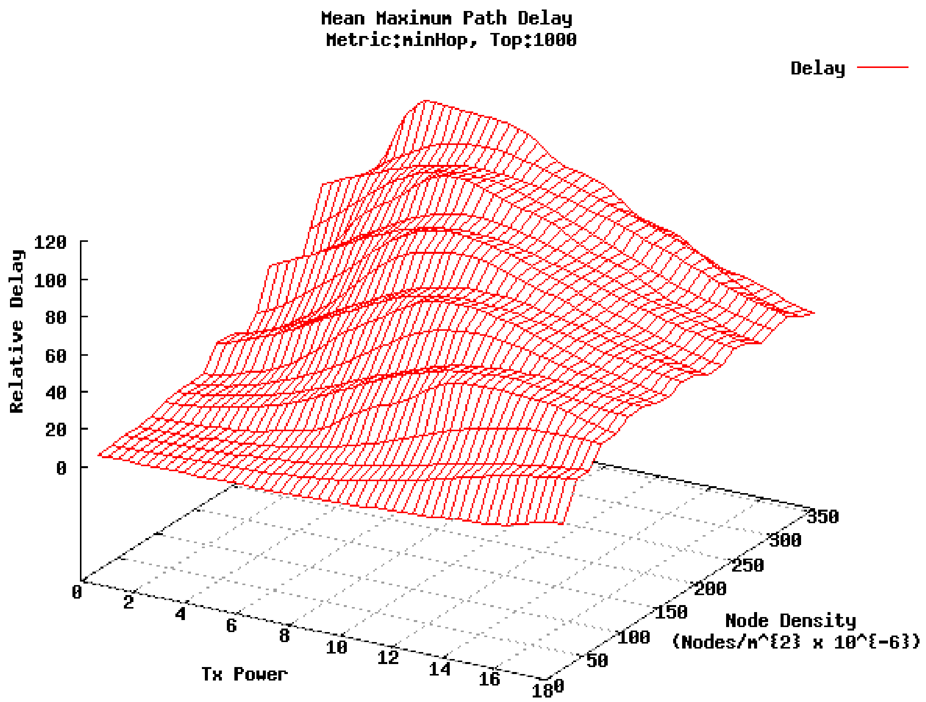


Figure 5-2 (a): Mean delay plot for minimum hop.

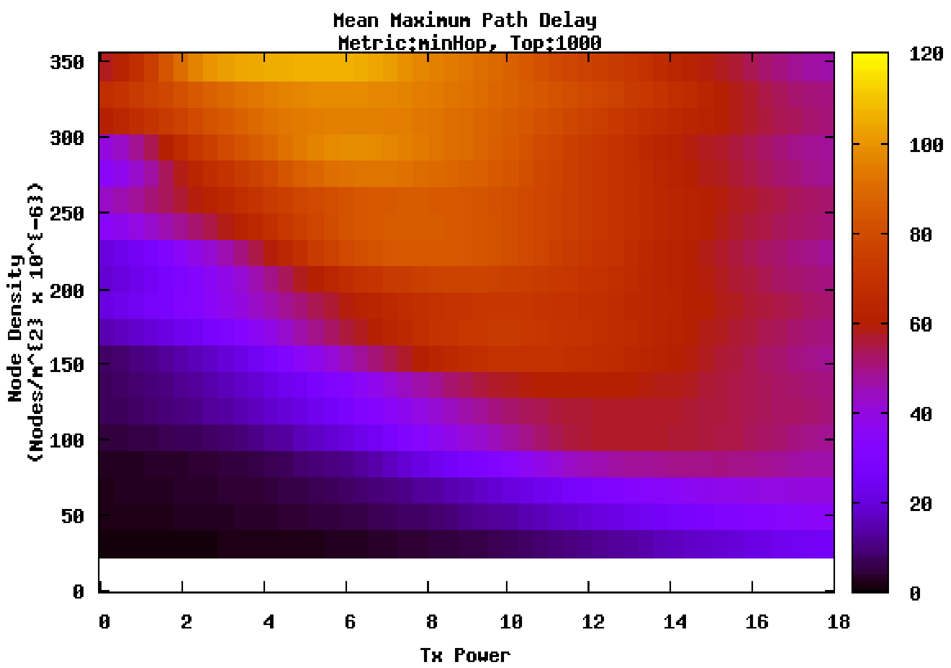


Figure 5-2 (b): Contour map of mean delay for minimum hop. Delay begins to decrease at higher Tx power.

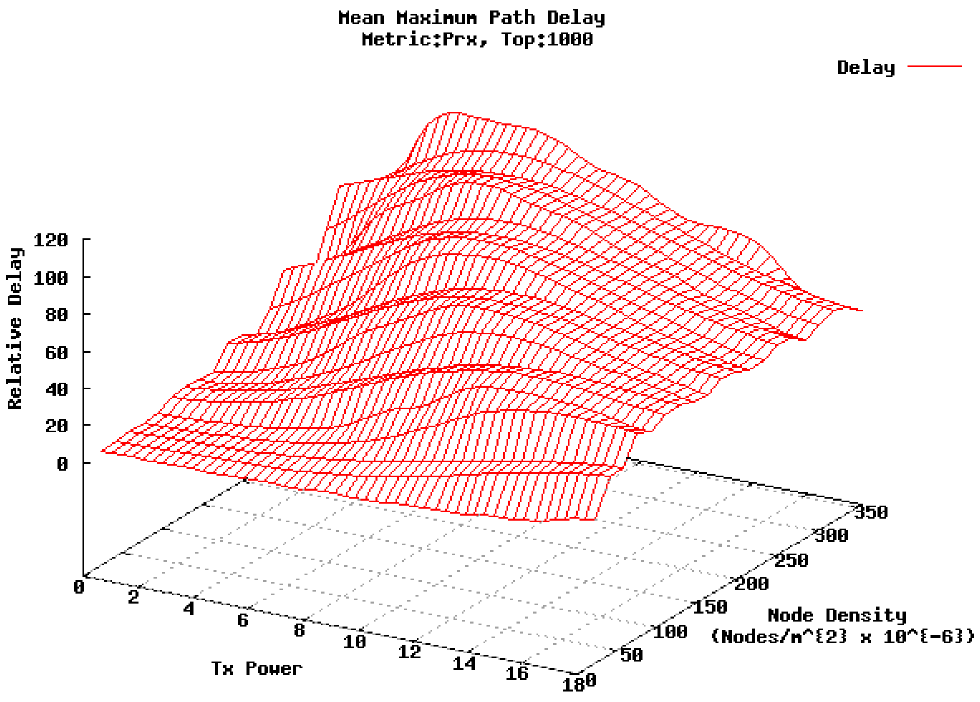


Figure 5-3 (a): Mean delay plot for minimum received power

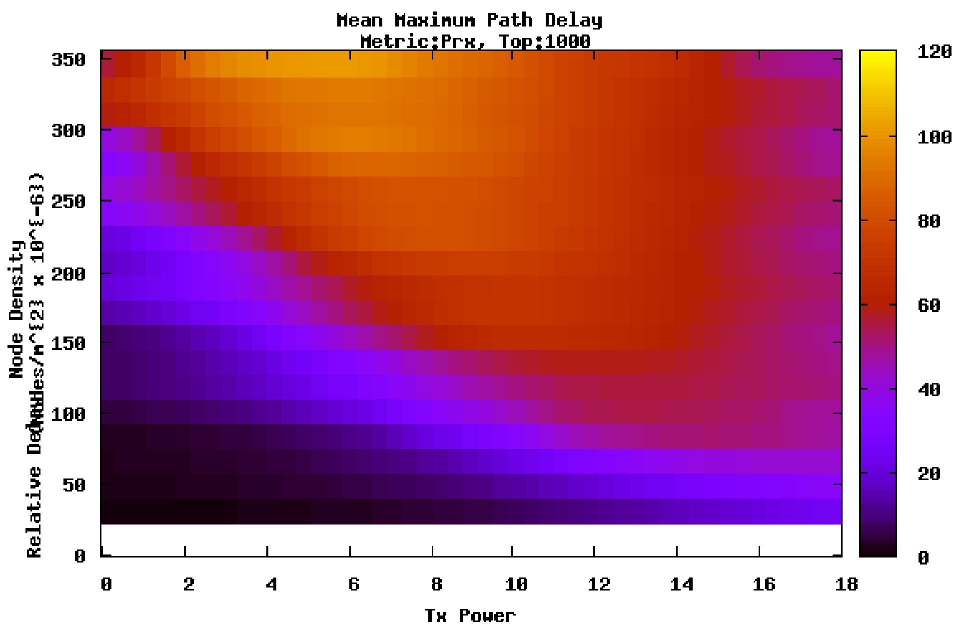


Figure 5-3 (b): Contour map of mean delay for minimum received power

The network performances, with respect to network delay, while using the previous three link cost metrics are similar due to how the spanning tree is formed. In the case of the first two, long links will be favoured (i.e. minimum contention will seek out routes around the edge of the working plane while minimum hop will cover the greatest distance with the least number of forwarding nodes). When minimum transmission power is used long links will again be selected. However, due to the reduced contention from neighbouring nodes (as a result of the low received power selection) we see an improvement when the transmit power is sufficiently high (Tx power > 16 dBm).

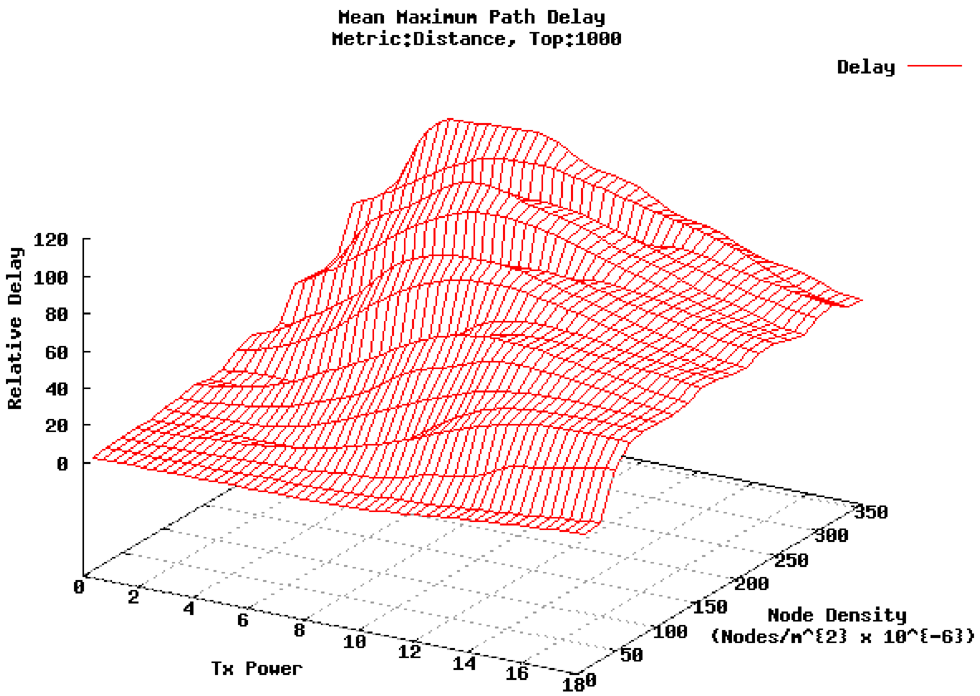


Figure 5-4 (a): Mean delay plot for minimum distance

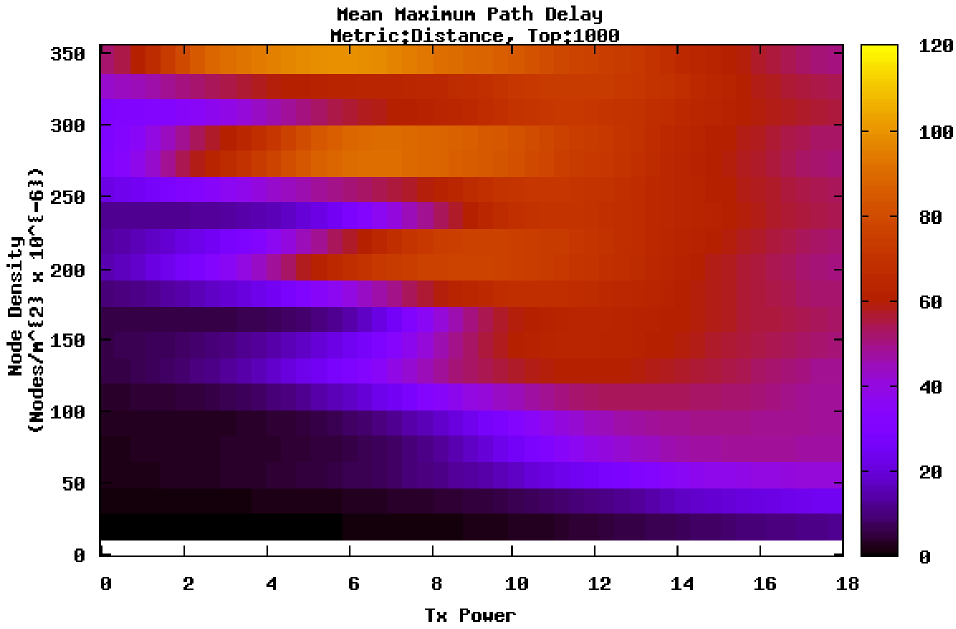


Figure 5-4 (b): Contour map of mean delay for minimum distance.

Figure 5-4 illustrates the network delay when a minimum distance link cost is used. We mentioned in section 4.1.5 that a minimum distance link cost is not practical in wireless networks. However, we use it in our simulation for evaluation purposes. Minimum distance will seek out a shortest physical path and as such will display a similar performance as minimum hop. There are slight variations in the delay characteristic when the transmit power is below 8 dBm, however, the general characteristic is similar. With regard to the variations in the delay characteristic further investigation is required. When the constellation diagrams are analysed it can be seen that there are occasions when a one hop path is equal to a two hop path. This occurs when a two hop path has a node at the midpoint along a path. This is significant only when the *minDist* link cost metric is used and can explain the variation in the delay characteristic. However, further investigation is required in order to determine why the variations occur at particular node densities.

In Figure 5-5 (a) we show a plot of the network delay when the ETT link cost metric is used. We observe from the diagram that the general shape of the plot is similar to the ones shown previously. However, one main distinction can be observed, that of noticeably lower delays when the transmit power is above 10 dBm. Furthermore, when we analyse the contour map Figure 5-5 (b), we observe that the maximum network delay is relatively low when compared to the maximum network delay for all other link cost metrics presented. Table 5-1 summarises the main characteristics of these plots.

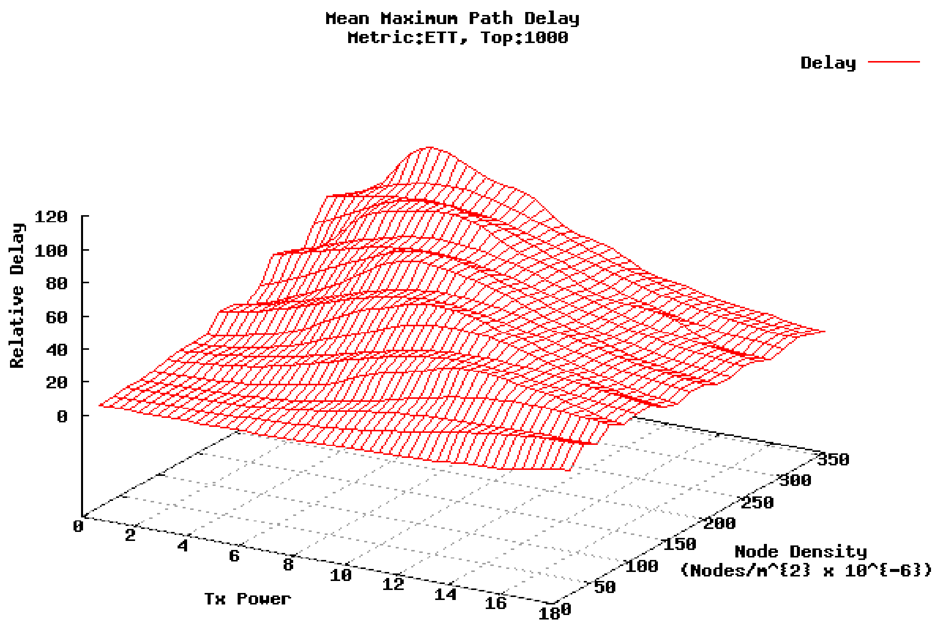


Figure 5-5 (a): Mean delay plot for ETT. Lower delays at higher Tx power.

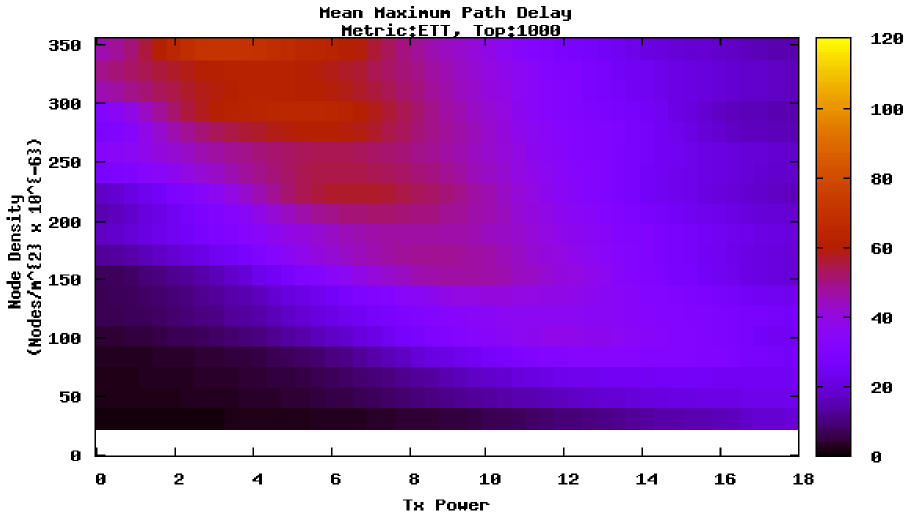


Figure 5-5 (b): Contour map of mean delay for ETT. Peak delay is lower than previous metrics

Minimum Contention

	Delay Max = 120	Comments
Tx Power	2 dBm – 10 dBm	A minimum delay is observed at low node density and Tx power. This can be misleading due to poor connectivity. Delay should be considered when the power and/or density are relatively high.
Density	> 200 x 10^{-6} Nodes/m^2	

Minimum Hop

	Delay Max = 120	Comments
Tx Power	2 dBm – 11 dBm	As previous case. Small improvement in delay as the power increases above 16 dBm.
Density	> 200 x 10^{-6} Nodes/m^2	

Minimum Received Power

	Delay Max = 110	Comments
Tx Power	2 dBm – 11 dBm	As previous case.
Density	> 200 x 10^{-6} Nodes/m^2	

Minimum Distance

	Delay Max = 100	Comments
Tx Power	3 dBm – 10 dBm	As previous case. Minimum distance is not practical for use as a link cost metric. It is however useful to validate simulation performance.
Density	> 250 x 10^{-6} Nodes/m^2	

ETT

	Delay Max = 60	Comments
Tx Power	2 dBm – 7 dBm	Significant improvement with lower delays across all densities and Tx power settings.
Density	> 275 x 10^{-6} Nodes/m^2	

Table 5-1: Comparison of main delay characteristics for each link cost metric.

5.1.2 Mean Throughput Characteristics

Our next step is to observe the effects of various link cost metrics on the mean network throughput performance (as outlined in section 4.1.4). As expected we find that the minimum contention (Figure 5-6), minimum hop (Figure 5-7) and minimum received power (Figure 5-8) perform similarly. For all three link cost metrics, we observe that for maximum throughput performance the transmit power and node density are inversely proportional to each other. Essentially this means that as the density of the network increases, the contention/interference increases and therefore the transmit power should be decreased in order to maximise the throughput.

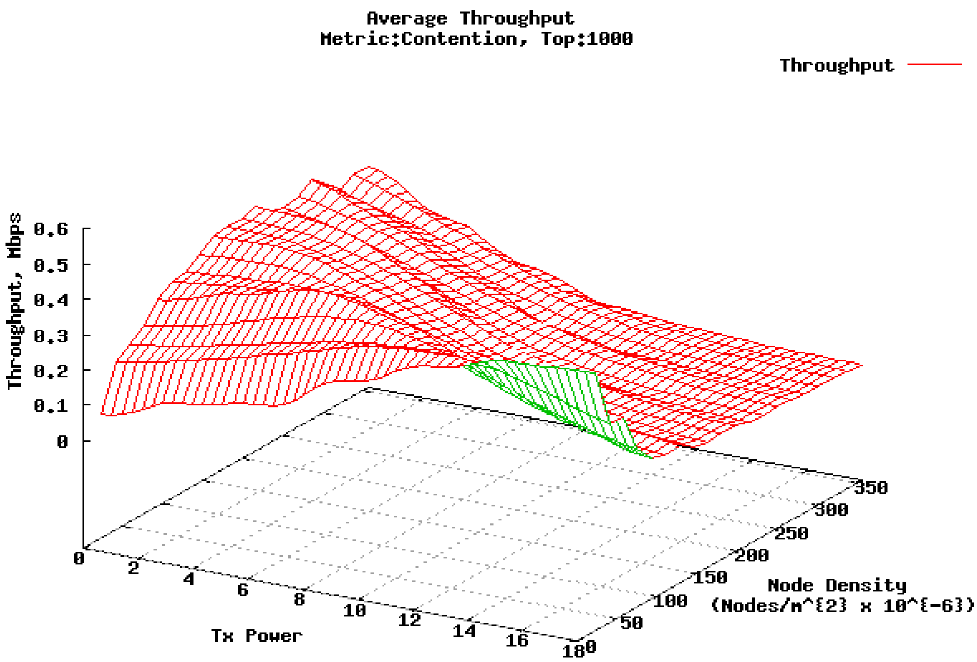


Figure 5-6 (a): Mean throughput plot for minimum contention

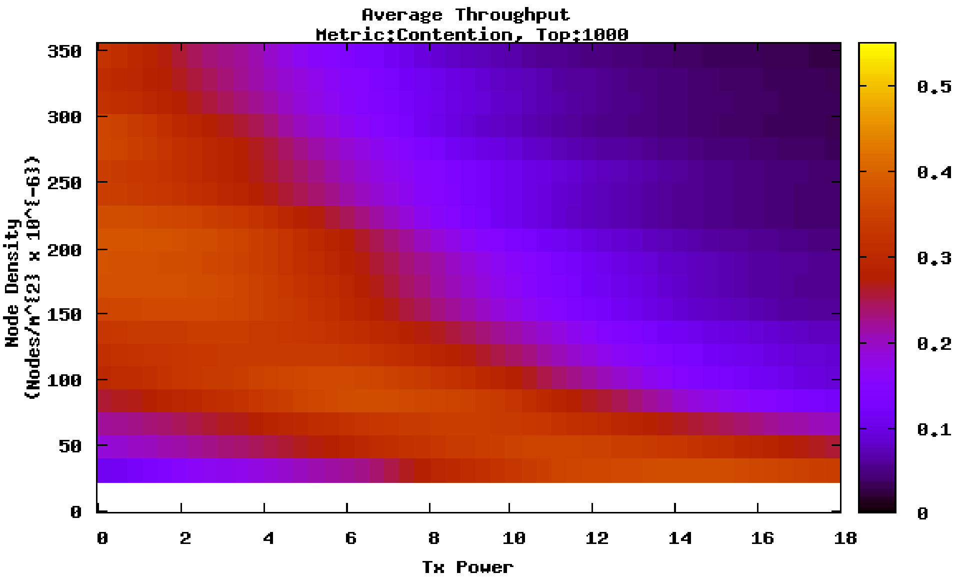


Figure 5-6 (b): Mean throughput contour map for minimum contention

The bottom left corner of each of the contour maps (light blue shading) indicates a network with poor connectivity (i.e. the density is too low and/or the transmit power is too low to establish paths).

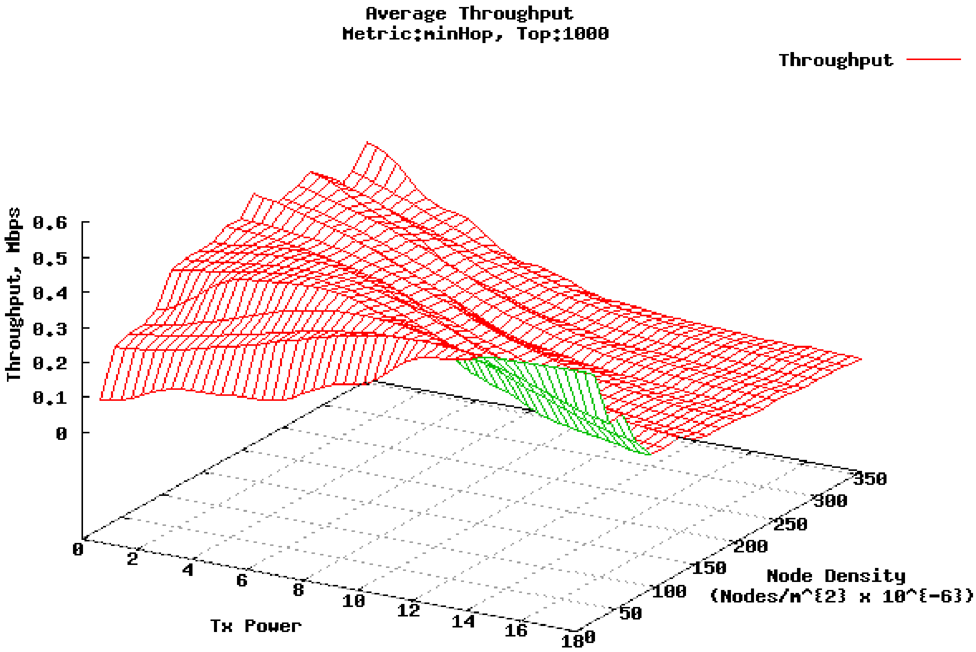


Figure 5-7 (a): Mean throughput plot for minimum hop.

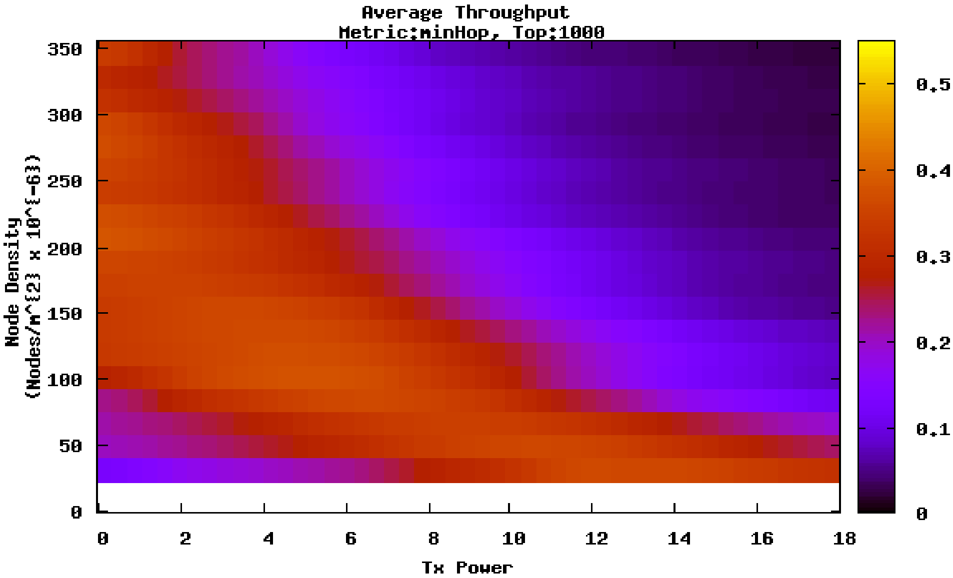


Figure 5-7 (b): Mean throughput contour map for minimum hop.

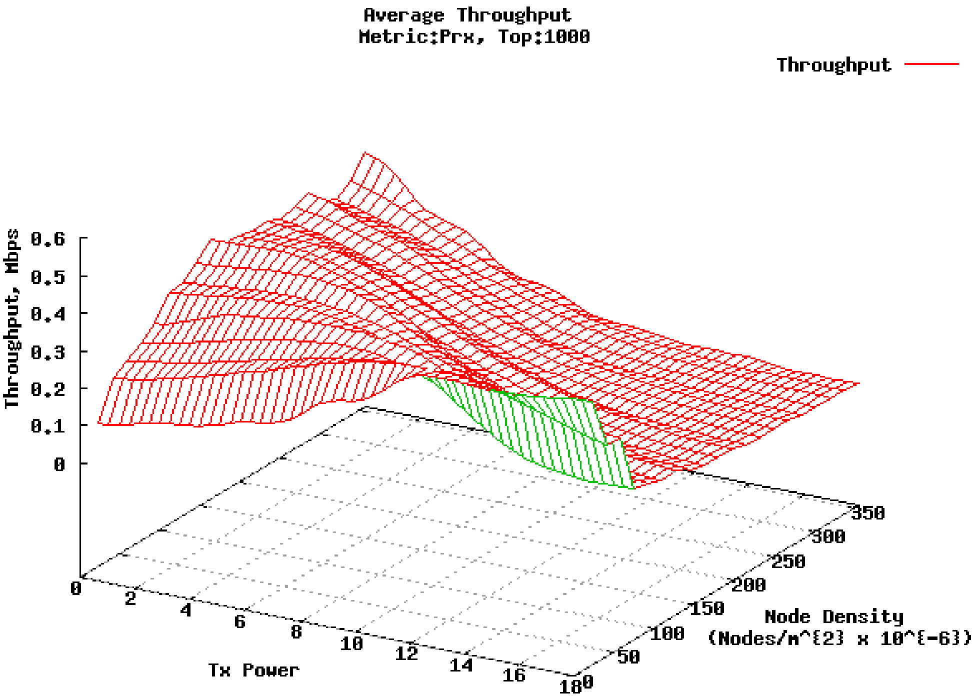


Figure 5-8 (a): Mean throughput plot for maximum received power.

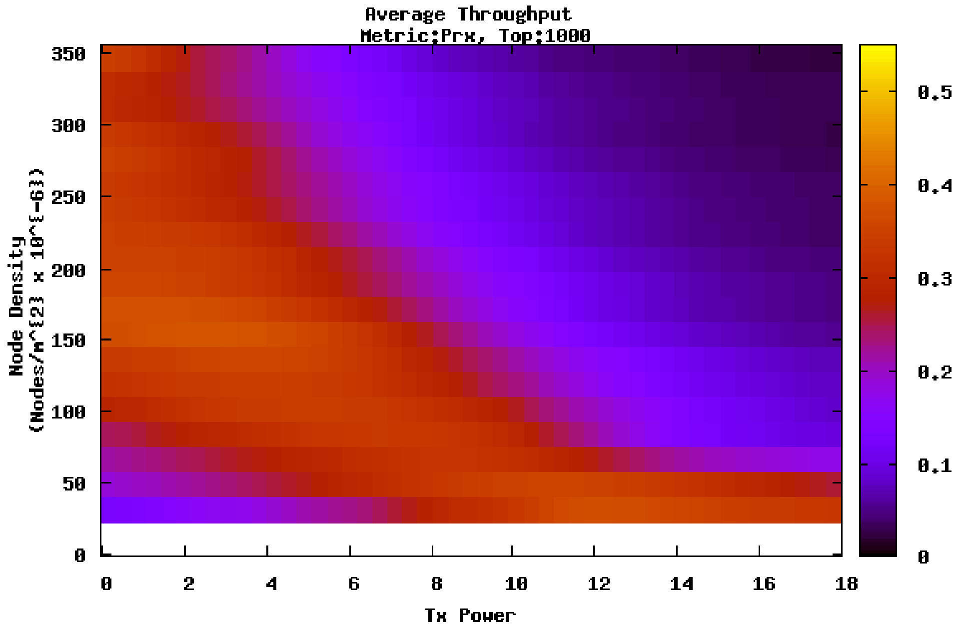


Figure 5-8 (b): Mean throughput contour map for maximum received power.

Similarly, the performance of the network when minimum distance is used (Figure 5-9) is again comparable with that of minimum hop.

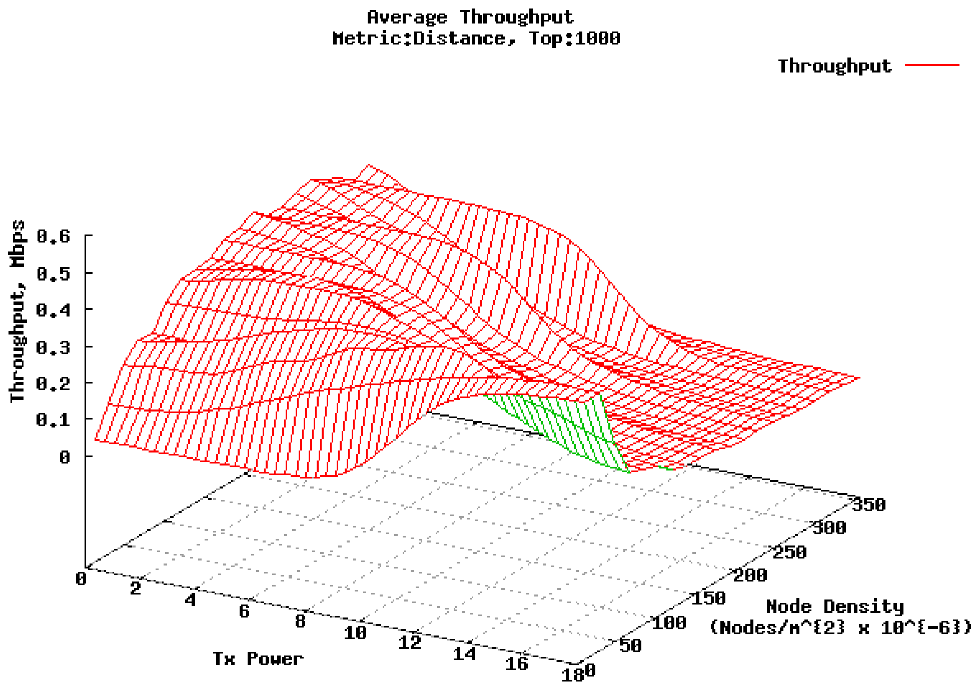


Figure 5-9 (a): Mean throughput plot for minimum distance.

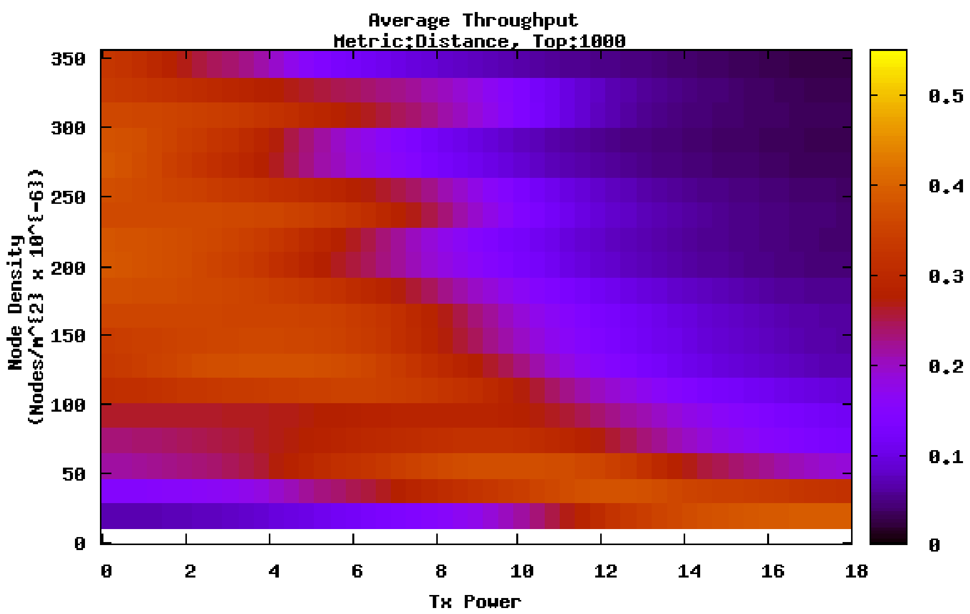


Figure 5-9 (b): Mean throughput contour map for minimum distance.

Variations can be seen in the throughput performance (and the delay performance). A more thorough investigation would be necessary to give an explanation as to why this occurs. Again, this metric is used for comparative performance analysis as it will exhibit similar behaviour to minimum hop.

In Figure 5-10 we examine the network throughput performance when the ETT link cost metric is used. Once again it can be observed that the ETT metric outperforms the previous metrics considered. The general characteristic is similar to the previous link cost metrics. However, the peak throughput is relatively higher. This is also coupled with the fact that the increase in performance is maintained over a greater node density and greater transmit power range. This characteristic would indicate that the link cost metric ETT would be more tolerant to variations in transmit power while also achieving higher throughputs.

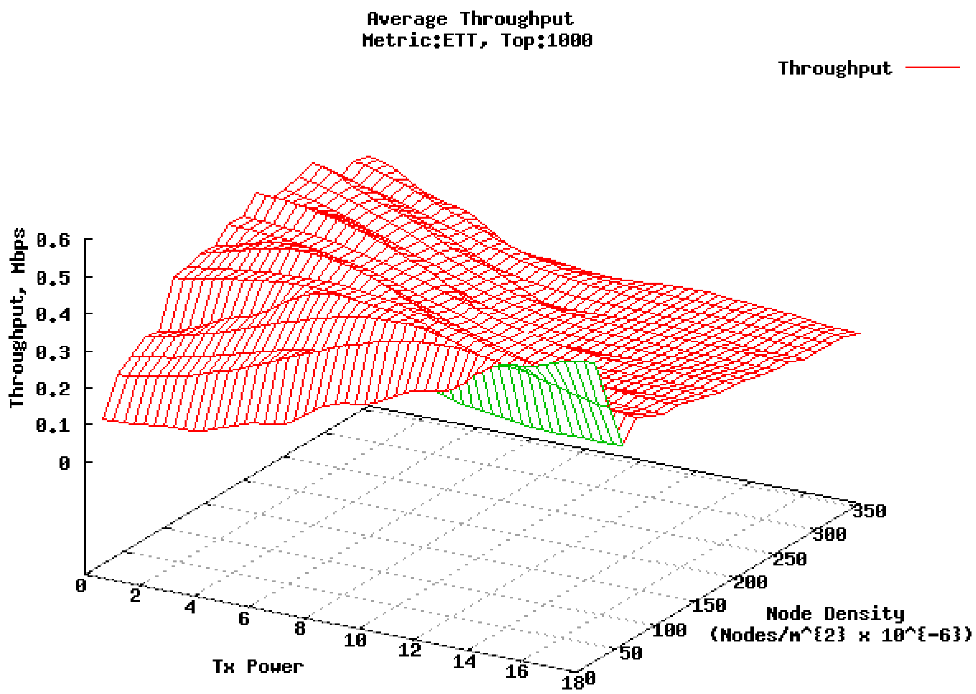


Figure 5-10 (a): Mean throughput plot for ETT.

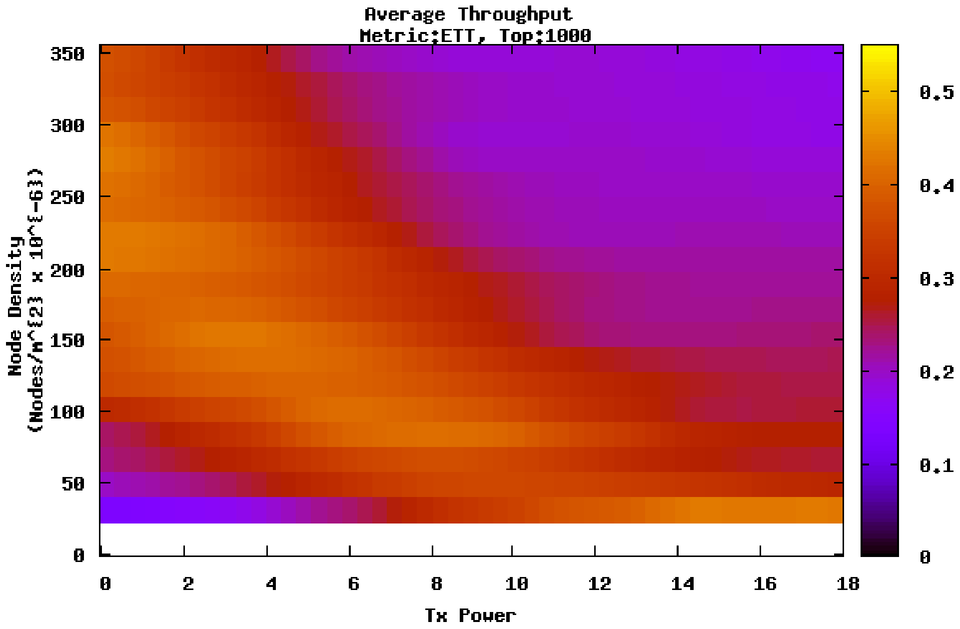


Figure 5-10 (b): Mean throughput contour map for ETT.

If we observe the network constellation diagrams using two of the link cost metrics we can observe how the paths are formed. In Figure 5-11 and Figure 5-12 we illustrate how paths are formed using minimum hop and ETT respectively. We can clearly see from Figure 5-11 that the use of minimum hop will result in links being chosen which will have a low transmission rate (indicated in green). In Figure 5-12 the majority of link rates are higher than 1 Mbps (2 Mbps yellow, 5.5 Mbps orange and 11 Mbps red).

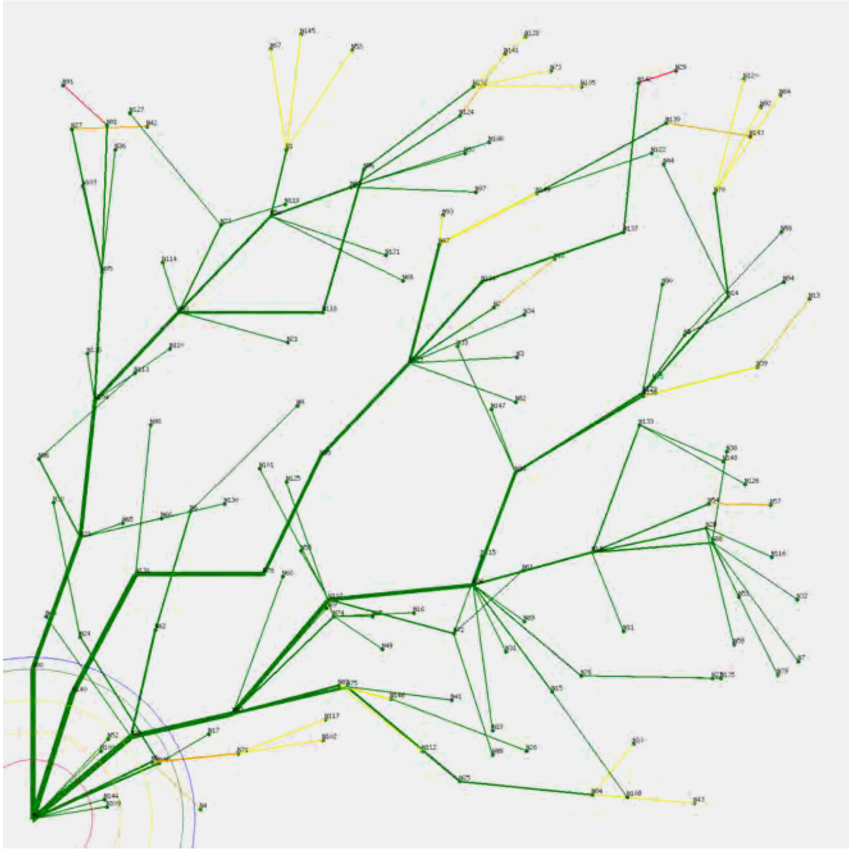


Figure 5-11: Network diagram using minimum hop. Majority of links are 1 Mbps (green)

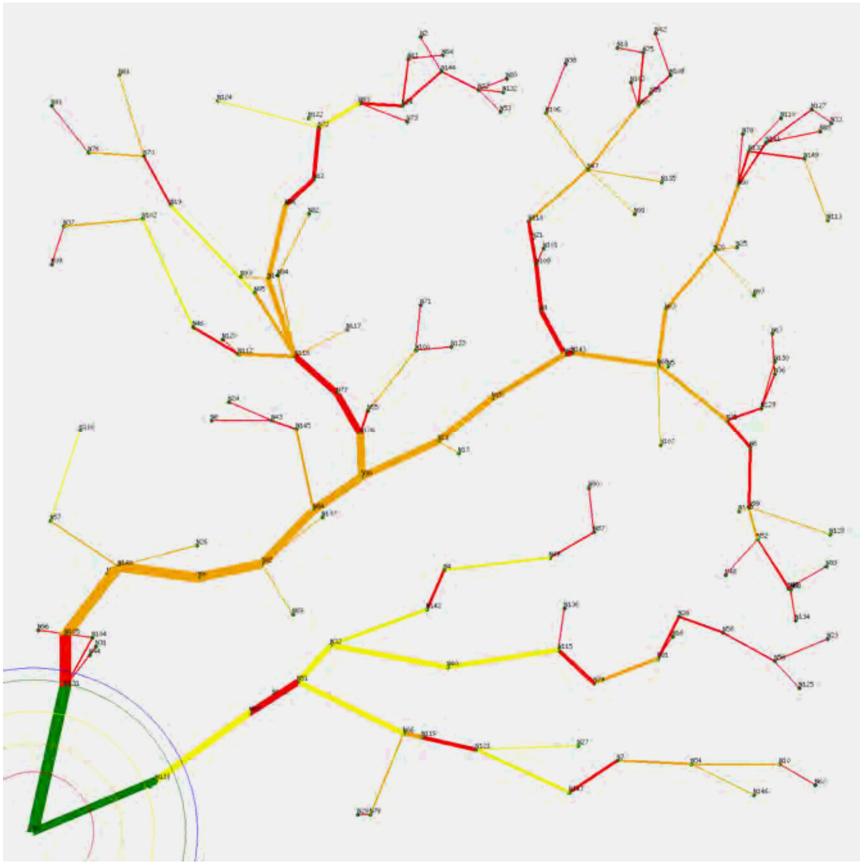


Figure 5-12: Network diagram using ETT. Majority of links have a rate greater than 1 Mbps

Table 5-2 summarises the main characteristics of these plots.

Minimum Contention

	Throughput Max = 0.3 Mbps	Comments
Tx Power	High power at low density Low power at high density	A minimum throughput performance is observed when the node density and Tx power are relatively high. Best performance achieved when values are inversely proportional
Density	Low density at high power High density at low power	

Minimum Hop

	Throughput Max = 0.3 Mbps	Comments
Tx Power	High power at low density Low power at high density	As in the previous case.
Density	Low density at high power High density at low power	

Minimum Received Power

	Throughput Max = 0.3 Mbps	Comments
Tx Power	High power at low density Low power at high density	As in the previous case
Density	Low density at high power High density at low power	

Minimum Distance

	Throughput Max = 0.3 Mbps	Comments
Tx Power	High power at low density Low power at high density	As in the previous case. Minimum distance is not practical for use as a link cost metric. It is however useful to validate simulation performance.
Density	Low density at high power High density at low power	

ETT

	Throughput Max = 0.55 Mbps	Comments
Tx Power	High power at low density Low power at high density	Significant improvement in max throughput performance. Higher throughput observed for greater range of node density and Tx Power. Suggests higher tolerance to changes in the network
Density	Low density at high power High density at low power	

Table 5-2: Comparison of main throughput characteristics for each link cost metric.

5.1.3 Node Coverage

In this section we will briefly examine the network coverage as the node density increases. As mentioned in section 4.5 we use Monte Carlo sampling to obtain the percentage of node coverage for each PHY rate. Samples determine the percentage coverage of nodes (at each PHY rate) connected to the multicast tree. Ideally we would hope to achieve 100% coverage at the maximum available transmission rate. However, this is highly impractical from a deployment cost point. Figure 5-13 illustrates a typical node coverage plot for a fixed transmission power.

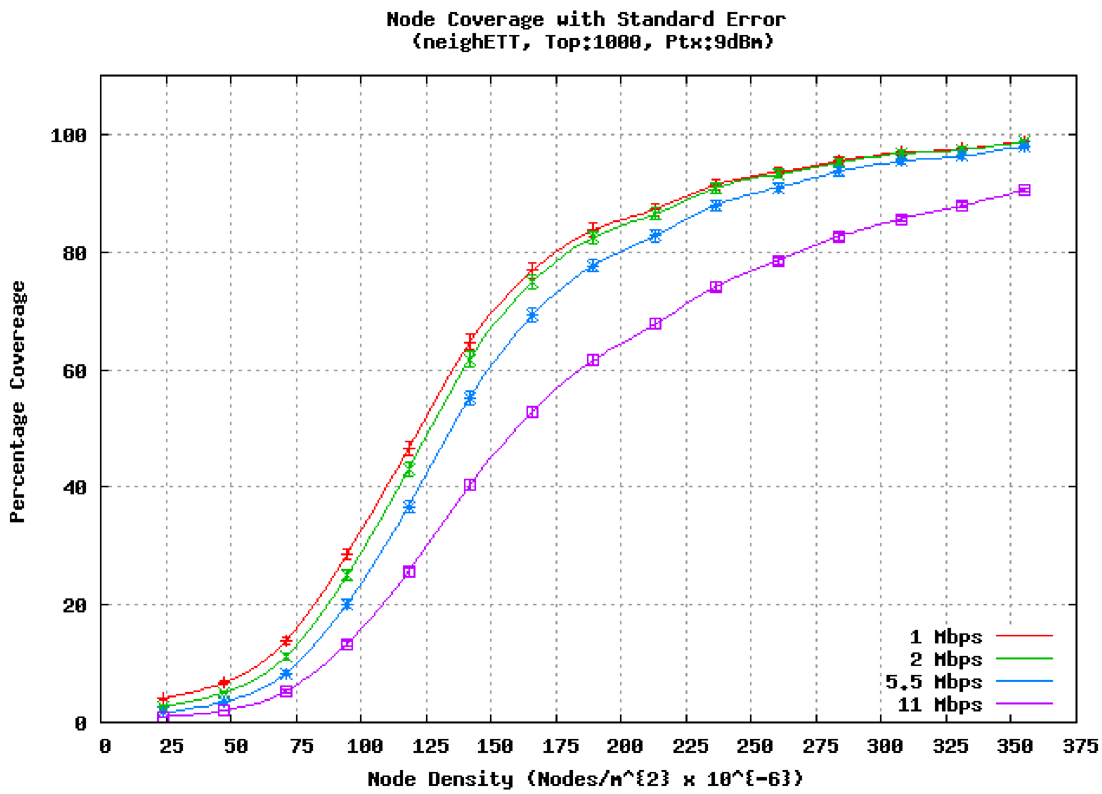


Figure 5-13: Percentage node coverage when using ETT at a fixed transmission power of 9 dBm.

In the figure we can examine the percentage coverage as the node density increases. Figure 5-13 plots the node coverage when using ETT and with a fixed transmit power of 9 dBm. As the node density increases so too does the percentage node coverage for each transmission rate. We can observe from the figure that at high node density deployments the coverage approaches 100%. It should be noted that the coverage does not guarantee that a particular rate will be used. The coverage is an indication of the maximum rate available; the multicast effect will ultimately determine the rate to be used.

For comparison we also provide two extreme cases for the same link cost metric (i.e. with the transmit power fixed at 3 dBm and 18 dBm). We can see from Figure 5-14 that for high node density deployment the coverage is comparatively low (less than 60% for 1 Mbps). In Figure 5-15 we observe a different effect. As a result of the high power setting, the transmission range of all nodes is extended significantly. As a result, node coverage (for all transmission rates) achieves maximum coverage at relatively low node densities. Although this high coverage is desirable, the high transmit power creates adverse effects due to increased interference and hence increased contention. This is evident in the delay and throughput plots in the previous sections.

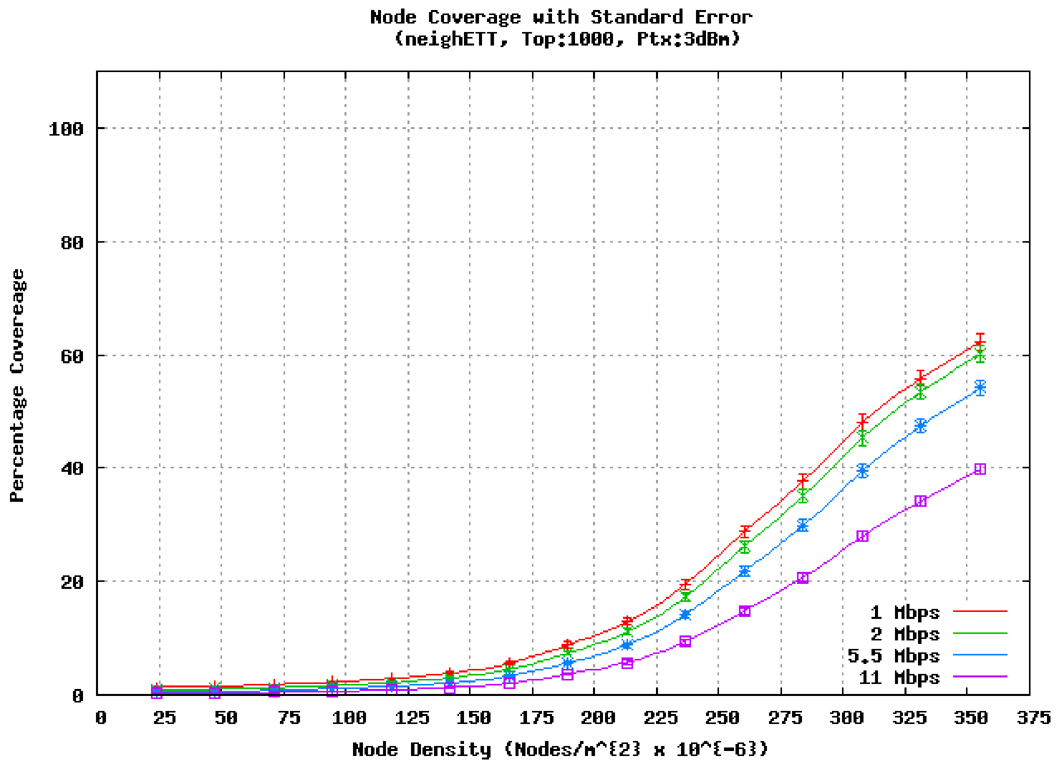


Figure 5-14: Node coverage when using ETT at 3 dBm.

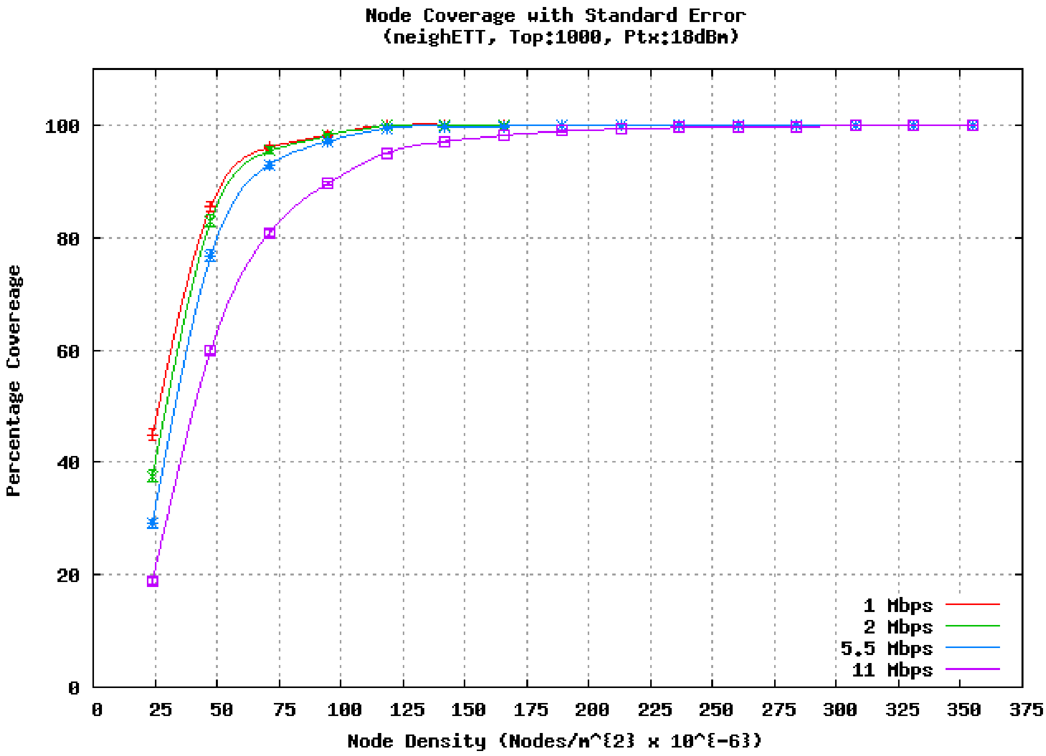


Figure 5-15: Node coverage when using ETT at 18 dBm.

5.1.4 Summary

In this section we have presented a comparison of the delay and throughput performance of five link cost metrics (minimum contention, minimum hop, minimum received power, minimum distance and ETT) for the Basic Model. We have shown that the performance of the first four is closely matched due to the construction of the spanning tree (i.e. long links are used with low transmission rates). The exception to this is when ETT is used. When ETT is used with Dijkstra's shortest path algorithm, shorter links with lower delay and higher transmission rates are selected. This results in *Child* nodes of the same *Parent* having a higher multicast rate.

The coverage plots show us the effects of increasing the node density and transmit power. Although increasing the node density and transmit power will lead to higher coverage at higher rates, there is no guarantee of higher throughput as can be seen from the throughput plots in section 5.1.2.

5.2 Midpoint Node Optimisation

In section 4.2 we described a method of eliminating slow 1 Mbps links in order to achieve higher link rates and hence higher multicast rates. In this section we will present the performance results from simulations using two link cost metrics (minimum hop and ETT) with Dijkstra's shortest path algorithm. The simulation set-up uses the Basic Model as set out in section 5.1. However, this time we will run simulations using a single fixed transmit power of 9 dBm. We choose the mid range transmit power based on the results from the un-optimised results. In the un-optimised results we observed that if the transmit power is too low then paths will not be formed. On the other hand if the transmit power is too high then nodes will interfere with transmissions of neighbouring nodes resulting in an increase in contention and hence lower throughput. From section 5.1 we observed that the minimum hop link cost metric displays similar performance characteristics as our other link cost metrics, with the exception of ETT. Therefore, we will analyse these two metrics in the following sections. We will first present the results for the case where we remove all 1 Mbps links by introducing additional relay nodes. We will further extend this by continuing to add relay nodes in order to guarantee all 11 Mbps links and hence an 11 Mbps multicast rate. We will end the section with the main findings from our observation and analysis.

5.2.1 Adding Relay Nodes on 1 Mbps Links using ETT

In this section we will analyse the network performance as we run our algorithm to place relay nodes along 1 Mbps links. In Figure 5-16 and Figure 5-17 we illustrate the effect on performance from adding new relay nodes for a series of fixed node densities. In both figures each of the coloured lines represents the performance of the network for a particular fixed node density. As the number of relay nodes increases, the throughput and delay will change accordingly.

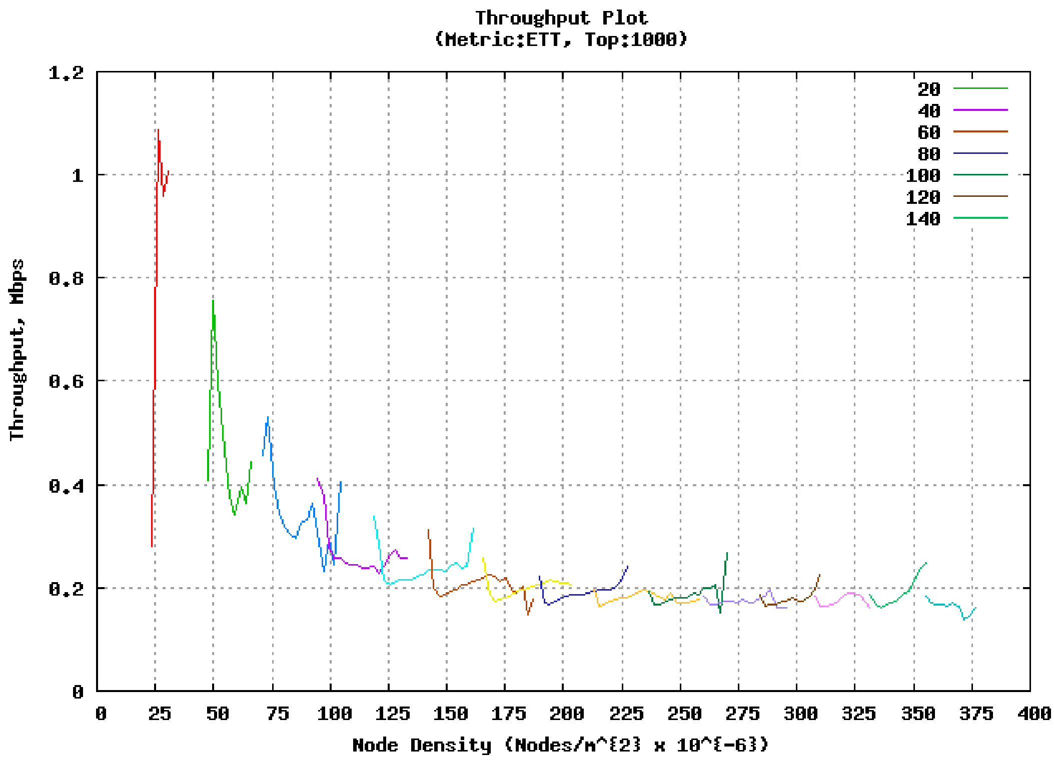


Figure 5-16: ETT throughput plot for increased fixed node density with relay nodes.

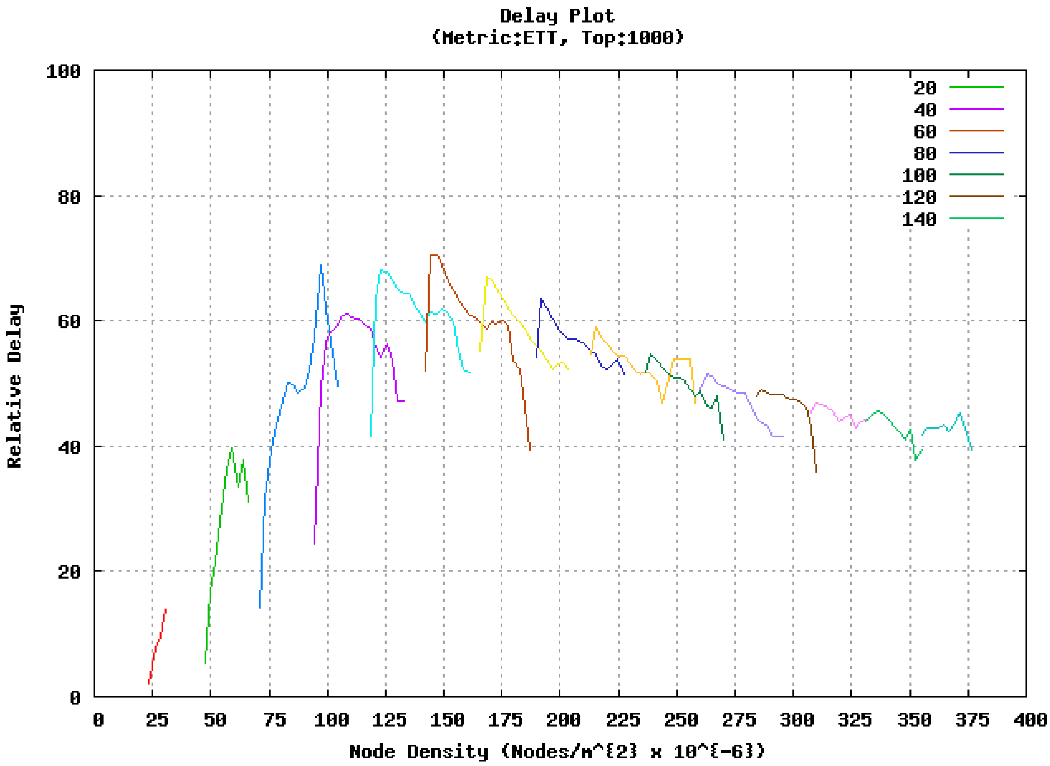


Figure 5-17: ETT delay plot for increased fixed node density with relay nodes.

We observe that both the throughput and delay follow a similar pattern as that shown in section 5.1.2 (i.e. the throughput decreases and the delay begins to decrease as the fixed node density increases). However, for this set of simulations we further increase the node density by adding new relay nodes. Therefore, the first point on each line represents the initial fixed node density. The chacteristic of each line changes as the number of relay nodes is added. The characteristics are discused in further detail later in this section and illustrated in Figure 5-18 and Figure 5-19. We can see from Figure 5-16 that for each fixed node density the throughput initially drops as relay nodes are added. Likewise, the delay increases as the node density increases. A reduced delay can only be seen once the fixed node density has reached a critical point of 150 x 10^{-6} Nodes/m^2.

In Figure 5-18 we present the mean network throughput (over 1000 topologies) when using 100, 120 and 140 fixed nodes to help illustrate this effect. As the number of additional relay nodes increases we begin to observe a gradual increase again in the throughput. The initial drop in performance is due to an increase in contention as we add relay nodes. If there is more than one *Child* node (on the same multicast branch) with a link rate of 1 Mbps then adding one relay node will result in a decrease in performance (due to increased contention). To observe the benefits from adding relay nodes we must continue to add nodes until all 1 Mbps links have been removed from a branch point (i.e. the rate to all *Child* nodes has increased by increasing the multicast rate).

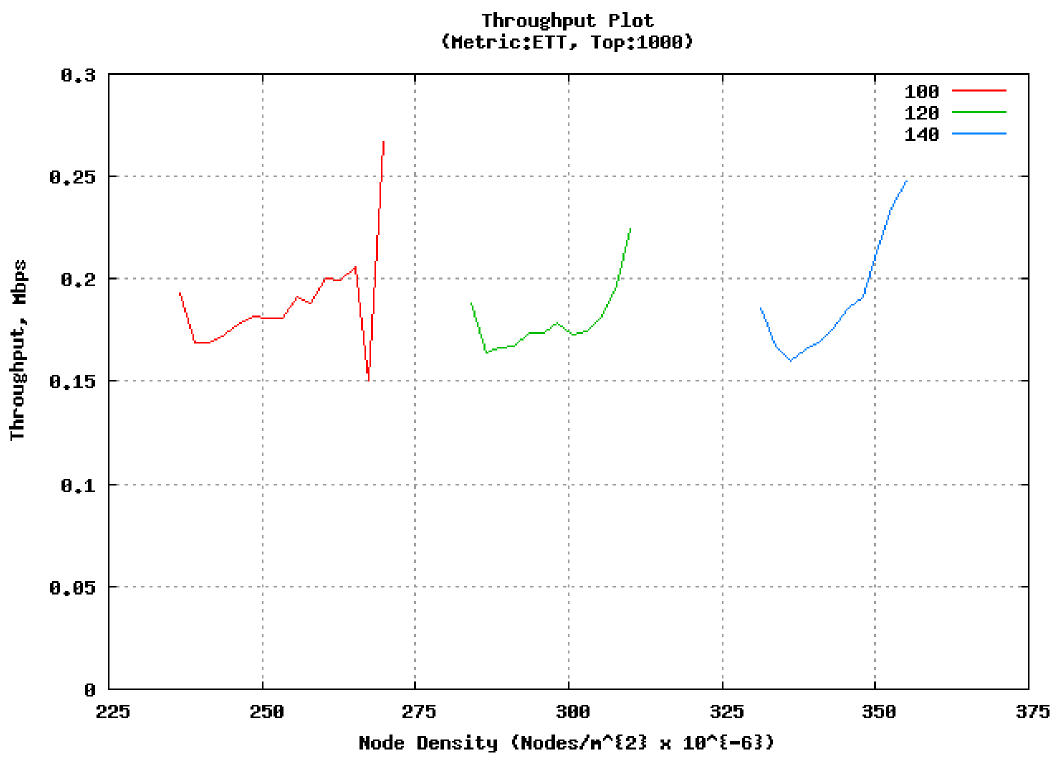


Figure 5-18: ETT throughput plot with relay nodes for 100, 120 & 140 fixed nodes.

Therefore, to observe the gain we should consider the difference between the starting throughput, (i.e. the throughput for a particular fixed node setting without relay nodes) and the final throughput (i.e. when all 1 Mbps links have been eliminated for a particular fixed node setting). Figure 5-19 represents the performance for a single topology using 100 fixed nodes. We can clearly see the effects of adding nodes. Recall from section 4.2 that relay nodes are first placed on point-to-multipoint 1 Mbps links before being placed on point-to-point 1 Mbps links. The stepped increase in throughput is due to a higher multicast rate being used for point-to-multipoint communication. The effects of adding nodes decreases in the latter stages as relay nodes are placed on point-to-point links. The same stepped characteristic can be seen in the delay plot also shown in Figure 5-19.

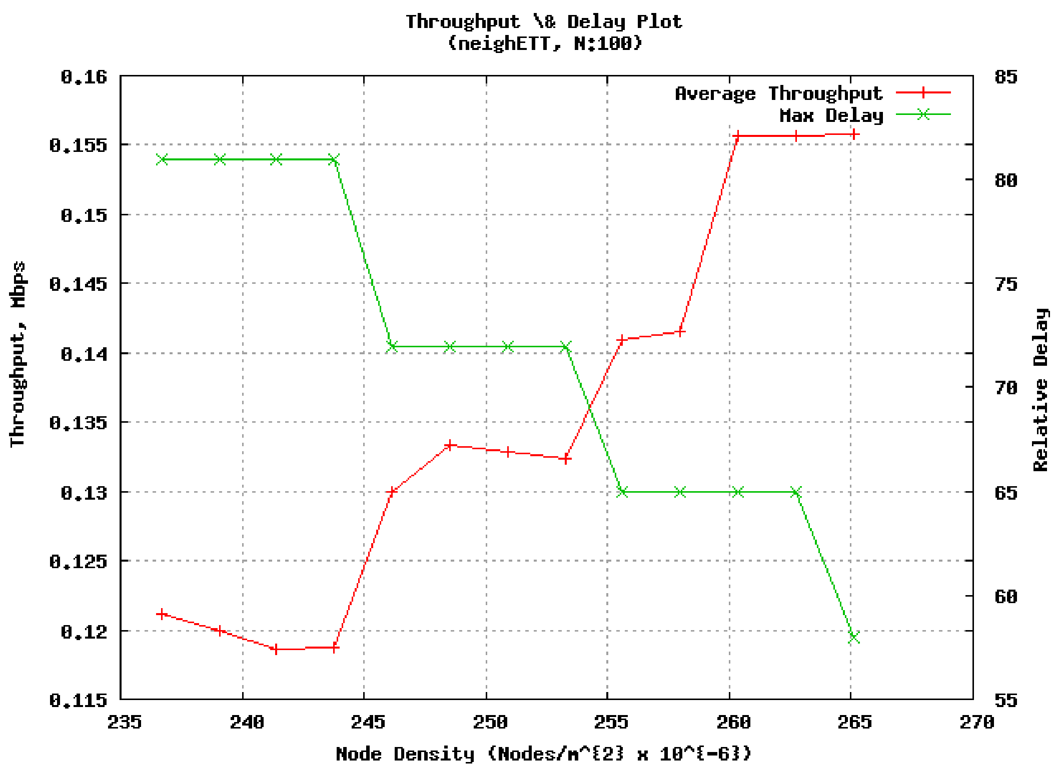


Figure 5-19: ETT throughput and delay plot for a single topology with 100 fixed nodes. Node density is the total fixed plus additional relay nodes used.

In the previous plots (Figure 5-16, Figure 5-17 and Figure 5-18) we characterise the performance using the mean values. The mean values provide us with useful information regarding general characteristics of the network behaviour. However, detail can be obscured or hidden when using such data. Figure 5-19 is sufficient for detailing the performance for a single topology. If we now plot the PDF distribution for all 1000 random topologies in a 100 fixed node simulation we can obtain a more insightful view of the network performance. In Figure 5-20 we display the PDF of mean network throughput for ETT and *MinHop* when no relay nodes are used (i.e. m = 0) and when the maximum number of relay nodes is used (i.e. m = 14 for ETT and m = 63 for *MinHop*).

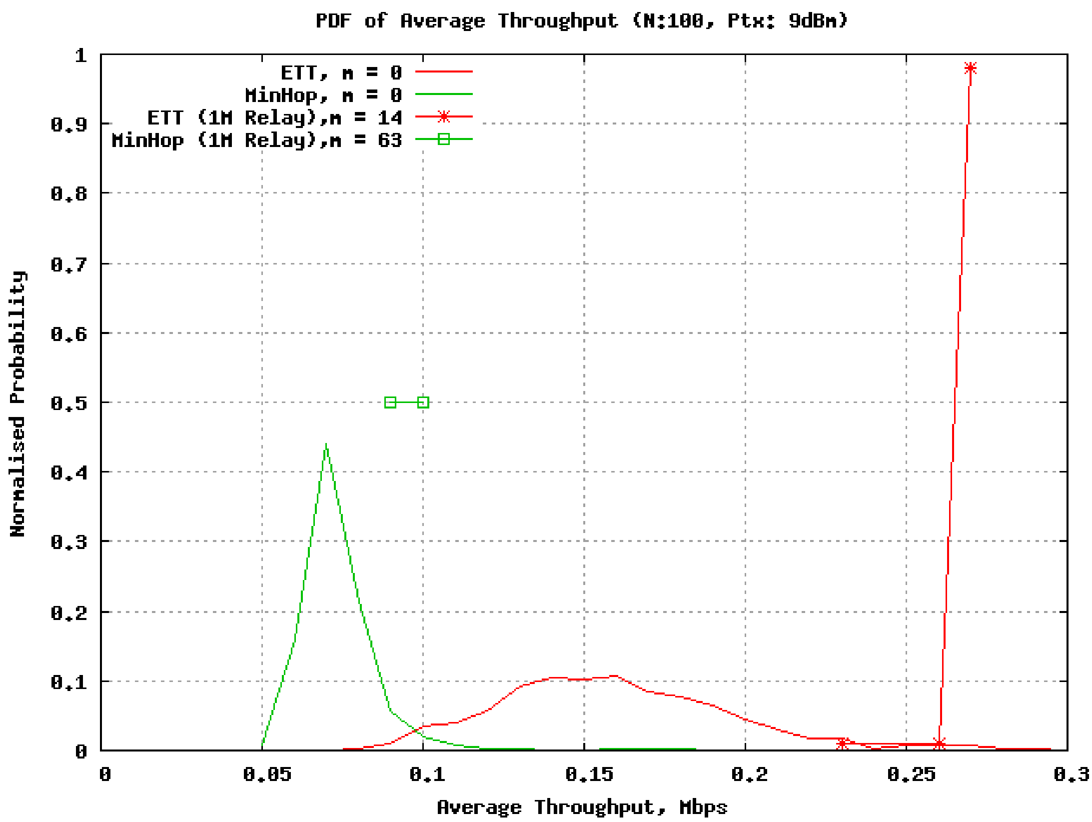


Figure 5-20: PDF of throughput for ETT and *MinHop*. Fixed nodes, N = 100, Relay nodes, m = 0

Using the peak probability values (rather than the mean throughput) we determine the performance gain as relay nodes are added. By graphing the probability density function (PDF) it is possible to quickly see the distribution of values including worst and best case topologies. For example using Figure 5-20, looking at the PDF for ETT we can see that the peak probability is 0.106 with a throughput of 0.16 Mbps. Using ETT while adding relay nodes to eliminate 1 Mbps links we find that max relay nodes, m_{peak} = 14, peak probability = 0.98 for a throughput of 0.27 Mbps (see Appendix C1 for probability tables and Table 5-3 for comparison of gains using ETT and *MinHop*). In the case of 100 fixed nodes the throughput gain is approximately 69% when the number of relay nodes peaks at 14. Due to the path selection when using *MinHop* the number of 1 Mbps links is comparatively high (see Figure 5-12) hence we see a narrow distribution of mean network throughput values when m = 0. Conversely, with ETT the number of 1 Mbps links is comparatively low when m = 0 (see Figure 5-12) hence a broader distribution of values due to multiple line rates being used. We will see in section 5.2.3 how we can continue to add relay nodes with the aim of improving the multicast rate further.

5.2.2 Adding Relay Nodes on 1 Mbps Links using MinHop

In this section we present a performance evaluation when using the minimum hop (*MinHop*) link cost metric along with our algorithm to place relay nodes. As in the previous section the results for throughput and delay follow the same general characteristics as the un-optimised results. From Figure 5-21 we can see that the throughput performance of *MinHop* with relay nodes is still lower than ETT. After the node density passes 200 x 10^{-6} Nodes/m^2 the relative throughput stays below 0.2 Mbps. The delay performance in Figure 5-22 displays a significant difference with a greater drop in relative delay as relay nodes are added. However, the overall delay performance after the maximum number of relay nodes have been added remains lower than that of ETT.

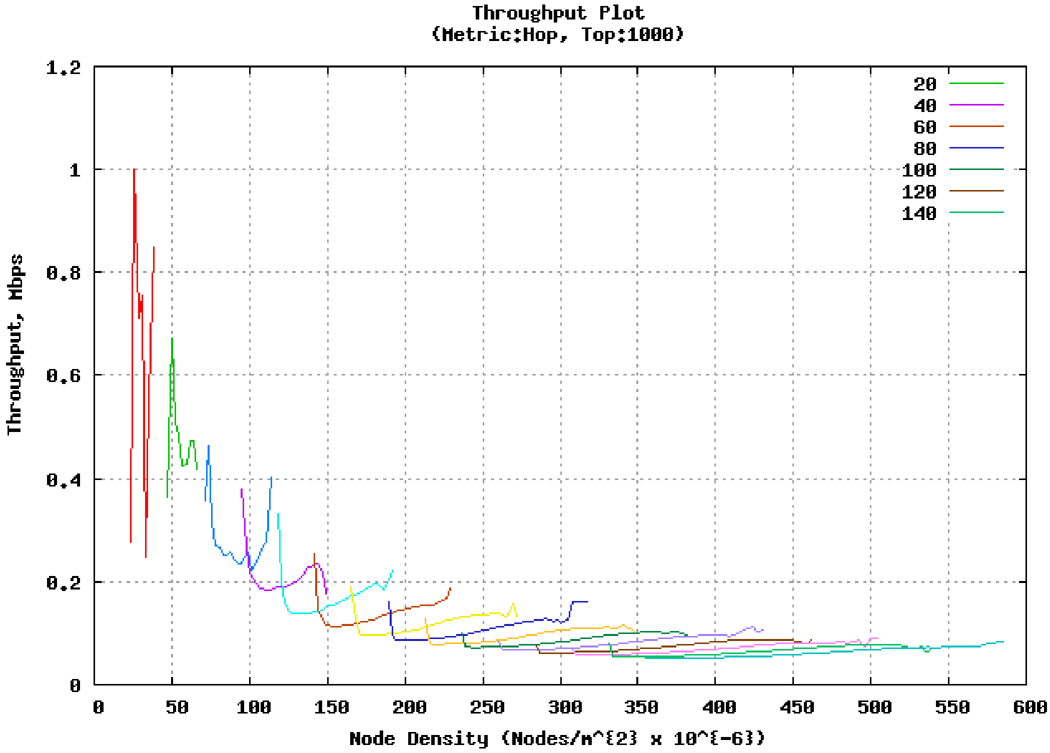


Figure 5-21: *MinHop* throughput plot for increased fixed node density with relay nodes.

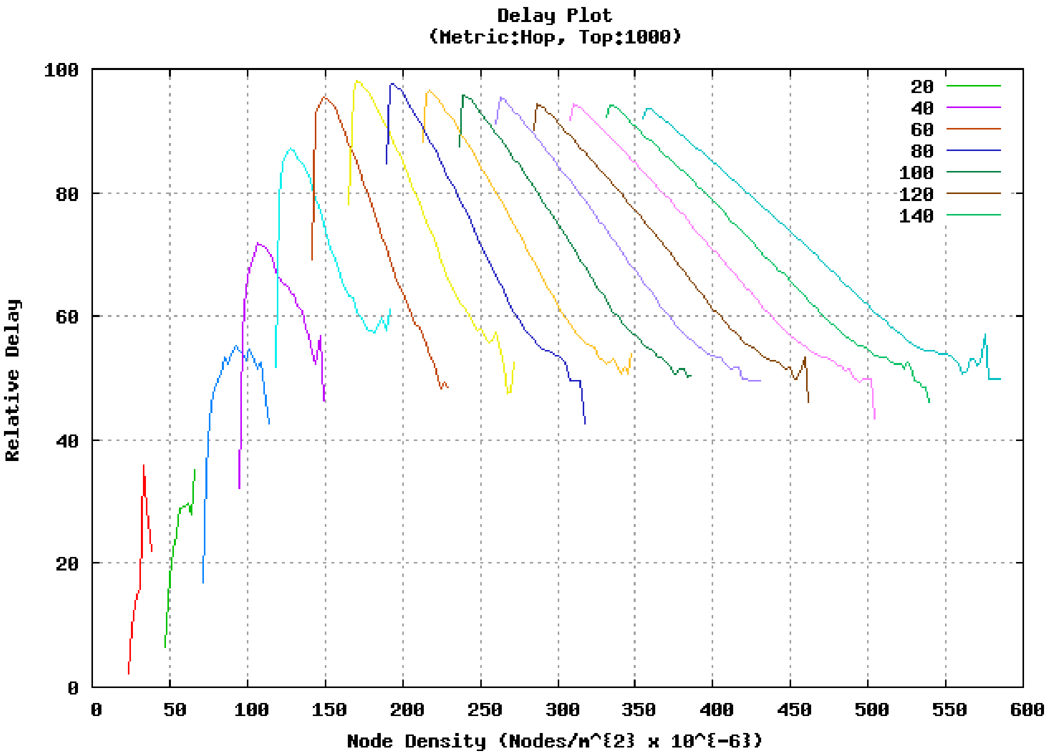


Figure 5-22: *MinHop* delay plot for increased fixed node density with relay nodes.

Again we take a closer look at individual fixed node densities to help clarify network performance characteristics. In Figure 5-23 we observe the effects on mean throughput performance for 100, 120 and 140 fixed nodes over 1000 topologies. We can see that the maximum throughput is approximately 0.1 Mbps for fixed nodes of 100 and above.

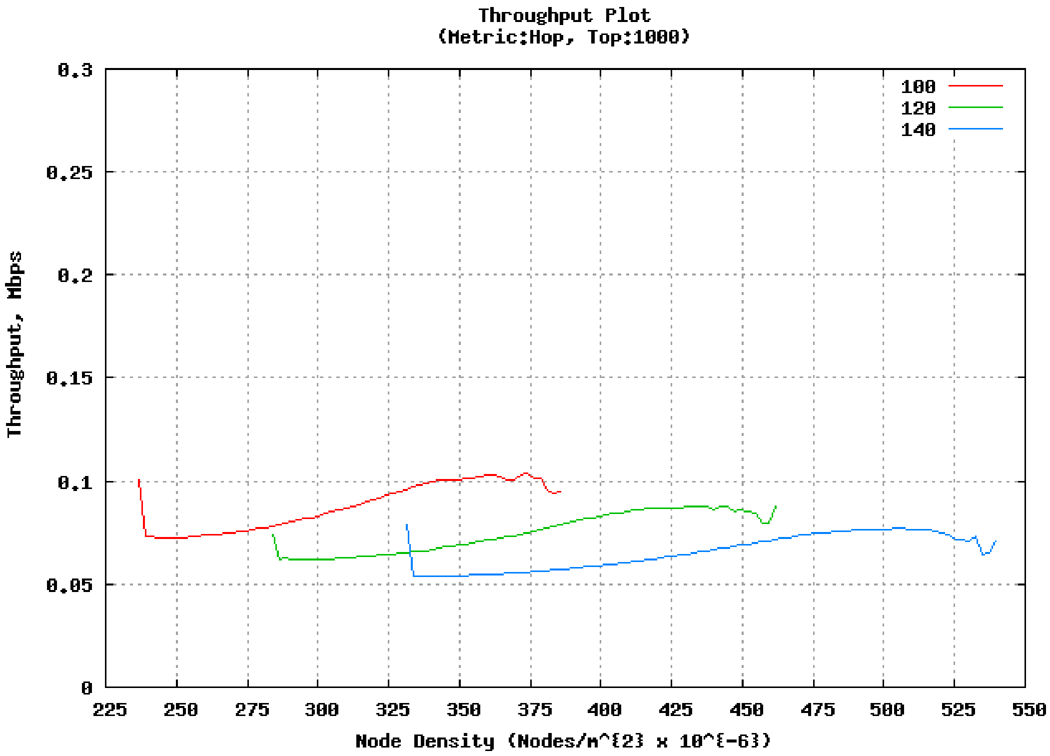


Figure 5-23: *MinHop* throughput plot with relay nodes for 100, 120 & 140 fixed nodes.

In Figure 5-24 we illustrate the case for a single topology with 100 fixed nodes. Similar to the results for ETT, we observe the stepped pattern as relay nodes are added. The gain in throughput is approximately 29% for an additional 55 relay nodes.

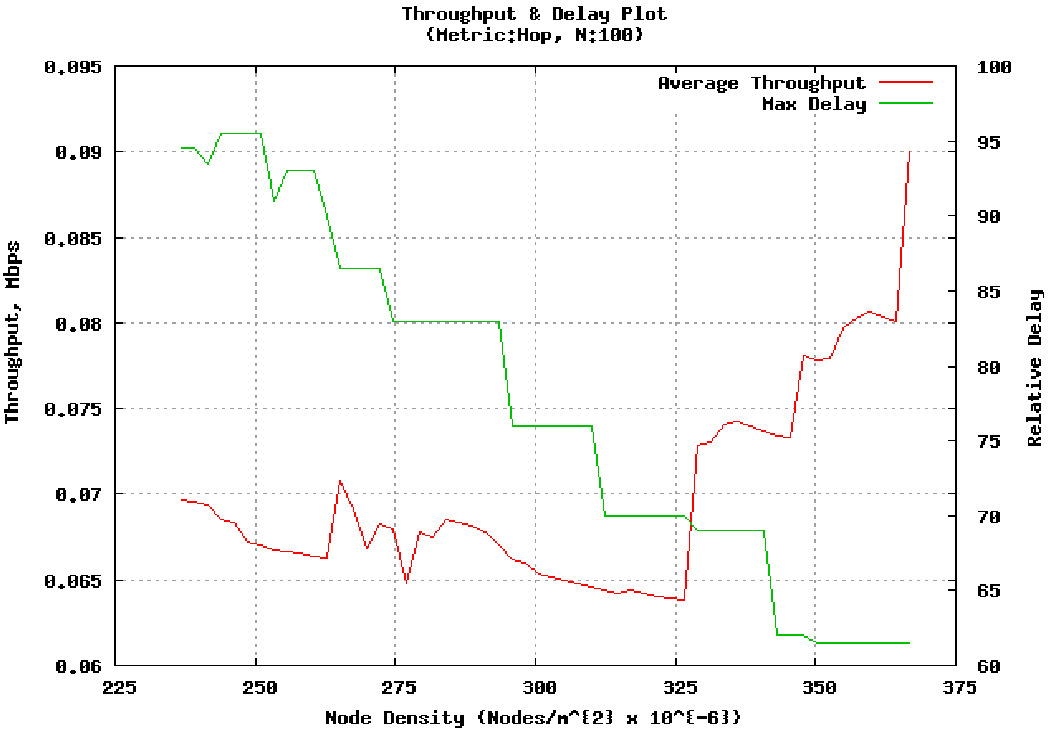


Figure 5-24: *MinHop* throughput and delay plot for a single topology with 100 fixed nodes.

As with ETT we use the peak PDF values to determine the gain (see Appendix C2 for PDF tables and plots). In the case of 100 fixed nodes the throughput gain is approximately 29% when the number of relay nodes peaks at, m_{peak} = 63 with a throughput of 0.09 Mbps. Due to the path selection when using *MinHop*, the number of 1 Mbps links is comparatively high, thus allowing for a greater number of relay nodes to be added. Table 5-3 compares the throughput gain, ρ and relay nodes, m_{peak} for eliminating 1 Mbps links when using both link cost metrics.

	m_{peak}	Throughput, Mbps	ρ
ETT	14	0.27	69%
MinHop	63	0.09	29%

Table 5-3: Percentage throughput gain for ETT and *MinHop*. Relay nodes are added to 1 Mbps links when 100 fixed nodes are used.

5.2.3 Adding Relay Nodes to Guarantee 11 Mbps Links (ETT)

As we have seen from the previous section, when Dijkstra's shortest path algorithm uses ETT it will generate fewer paths with 1 Mbps links. Therefore there are less links to optimise. We will expand upon the previous simulations by searching through all links in the network in order to place relay nodes so as to guarantee 11 Mbps rates (i.e. we continue adding relay nodes to the network until the multicast rate of each *Parent* is 11 Mbps). In Figure 5-25 and Figure 5-26 we present the network performance for throughput and delay. Each coloured line represents a specific fixed node density. As before, relay nodes are added and the network performance is recorded and plotted. The general characteristic of the throughput is consistent with the performance seen so far (i.e. the throughput decreases as the node density increases and the relative delay initially increases before decreasing at higher node densities).

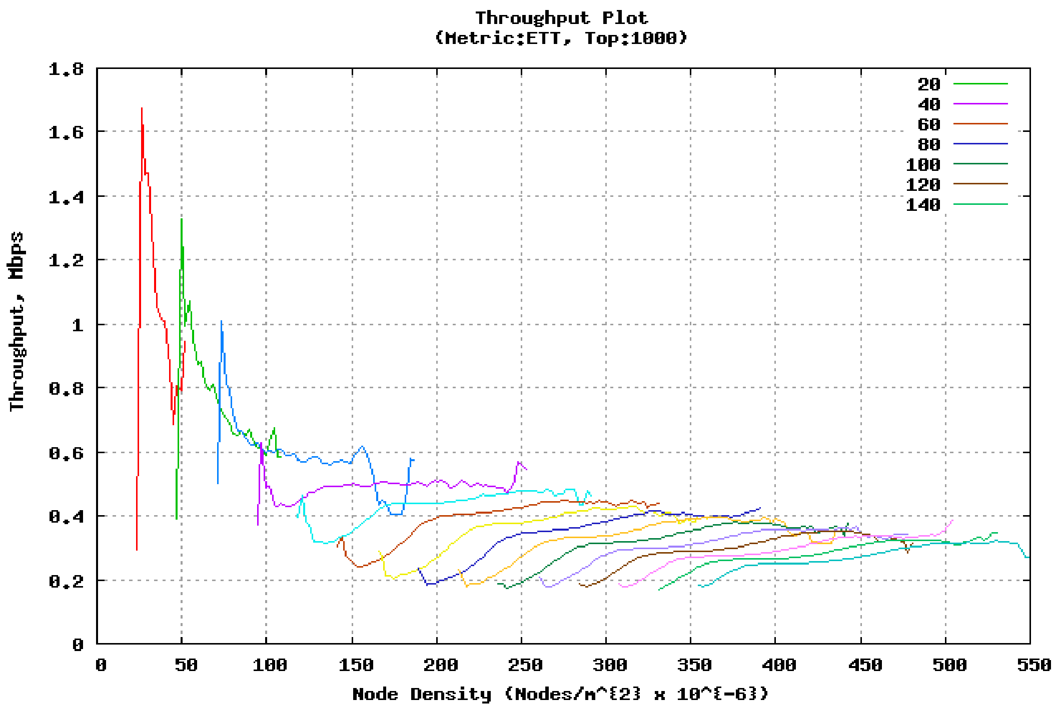


Figure 5-25: ETT throughput plot for guaranteed 11 Mbps multicast rate.

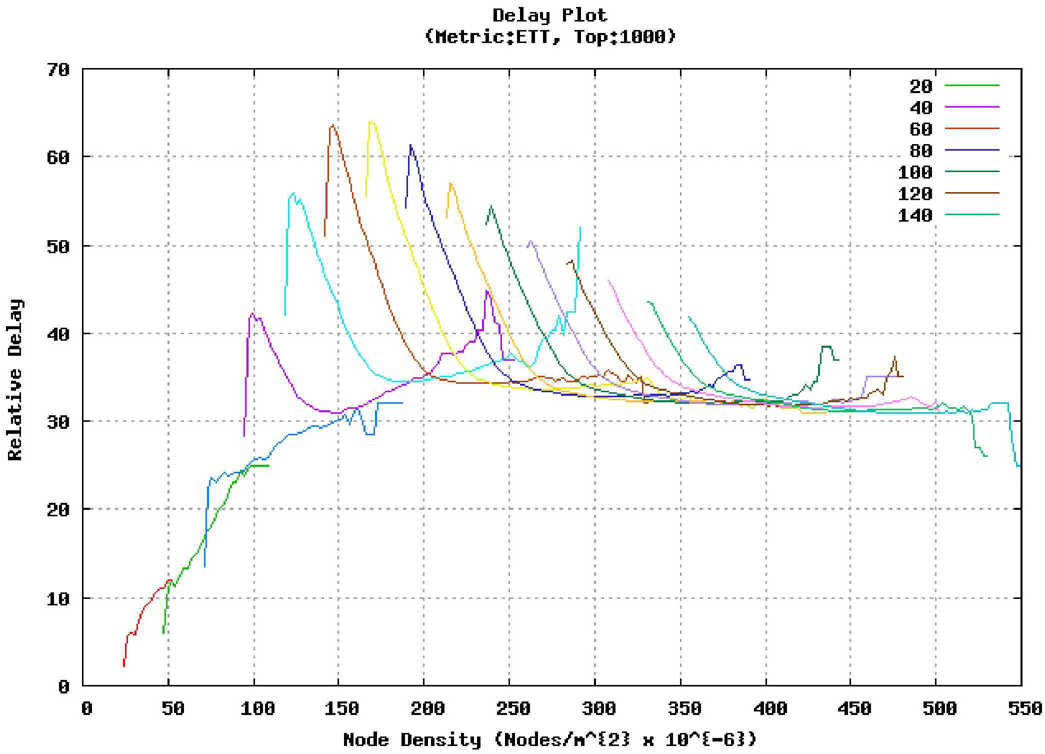


Figure 5-26: ETT delay plot for guaranteed 11 Mbps multicast rate.

To make clear the difference before and after relay nodes have been added, we present in Figure 5-27 the throughput and delay curves using only the data for relay nodes, m = 0 (dashed line) and m = max (solid line). We can see from the plots that the node density has increased by approximately 200 x 10^{-6} Nodes/m^2 at the upper fixed node density (i.e. when 150 fixed nodes are used). This corresponds to an increase of approximately 55%. The largest percentage node increase of approximately 150% occurs between (95 x 10^{-6} and 165 x 10^{-6}) Nodes/m^2 (i.e. 50 – 70 fixed nodes). A larger increase in the amount of relay nodes used is expected in this region due to node spacing and the operation of ETT. As the node density increases the connectivity and node coverage will improve as we have previously shown in section 5.1.3. When the fixed node density increases further the proximity of node placement will decrease allowing for Dijkstra's shortest path algorithm to select paths with a lower ETT (i.e. shorter links as we have seen in Figure 5-12). This will result in a reduction in the possibility for relay nodes to be added at higher densities.

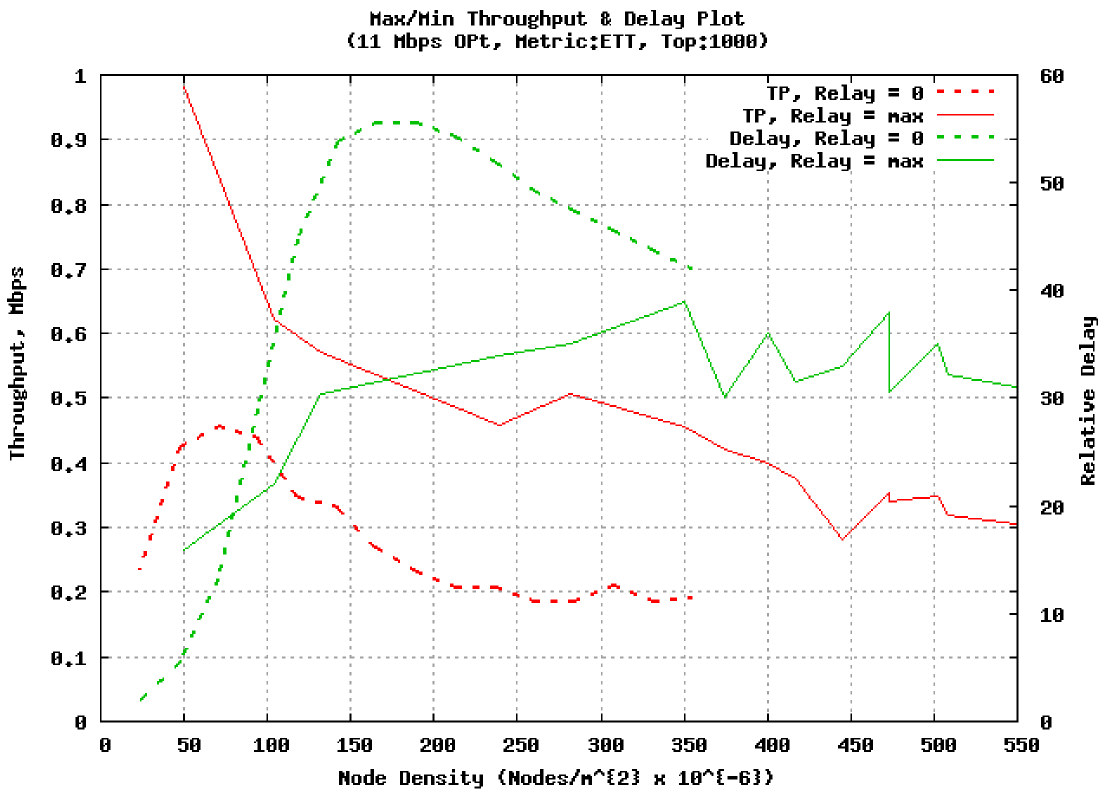


Figure 5-27: ETT Throughput and delay plot showing maximum and minimum relay nodes.

On closer inspection of the graph in Figure 5-25 we can see that there is an increase in the throughput as relay nodes are added to the network. In Figure 5-28 we plot the throughput results for 100, 120 and 140 fixed node densities as we add relay nodes. Each curve can be divided into three sections with two turning points. These three sections represent relay nodes being added at 1 Mbps, 2 Mbps and 5.5 Mbps links. If we use the peak probability values for throughput we can calculate the performance gain for adding relay nodes. For

100 fixed nodes we can obtain an increase in throughput of approximately 167% when the number of relay nodes is at a maximum (i.e. no more links remaining which require relay nodes) of m_{peak} = 90 with a throughput of 0.40 Mbps.

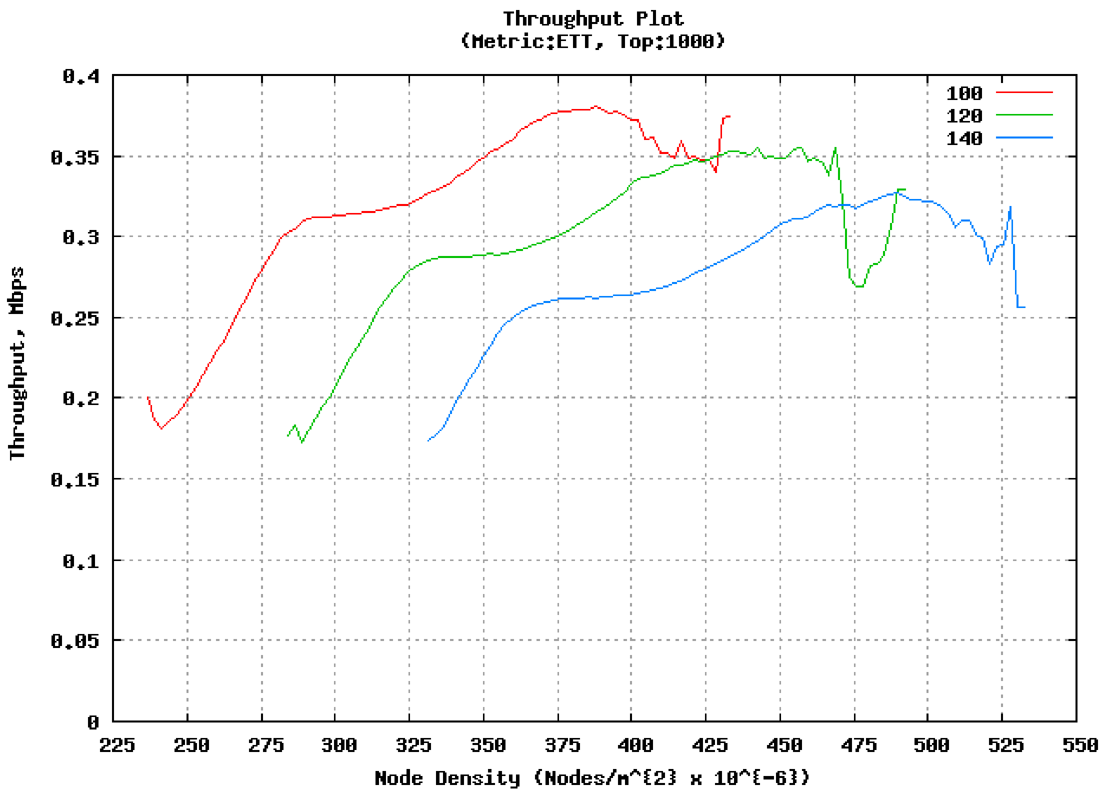


Figure 5-28: ETT throughput plot with relay nodes for 100, 120 & 140 fixed nodes.

5.2.4 Adding Relay Nodes to Guarantee 11 Mbps Links (MinHop)

In this section we present the results for our simulations using the link cost metric minimum hop (*MinHop*) while placing relay nodes to guarantee 11 Mbps links. As in the previous section each coloured line represents a specific fixed node density. In Figure 5-29 we present the graph for the throughput as relay nodes are added. The general characteristic for this plot (and the delay plot) are similar to those shown in section 5.2.2 when eliminating 1 Mbps links. However, as expected the overall node density has increased significantly.

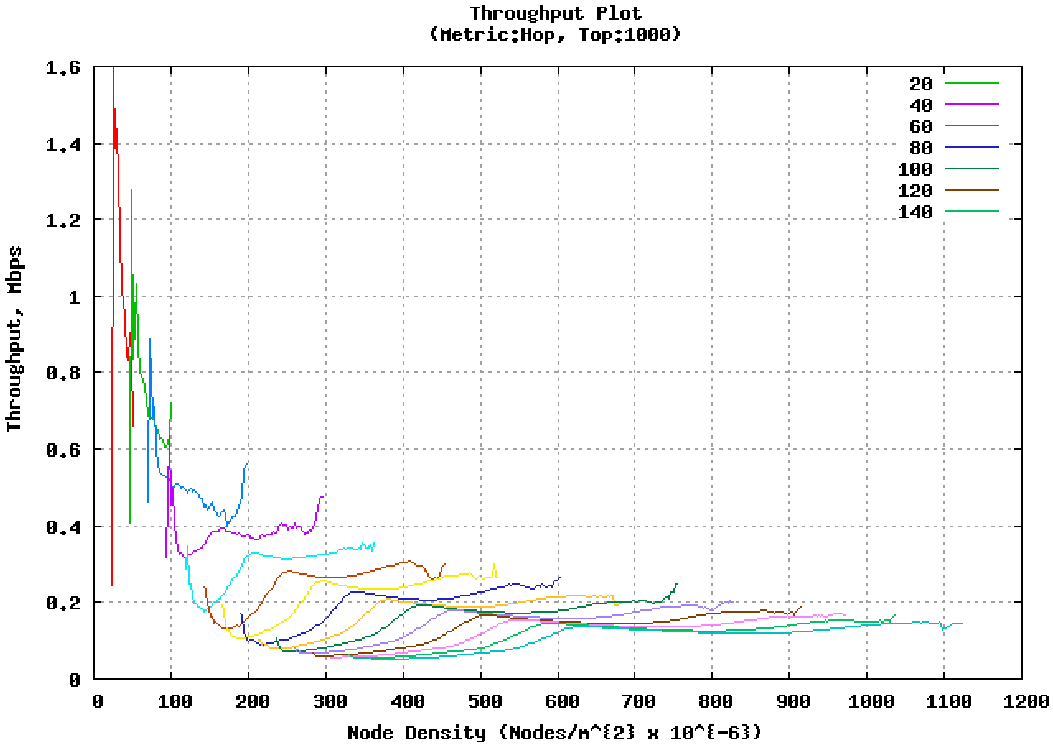


Figure 5-29: *MinHop* throughput plot for guaranteed 11 Mbps multicast rate.

Figure 5-30 illustrates the results for the delay performance. In Figure 5-30 we notice that the delay initially increases before a significant decrease. This is due to relay nodes initially having a negative effect on the network until the multicast rate has been increased. This effect can also be seen in the throughput plot but to a lesser extent (i.e. an initial drop in throughput before gradually increasing).

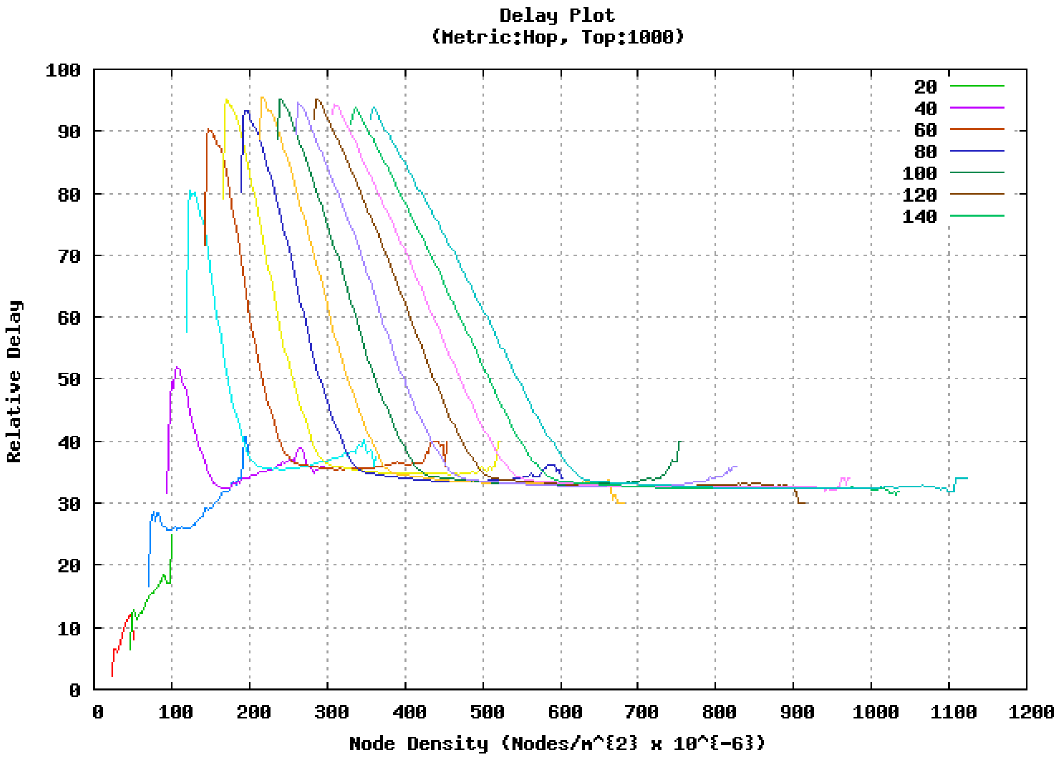


Figure 5-30: *MinHop* delay plot for guaranteed 11 Mbps multicast rate.

Again, to help make clear the difference in the network performance as relay nodes are added for each fixed node density we will plot the throughput and delay curves (Figure 5-31) using only the data for relay nodes, m = 0 (dashed line) and m = max (solid line).

We can see from the plots that the node density has increased by over 750 x 10^{-6} Nodes/m^2 at the upper fixed node density (i.e. when 150 fixed nodes are used). This amounts to a node density increase of approximately 215%. Unlike ETT, *MinHop* will almost always produce long links. Therefore, the percentage of relay nodes increases as the fixed node density increases.

If we use the PDF values (see Appendix C4) for throughput we can calculate the performance gain for adding relay nodes. For 100 fixed nodes we can obtain an increase in throughput of approximately 257% when the number of relay nodes is at a maximum (i.e. no more links remaining which require relays) of m_{peak} = 219 with a throughput of 0.25 Mbps.

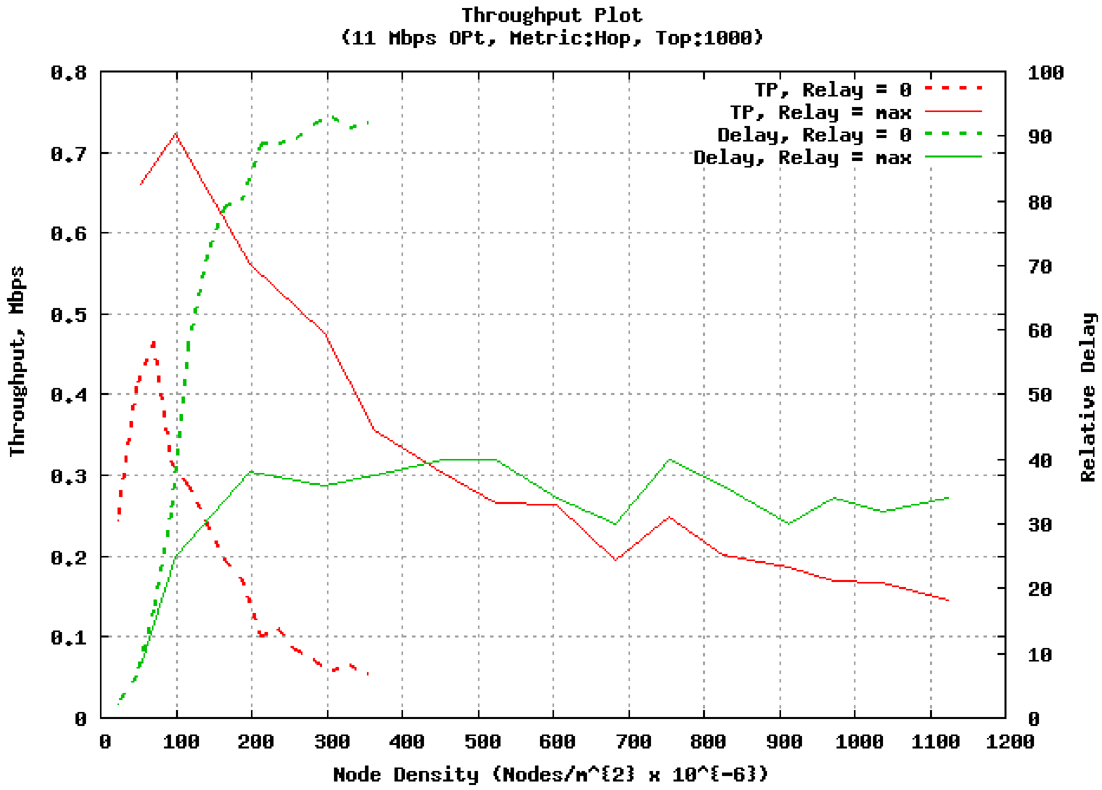


Figure 5-31: ETT Throughput and delay plot showing maximum and minimum relay nodes.

Table 5-4 compares the throughput gain, ρ and relay nodes, m_{peak} for guaranteed 11 Mbps multicast rates when using both link cost metrics.

	m_{peak}	Throughput, Mbps	ρ
ETT	90	0.40	167%
MinHop	219	0.25	257%

Table 5-4: Percentage throughput gain for ETT and *MinHop*. Relay nodes are added to 100 fixed nodes for guaranteed 11 Mbps multicast rate.

5.2.5 Summary

In the previous sections we presented the results for adding relay nodes to a network. We presented throughput and delay performance for the link cost metrics ETT and *MinHop*. We began by adding relay nodes to eliminate 1 Mbps links on the network with the intention of improving the multicast rate. We further extended upon this by continuing to add relay nodes in order to provide 11 Mbps multicast rates throughput the network. We presented the performance gains for the case of 100 fixed nodes for both 1 Mbps and 11 Mbps optimised rates and compared the gains for each link cost metric.

We present the combined results from the previous sections in Table 5-5 for convenience. As we are using an example of 100 fixed nodes the value for the maximum relay nodes added, m_{peak} will also represent the percentage of nodes added. We can see from the table below that when we add relay nodes to eliminate 1 Mbps links, ETT will yield larger gains than *MinHop* using 4.5 times less relay nodes. Using ETT will also provide higher throughputs as was the case in the original un-optimised network performance.

When we continue to add relay nodes in order to provide 11 Mbps multicast rates throughout the network, we observe that *MinHop* can produce a higher gain in throughput than ETT. However, ETT continues to outperform *MinHop* with regard to the mean network throughput (approximately 2 times higher) and requires significantly less relay nodes (approximately 2.5 times less).

	m_{peak}	**Throughput, Mbps**	ρ
	Eliminating 1 Mbps links		
ETT	14	0.27	69%
MinHop	63	0.09	29%
	Guaranteed 11 Mbps rates		
ETT	90	0.40	167%
MinHop	219	0.25	257%

Table 5-5: Comparison of performance gains from previous sections.

By provisioning relay nodes in a deliberate and specific manner we have shown that it is possible to improve the network performance with regard to throughput and delay. By placing nodes in key locations we can improve the multicast rate by targeting links with low link rates. The number of nodes required to improve the performance gain can be seen to be considerably less than that suggested by Gupta and Kumar in [GuK00]. However, it should be noted that the quantity of nodes required to provide such performance gains; in the case of ETT is almost equal to the number of fixed network nodes, or in the case of *MinHop* over double the number of fixed network nodes.

5.3 Power Optimisation Results

In this section we present the results for the power control algorithm described in section 4.3. Based on previous results we have chosen the link cost metric ETT in our simulation. A set of simulations is conducted using the same settings as outlined in section 5.1 with the following changes;

Tx Power: The initial starting power is fixed for all nodes. The power control algorithm adjusts the transmit power on a per node basis. Series of simulations are run using starting powers of 0 dBm – 18 dBm with increments of 1 dBm.

Link Cost Metric: ETT will be used throughout all simulations.

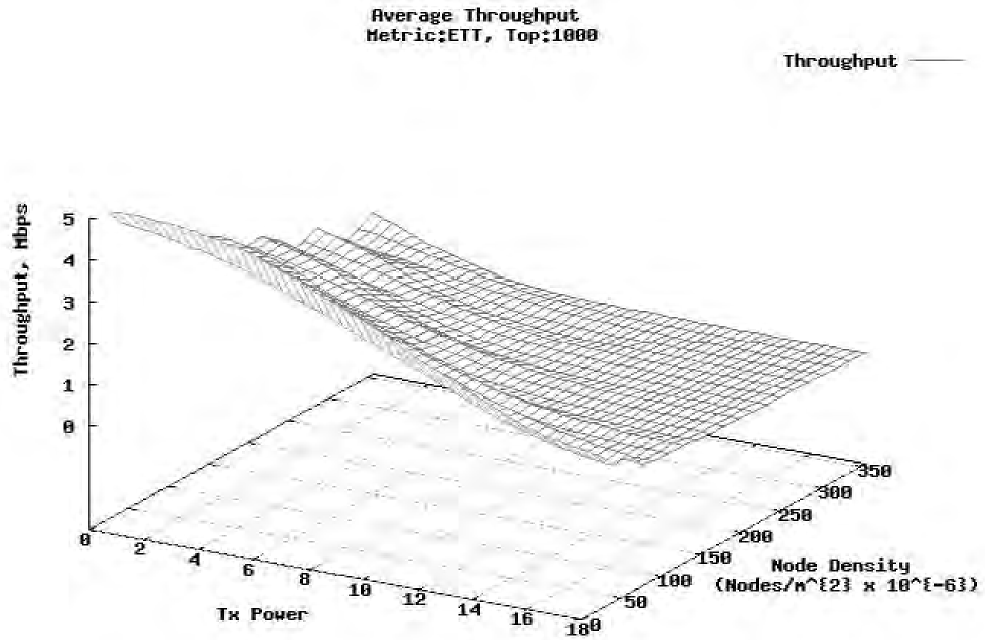
Optimal Tree: A combination of Dijkstra’s shortest path algorithm and our power control algorithm will be used to search for an optimal tree within each topology.

5.3.1 Power Optimisation using ETT

We begin our analysis by examining the results from simulations using the power optimisation algorithm and comparing them to results without any optimisation techniques used. Recall that the method for controlling the transmit power and subsequent search for an optimal spanning tree is described in section 4.3 and section 4.4.

As one of our main objectives is to develop a method of improving the multicast throughput (and hence the overall tree throughput) we will begin by examining the behaviour of the mean throughput across all power settings (0 dBm to 18 dBm, in 1 dBm steps) as the fixed node density is increased. Figure 5-32 illustrates the throughput using ETT. The upper diagram in Figure 5-32 plots the throughput for increasing transmit power and increasing node density. The lower diagram displays the contour map for the same results. The results for the mean network throughput without optimisation are given in Figure 5-33.

When we compare the throughput plots below, the most notable differences are at the lower density and lower transmit power levels. The throughput performance at the opposite end of the scale (i.e. high density and high transmit power) is closely matched. When we increase the node density or transmit power we are effectively increasing the transmission contention. The reduced throughput at higher node density and transmit power is expected due to the fact that our throughput performance metric is a function of the contention. Although Figure 5-33 appears to be uniform it does in fact follow the same general characteristic as Figure 5-32. However, due to the scaled performance improvement of Figure 5-32 the difference at the lower transmit power and node density is less significant in comparison. (The difference can be seen in the un-optimised results in Figure 5-10). We will examine the throughput performance in more detail in section 5.3.2.



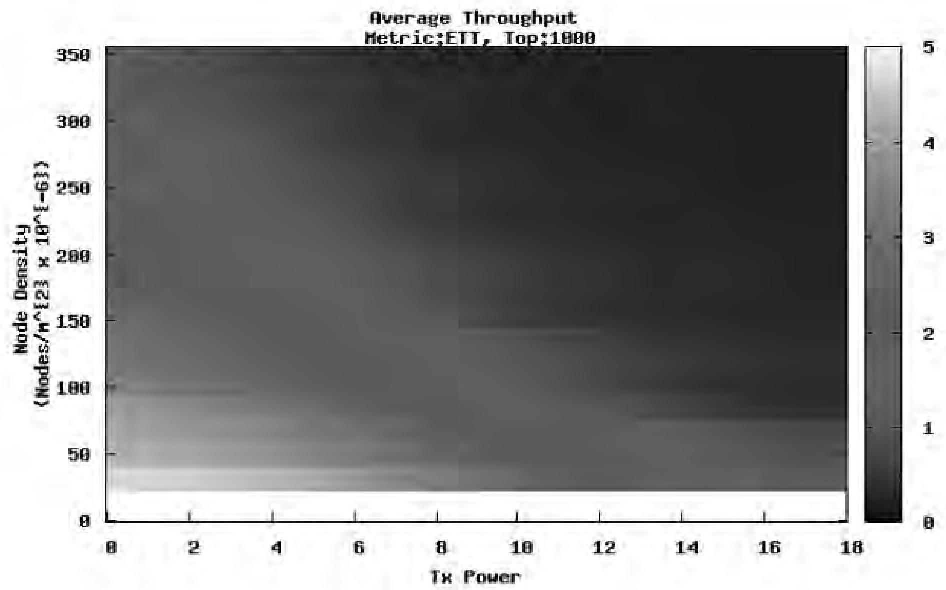
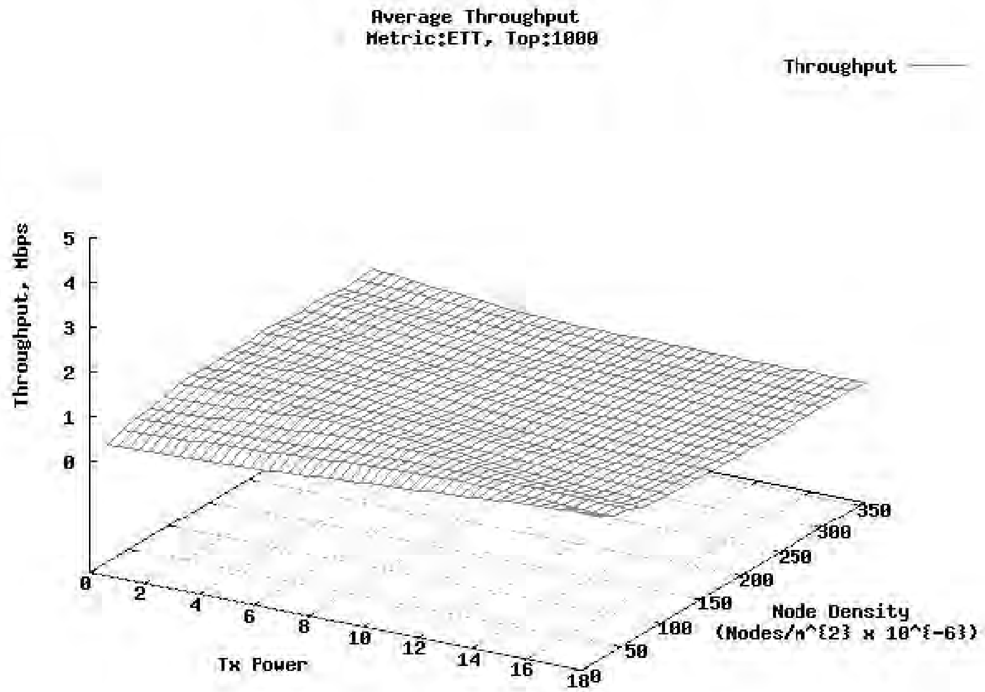


Figure 5-32: Throughput plot using power control algorithm. Upper: 3D plot. Lower: contour map.



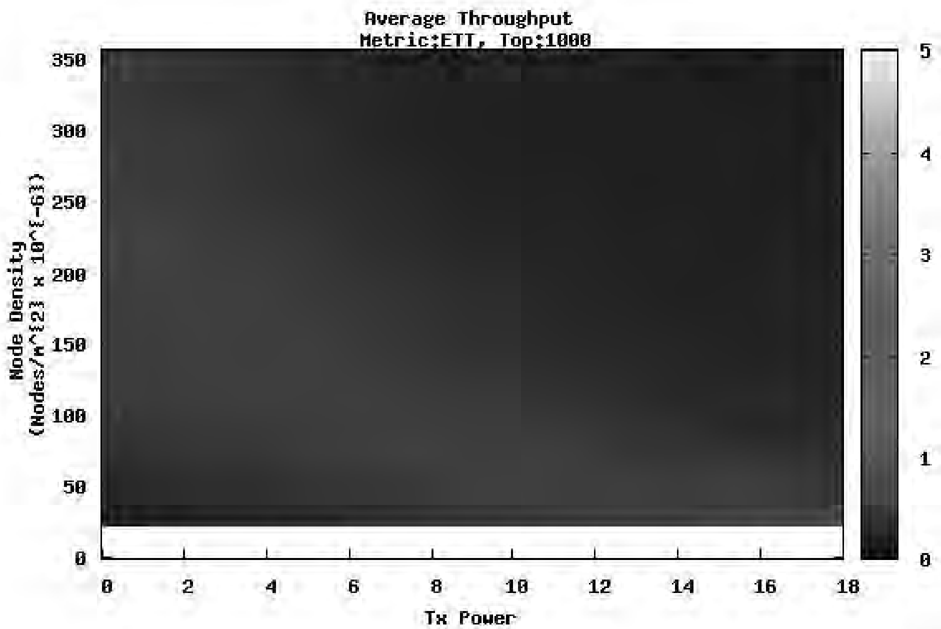
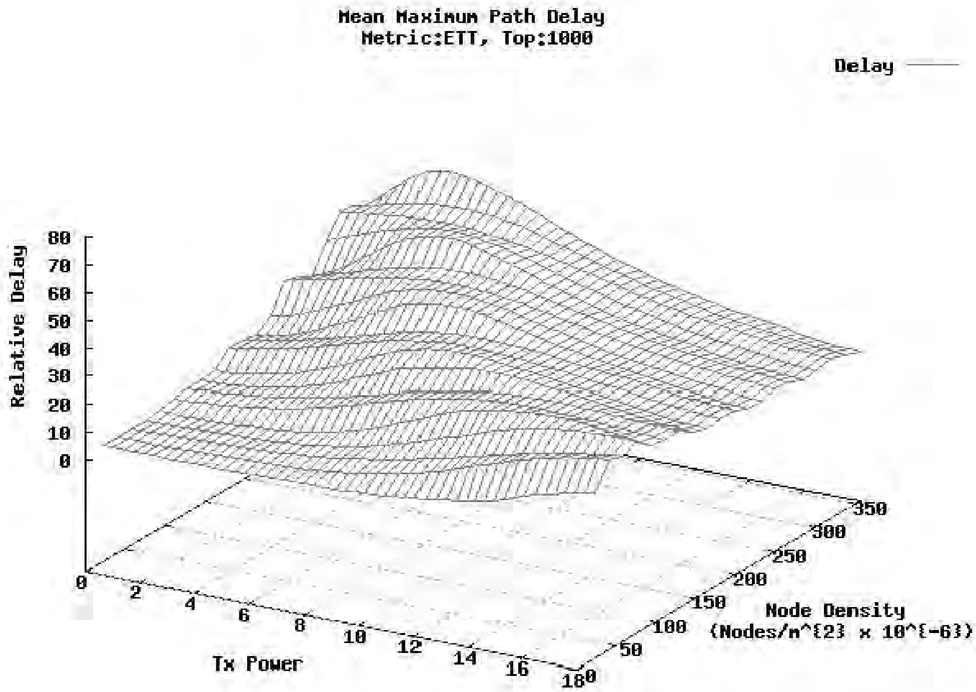


Figure 5-33: Throughput plot without optimisation. Upper: 3D plot. Lower: contour map.

Next we will briefly examine the network performance for the delay when comparing the un-optimised network to one using our power control algorithm. As done previously, we will examine the performance across all power settings (0 dBm to 18 dBm, in 1 dBm steps) as the fixed node density is increased. Figure 5-34 illustrates the network delay performance when power control is used while Figure 5-35 illustrates the network delay performance without any optimisation.

In both figures, the light colours in the contour maps represent concentrations of relatively high delays. The performance of the network when using the power control algorithm appears reduced in comparison to the un-optimised network. This can be seen by the lower peak levels in the upper graph in Figure 5-34 when compared to Figure 5-35. In general, using the power control algorithm reduces the network delay across all transmit power levels and node densities. However, a significant difference in delay can be observed for the reduction of peak values (i.e. the delay is greatest at high density and low transmit power. Relatively high delays are also observed when the density and transmit power are varied inversely to each other, e.g. decreasing the density and increasing the power).



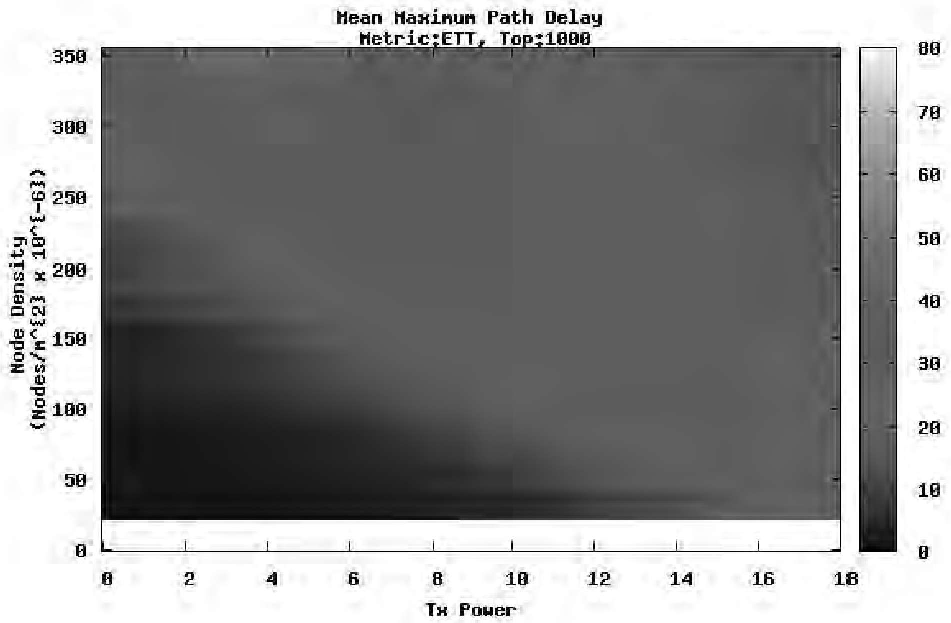


Figure 5-34: Delay plot using ETT with power control algorithm.

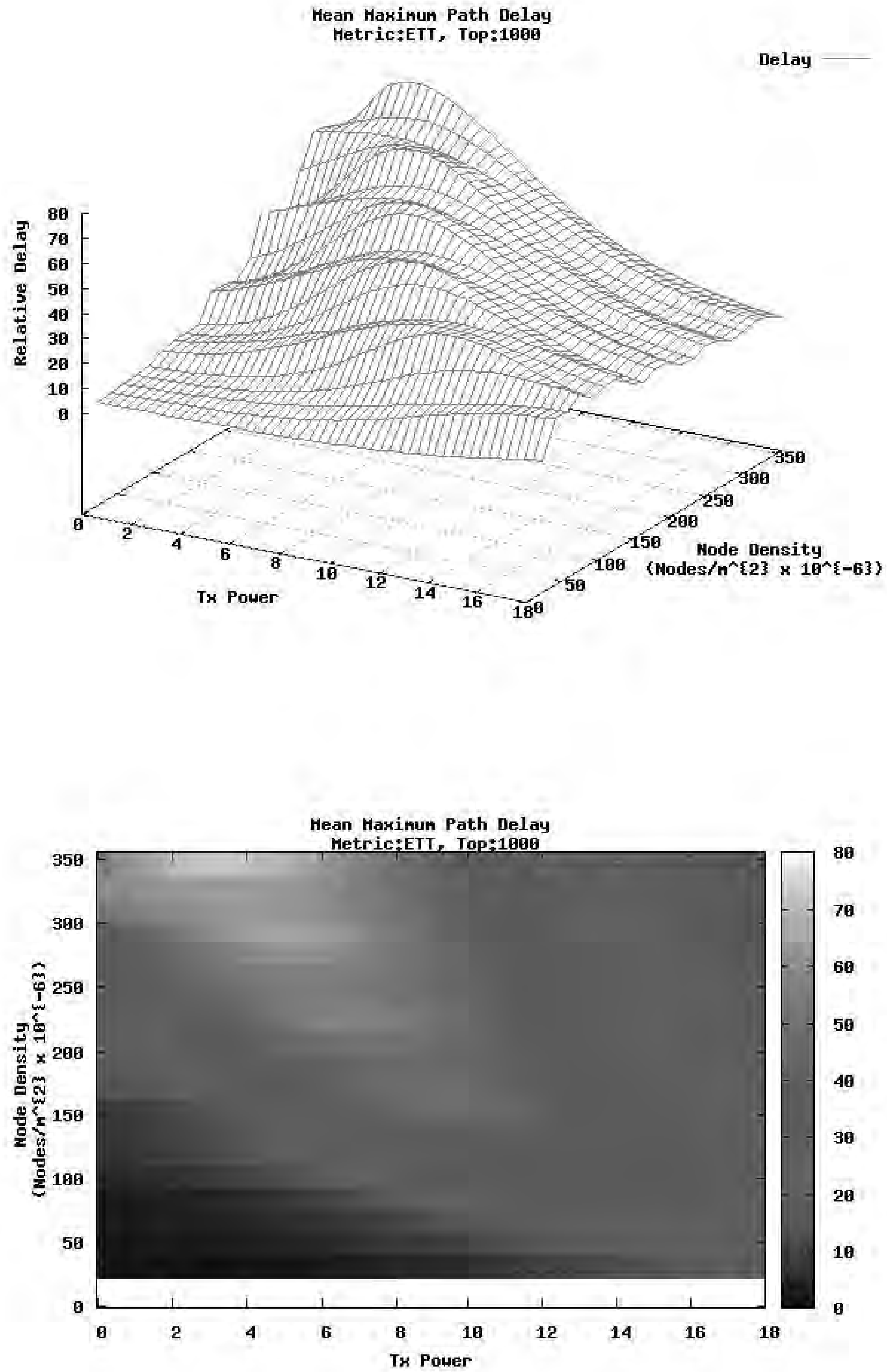


Figure 5-35: Delay plot using ETT without optimisation.

5.3.2 Detailed Analysis (Throughput and Delay)

In this section we will describe the performance of the network in terms of throughput and delay when using the power control algorithm. To achieve this we will observe the effects of the network performance for a particular initial power setting and fixed node density. In the following example we use 100 nodes (i.e. a fixed node density of 236×10^{-6} Nodes/m^2) with an initial transmit power of 9 dBm for all nodes. In Figure 5-36 we illustrate the network constellation before and after the use of the power control algorithm.

The most significant change in the diagram after the use of the power control algorithm is the elimination of the 1 Mbps links (green coloured) and the increase in the number of 11 Mbps links (red coloured). Less obvious from the diagram are minor changes in the routes taken by nodes (i.e. *Child* nodes switching to a new *Parent*). As we are using successive runs of Dijkstra's shortest path algorithm after each power adaptation, all paths will again be searched in order to find an optimal tree.

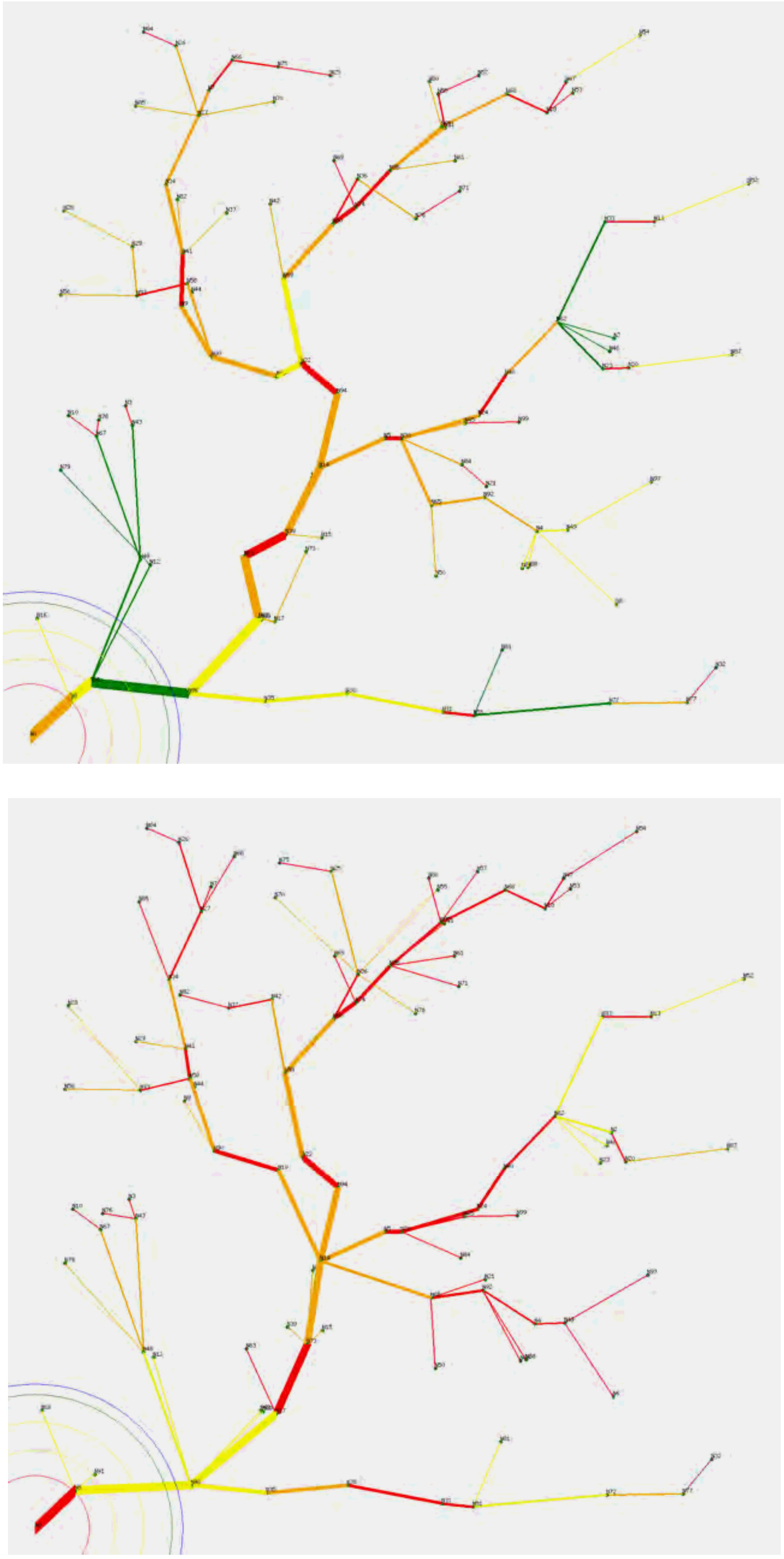


Figure 5-36: Comparison of network diagram before and after using power control algorithm. The upper diagram represents the original network and the lower diagram represents the network after using the power control algorithm.

In Figure 5-37 we present the results when an initial transmit power of 9 dBm is set for all nodes. Solid lines indicate the performance after power control is applied and an optimal tree has been found. As can be seen in the overall plot for all power settings in Figure 5-32, the throughput is relatively higher for lower node densities. As the node density increases from the initial starting value to approximately 150 x 10^{-6} Nodes/m^2 the throughput decreases monotonically. At this point the working plane becomes densely populated with nodes resulting in an increase in contention. As we have seen previously and again here, the node density is a major contributing factor to the performance of the network. While the density is low, the power control algorithm is able to take full advantage of its ability to increase the transmit power and improve the multicast rate. This is due to nodes being placed at greater distances from their neighbours resulting in lower received power which results in a lower link rate. The power control algorithm increases the power of *Parent* nodes thus increases the received power of the *Child* node.

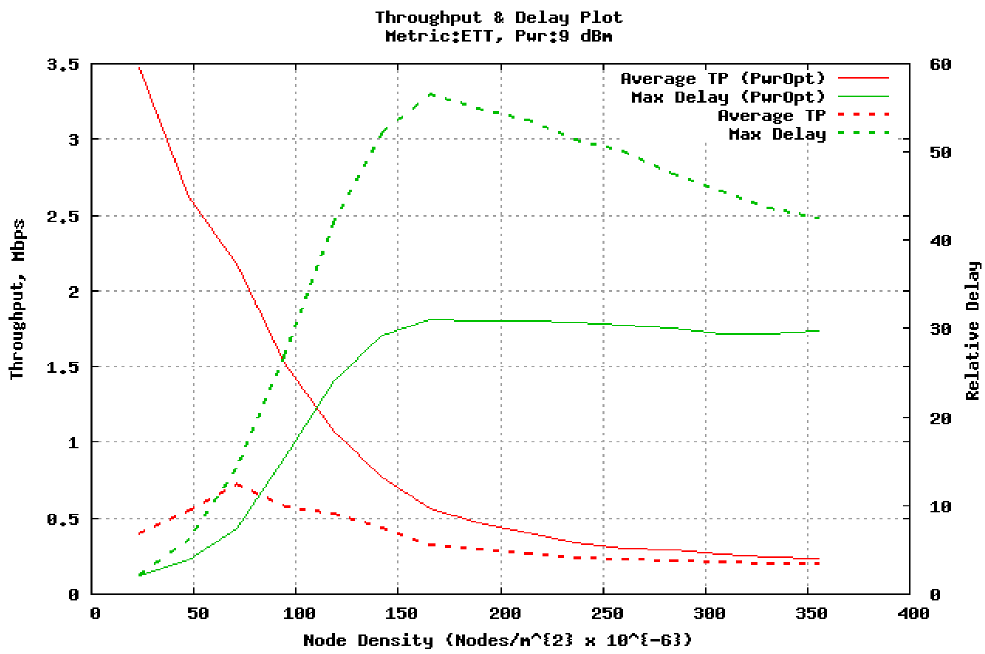


Figure 5-37: Throughput and delay performance comparison.

As the density increases beyond 150 Nodes/m^2 x 10^{-6} there is less opportunity for the algorithm to improve the multicast rate by increasing the transmit power. In fact, increasing the transmit power excessively at this point would typically result in a reduction in the throughput due to an increase in the number of neighbours coming into interference range of a *Parent* node. For this reason we have allowed our algorithm to back off (i.e. reduce) the transmit power to a lower value if a gain in throughput is not achieved. This is reflected in the delay performance of the network. We can see in Figure 5-37 that when the density reaches 150 x 10^{-6} Nodes/m^2 the delay begins to level off at approximately 30 units. It should be noted that the relative delay will naturally begin to decrease as the density increases. This is due to an increase in the availability of higher link rates for path selection (see section 5.1.1). This can be seen in the delay curve for the un-optimised network in Figure 5-37 (i.e. the dashed green line). Recall from previous results that more paths will be formed as the node density increases which will initially result in an increase in network delay. As the density increases further, nodes will be placed closer together resulting in shorter hops with increased data rates and hence reducing the delay. There are two contributing and opposing factors with increased node density; shorter paths with higher link rates and increased contention. Shorter paths with higher link rates will decrease the delay (and increase throughput). However, this will be negated by increased contention which will increase delay (and decrease throughput).

5.3.3 Detailed Analysis (Contention and Power)

We can see in Figure 5-38 how the path contention increases monotonically with the node density. The increase in contention can be observed in both the un-optimised network and the network when power adaptation is used. However, as the density increases the power control algorithm allows the *Parent* node to reduce its transmit power when a gain in throughput is not achieved. This can be clearly seen in Figure 5-38 (solid line), as the node density increases beyond 150 x 10^{-6} Nodes/m^2 the path contention for the power adapted network is approximately 100 units less than the original network.

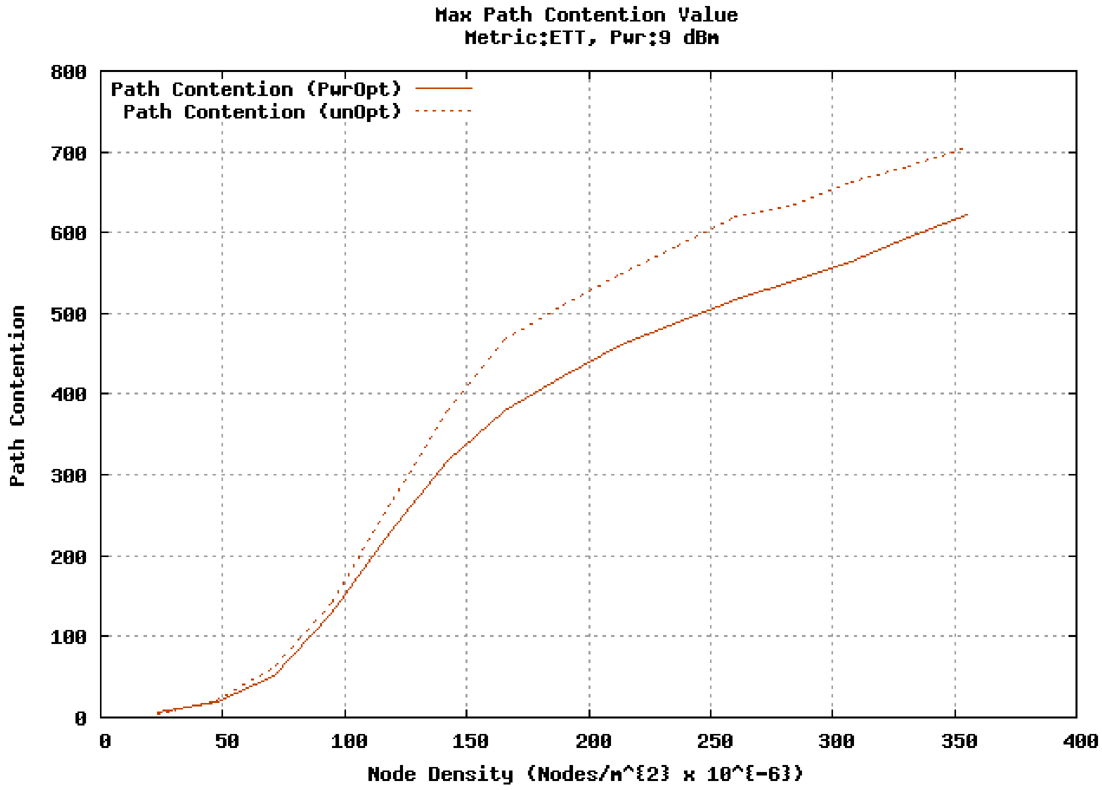


Figure 5-38: Path contention comparison using the power control algorithm.

If we plot the mean power for this same network we can observe how the transmit power and the network react to the change in node density. Figure

5-39 illustrates the transmit power (Ptx) of a network with an initial starting power of 9 dBm. As we can see from the diagram, the mean network transmit power varies between approximately 9 dBm and 10 dBm. There are two turning points in the curve, both of which occur for the same reason. When the node density is low few paths will be formed due to poor connectivity. The power adaptation will attempt to increase the power in order to improve the multicast throughput. If this cannot be achieved the power will reduce which results in the first dip in the curve. As the node density increases, the connectivity improves allowing more paths to be formed. This allows the power control algorithm to increase the transmit power in order to improve the multicast rate. Once the node density (and consequently transmit power) has reached a critical point (in the case of this example, a node density above 150 x 10^{-6} Nodes/m^2) there are less opportunities to increase the transmit power (due to a densely packed network with shorter paths), therefore the power control algorithm falls back to a lower value in order to reduce contention.

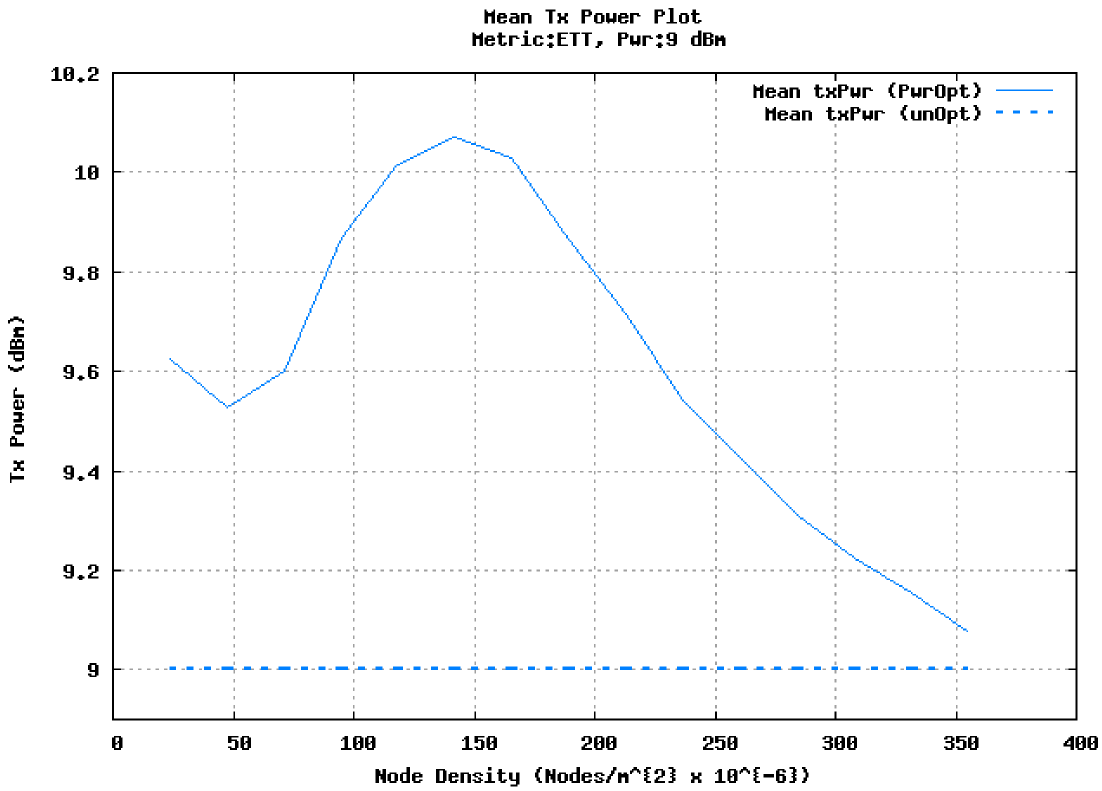


Figure 5-39: Mean transmit power (*Ptx*). An initial starting power of 9 dBm is used for all nodes.

This peak in the characteristic can be seen to shift depending on the initial transmit power. A complete set of mean transmit power plots for each power setting can be found in Appendix D. Therefore we can observe that the critical point in a network (with regard to tuning the transmit power) will depend on the initial starting power and node density which ultimately means the connectivity. As an example we provide the mean transmit power using an initial transmit power of 6 dBm and 12 dBm in Figure 5-40 and Figure 5-41 below.

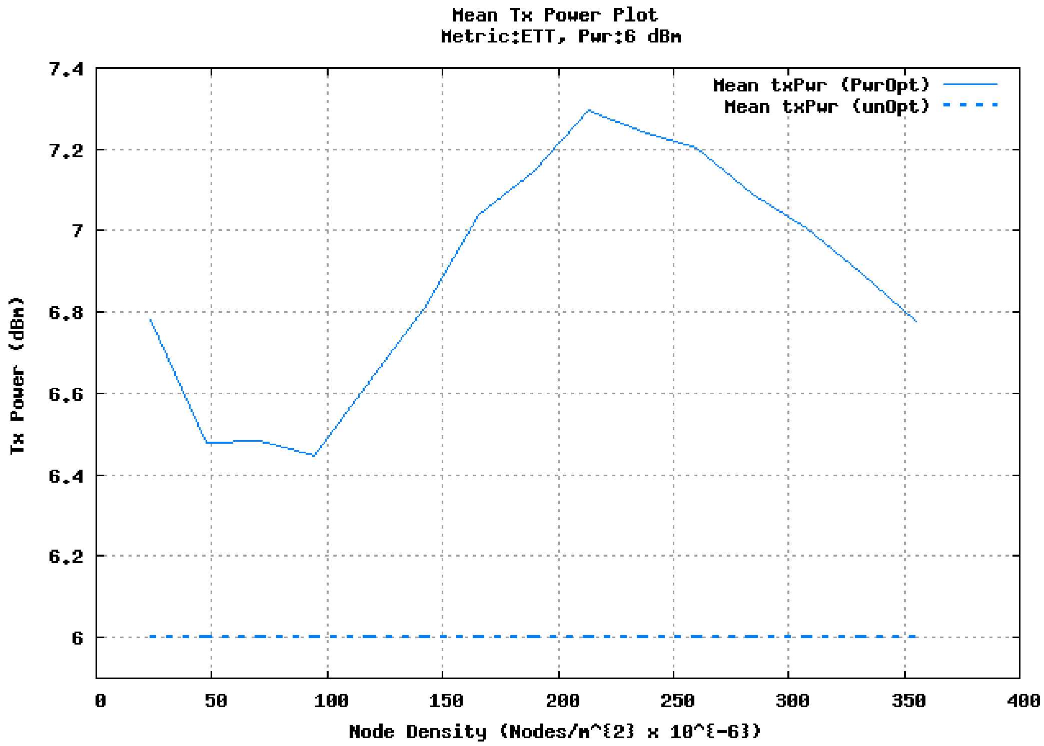


Figure 5-40: Mean transmit power for 6 dBm initial power.

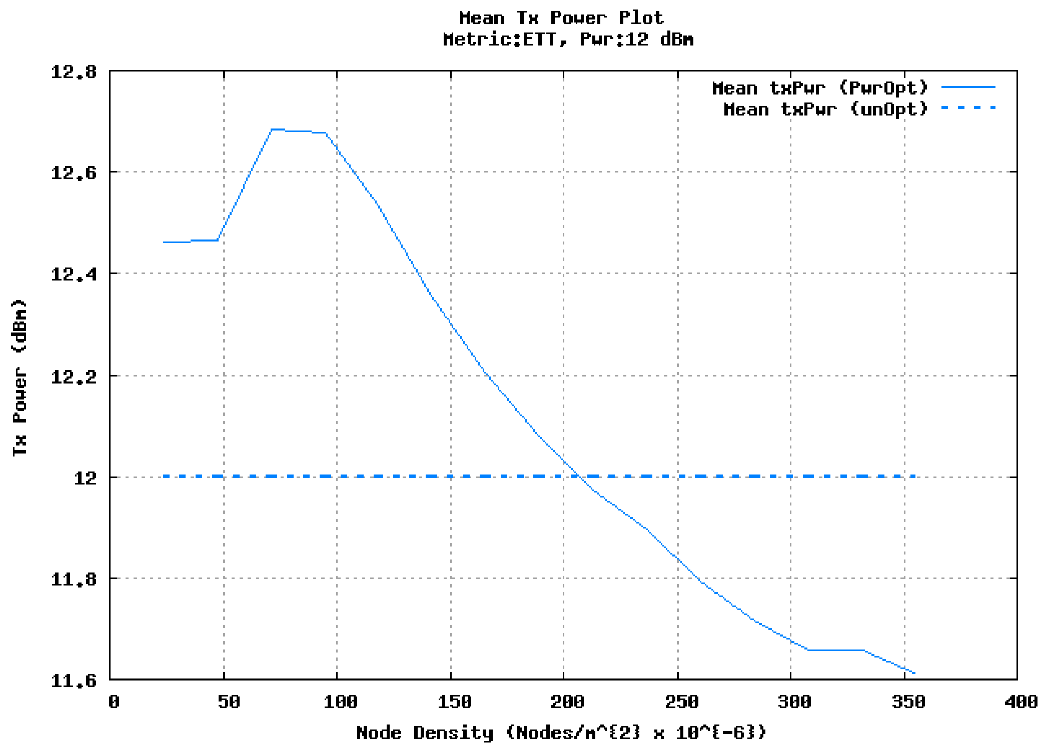


Figure 5-41: Mean transmit power for 12 dBm initial power.

5.3.4 Detailed Analysis (Node Coverage)

So far we have concentrated on the performance of relaying traffic across high throughput paths on the network by improving the multicast rate. We have shown that there is a significant gain in throughput performance when the node density is relatively low (see Figure 5-32 and Figure 5-37). However, this throughput gain can be misleading if we do not take into account the connectivity and node coverage under these conditions.

In Figure 5-42 we present the percentage node coverage using each of the available PHY rates with an initial power of 9 dBm. Solid lines represent the coverage for the power adapted network while dashed lines represent the original un-optimised network. From the diagram we can see that using the power control algorithm improves the coverage for all available rates over all node densities. When we use the power control algorithm, nodes which were initially out of communication range are now be capable of receiving transmissions of a nearby neighbour. This typically occurs at the boundary edges of the working plane. These nodes will now become *Leaf* nodes (i.e. non forwarding last hop nodes). In general such nodes will consequently have a low link rate due to their proximity to the *Parent*. In such cases, the mean network throughput will appear to be artificially low.

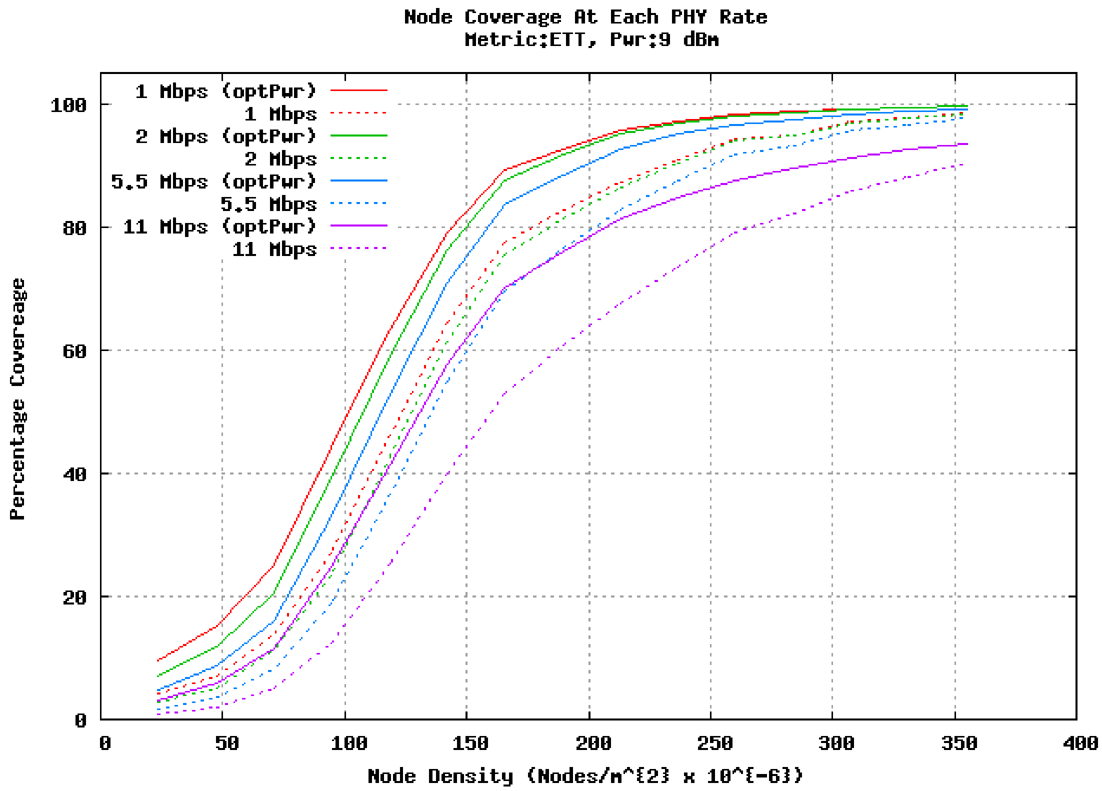
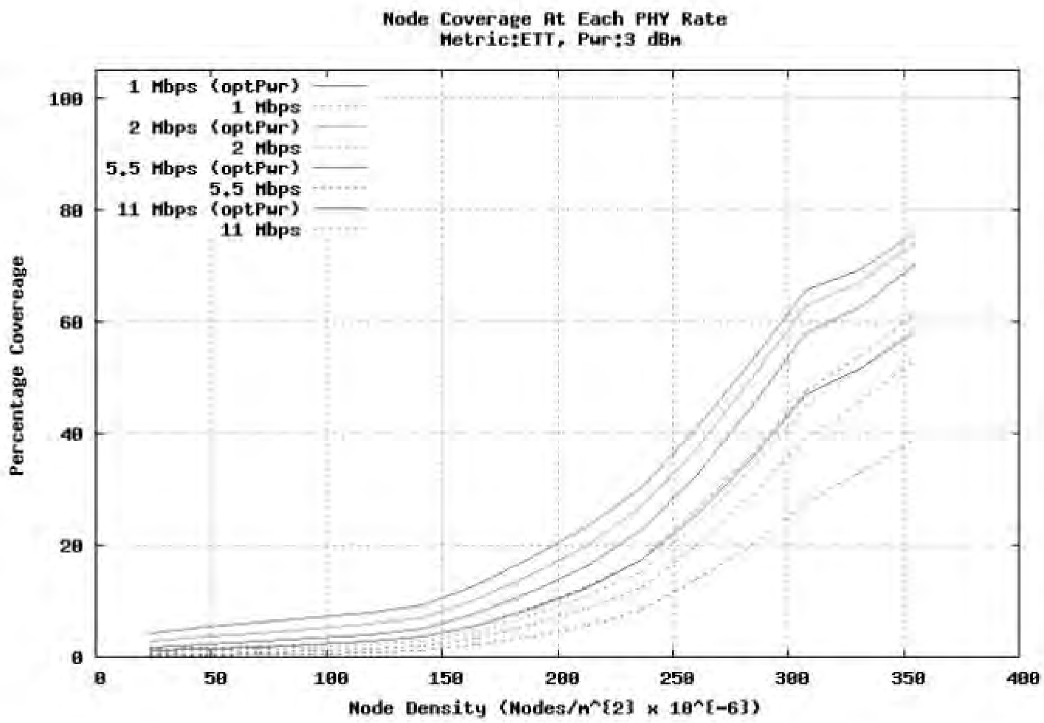


Figure 5-42: Comparison of node coverage for each PHY rate. An initial starting power of 9 dBm is used.

We also note that the coverage is relatively poor below 150 x 10^{-6} Nodes/m^2. This is due to poor connectivity as a result of the transmit power being too low to develop paths in a low density network. We further highlight this by showing two extreme cases when the transmit power is initially set relatively low and high (i.e. 3 dBm and 18 dBm respectively). In Figure 5-43 we compare the difference between both of these instances. We can see from the figure that using a power setting which is relatively low will result in poor node coverage. We can opt to use a high initial power setting to achieve optimal node coverage as shown in the figure when 18 dBm is used. However, as we observed from the throughput and delay results this would not yield a high performing network. Likewise, if the initial node throughput is set relatively low we can obtain higher throughputs however, the node coverage will be insufficient.



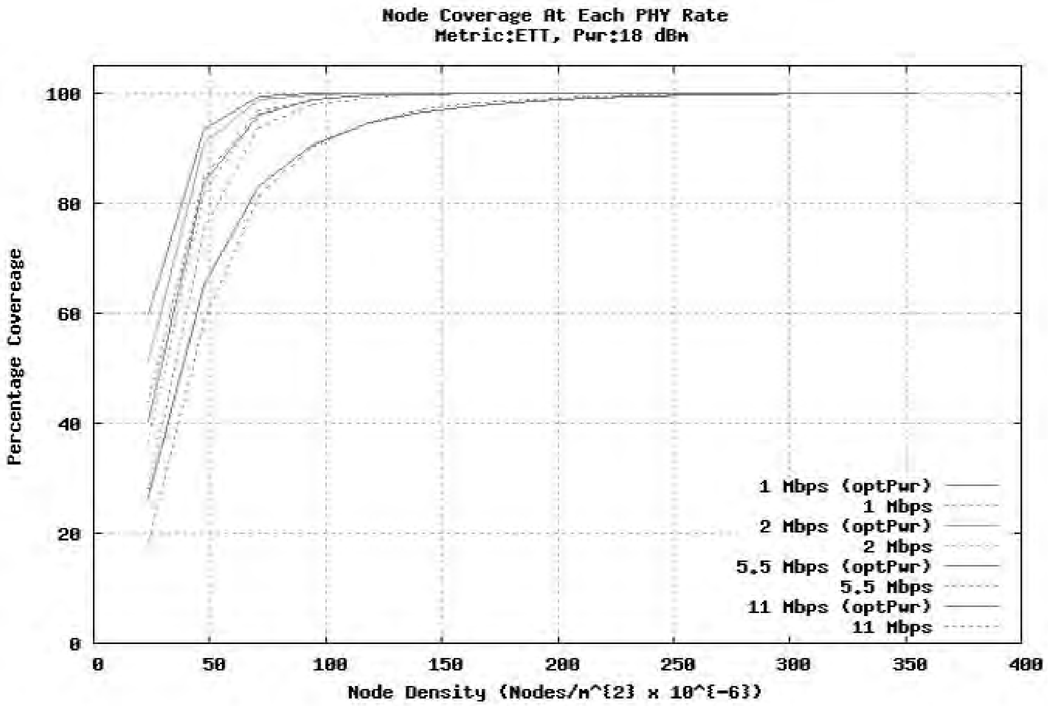


Figure 5-43: Comparison of node coverage with low and high percentage coverage. Low coverage achieved when initial power is 3 dBm (upper) compared to that of 18 dBm (lower).

5.3.5 Summary

In this section we have presented the results when using the power control algorithm as described in section 4.3. We first present the results for a range of initial power settings before selecting an individual case using 9 dBm. We observed that the power control algorithm works best at creating high throughput multicast branch rates by increasing the power on a per node basis when the node density is relatively low. As the node density increases there is less opportunity to increase the throughput due to increased contention. At this stage the power control algorithm makes use of its power fall back mechanism in order to reduce the transmission contention and hence increase the throughput.

As a result of the increased multicast rate the delay performance of the network will also show an increase (i.e. an overall decrease in the maximum relative path delay). The relative path delay is based on the path rate and the maximum path delay. As such, the performance will not display the same large gains as can be seen for the throughput when the node density is low. Figure 5-44 illustrates the percentage gain of throughput and decreased delay when using an initial power of 9 dBm. A full set of plots for each power setting (0 dBm to 18 dBm) can be found in Appendix D.

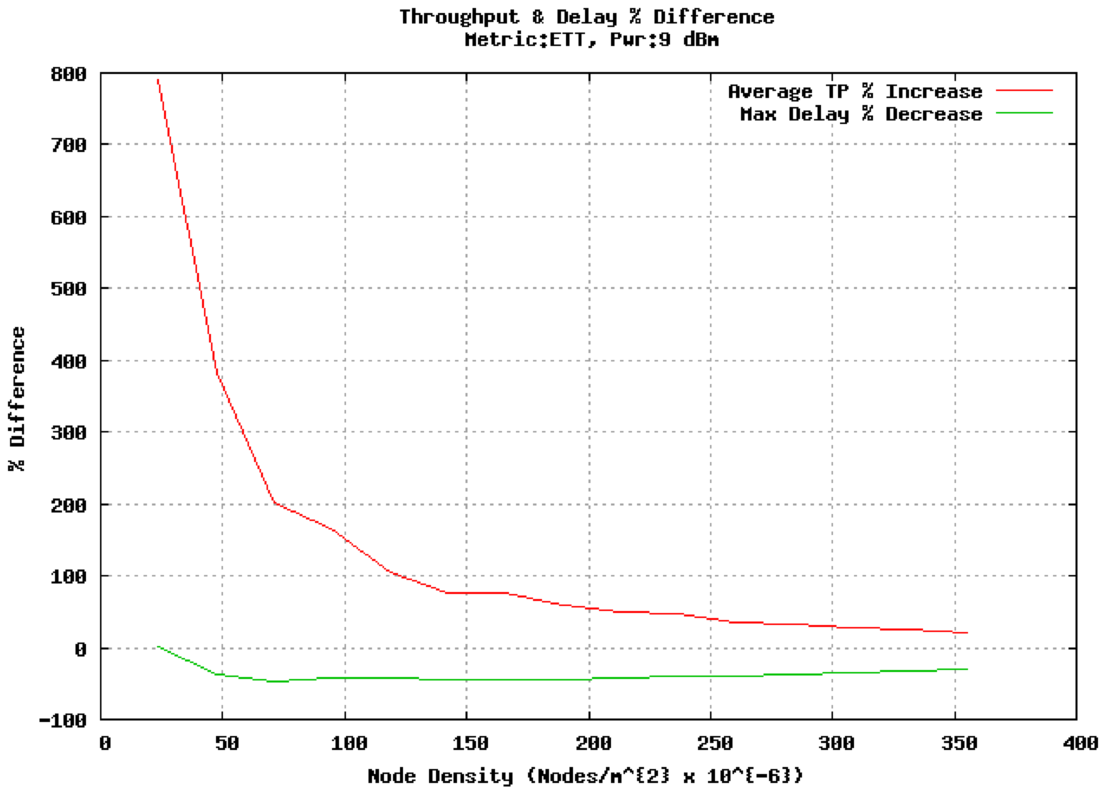


Figure 5-44: Performance gains when using the power control algorithm.

In the graph shown in Figure 5-45 we plot the maximum and minimum throughput values when using all 1 Mbps and 11 Mbps links and a power setting of 9 dBm. If we assume a receiver sensitivity of -95 dBm for interference and a path loss coefficient, $\gamma = 3$ we can calculate the coverage area by manipulating the formula in section 2.9.1. Then, for any given node density we can estimate the number of nodes within range. Then taking either 1 Mbps or 11 Mbps as the link rate and their corresponding efficiency factor as given in [Uni05] we can plot the maximum and minimum achievable throughputs for our simulator.

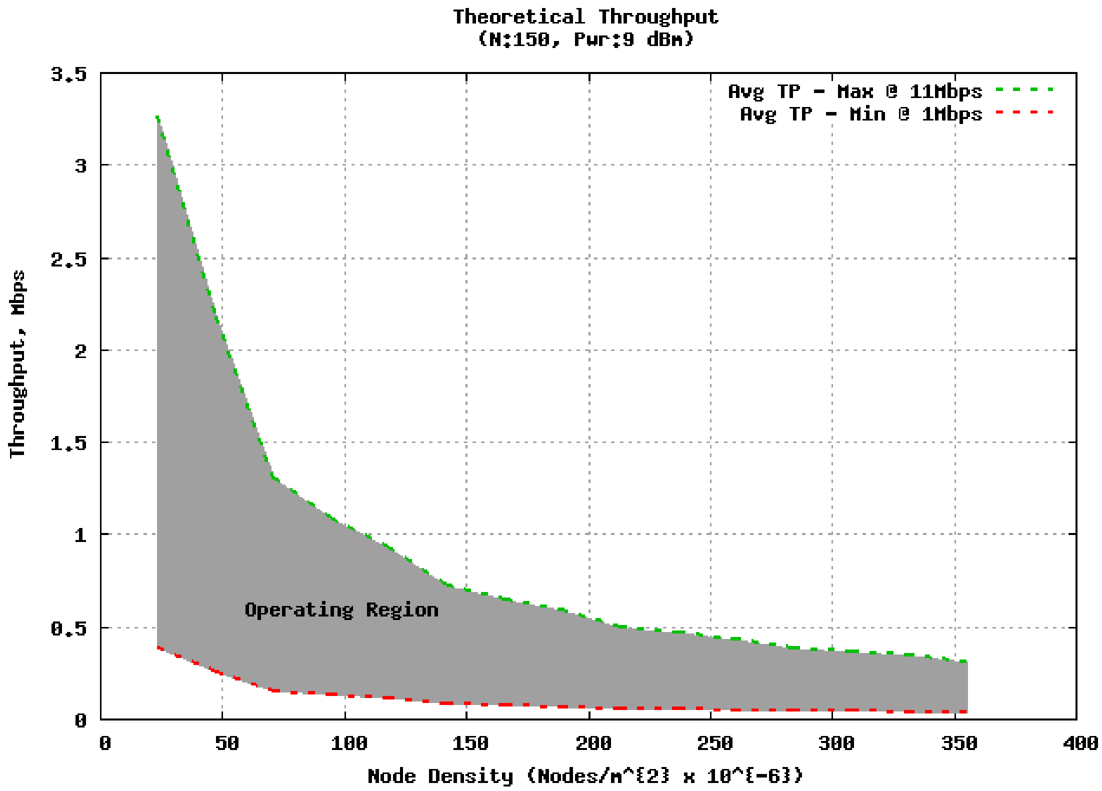


Figure 5-45: Theoretical operating region for power optimisation algorithm.

The grey area in Figure 5-45 represents the region in which our power optimisation algorithm will operate. As we can see by comparing this graph to the throughput results in Figure 5-37 the performance of our network after power adaptation is closely matched. Our optimised network fails to find optimal trees at higher node densities, however it will still provide a performance gain in the region of 25% to 40%.

We have also shown that our power control algorithm will conserve power in the network in order to reduce the overall network contention. Power conservation is normally not an issue when dealing with WMNs however, this algorithm can easily be adapted for use in low power sensor networks.

5.4 Comparison to Fixed Line Rate Network

In the previous sections a performance evaluation of the optimised network (i.e. through relay nodes and power adaptation) compared to the Basic Model was presented. Recall however, that the Basic Model allows for different multicast branch line rates to be selected based on the poorest performing *Child* node without any further optimisation. In this section the case of using a fixed line rate and fixed transmit power throughout the network for all nodes is considered for performance comparison. Two scenarios are considered; a 1 Mbps line rate using a transmit power of 9 dBm and a 1 Mbps line rate using a transmit power of 18 dBm. (Plots for the transmit power ranging from 0 dBm – 18 dBm can be found in Appendix D).

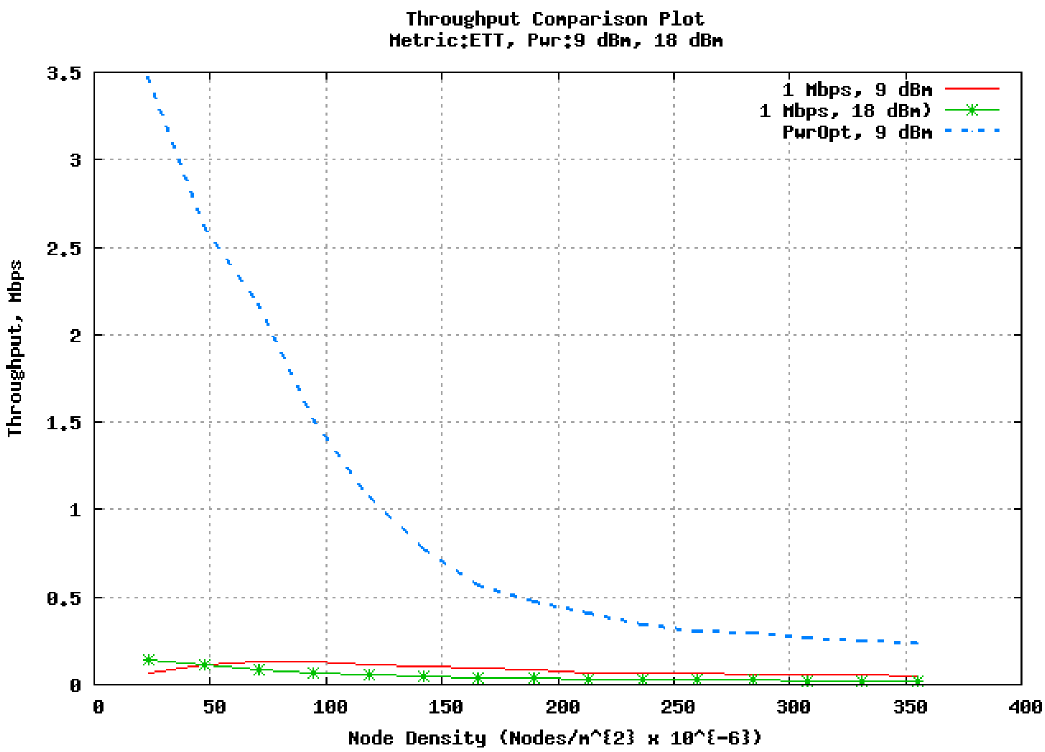


Figure 5-46: Throughput comparison with 1 Mbps line rate.

In Figure 5-46 the throughput performance using the power adaptation algorithm (dashed blue line) is compared to both of the 1 Mbps fixed line rate cases (i.e. with a transmit power of 9 dBm (red line) and 18 dBm (green line)). A significant increase in throughput can be observed when using the power adaptation algorithm at lower node densities. This is due to the algorithm adjusting the transmit power to take full advantage of the higher line rates available and thus increasing the multicast branch rates.

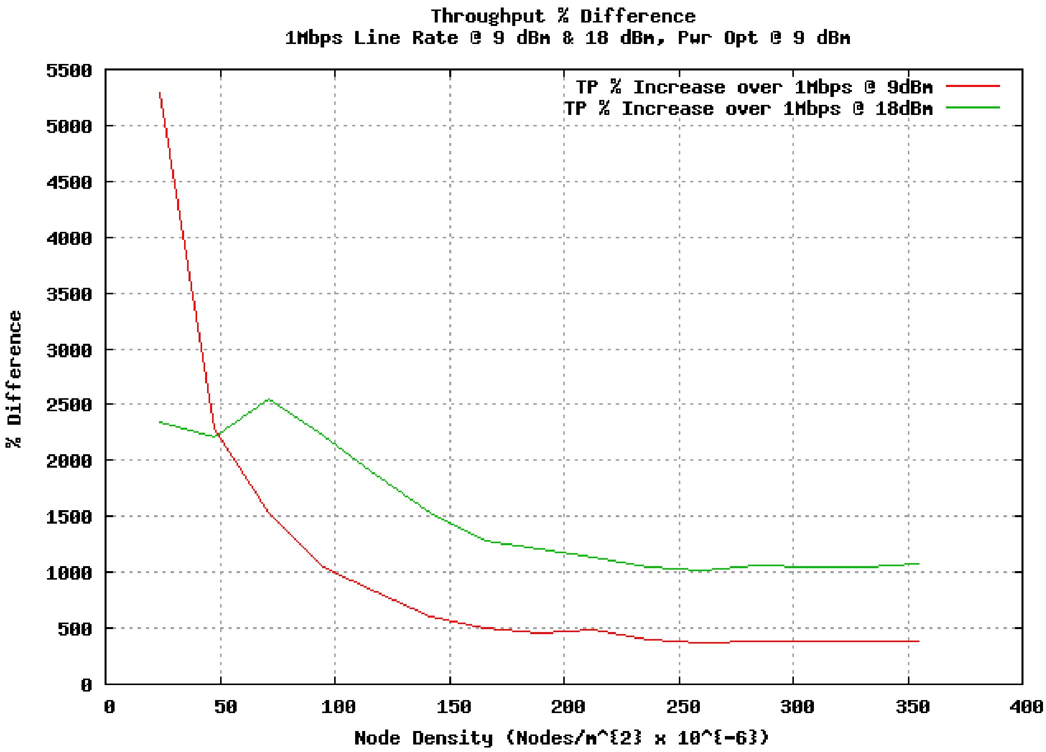


Figure 5-47: Throughput % difference with 1 Mbps line rate.

Figure 5-47 represents a plot of the percentage difference in throughput between the 1 Mbps fixed line rate cases and the power adaptation method as the node density increases. For a node density of 237 x 10^{-6} Nodes/m^2 (i.e. 100 nodes) there is approximately a fourfold increase in throughput over a fixed line rate with a transmit power of 9 dBm and a tenfold increase over a fixed line rate with a transmit power of 18 dBm.

Similarly, the relative delay performance for the same configurations is presented below. Again, the power adaptation algorithm (dashed blue line) outperforms both of the fixed line rate cases as illustrated in Figure 5-48. A significant improvement can be seen over a fixed line rate with a transmit power of 9 dBm. This is due to a lower transmit power requiring more hops (i.e. *Parent* nodes) to reach the destination nodes than would be necessary using a higher transmit power or the power adaptation algorithm. When a fixed line rate is used, the advantage of using the ETT link cost metric is negated.

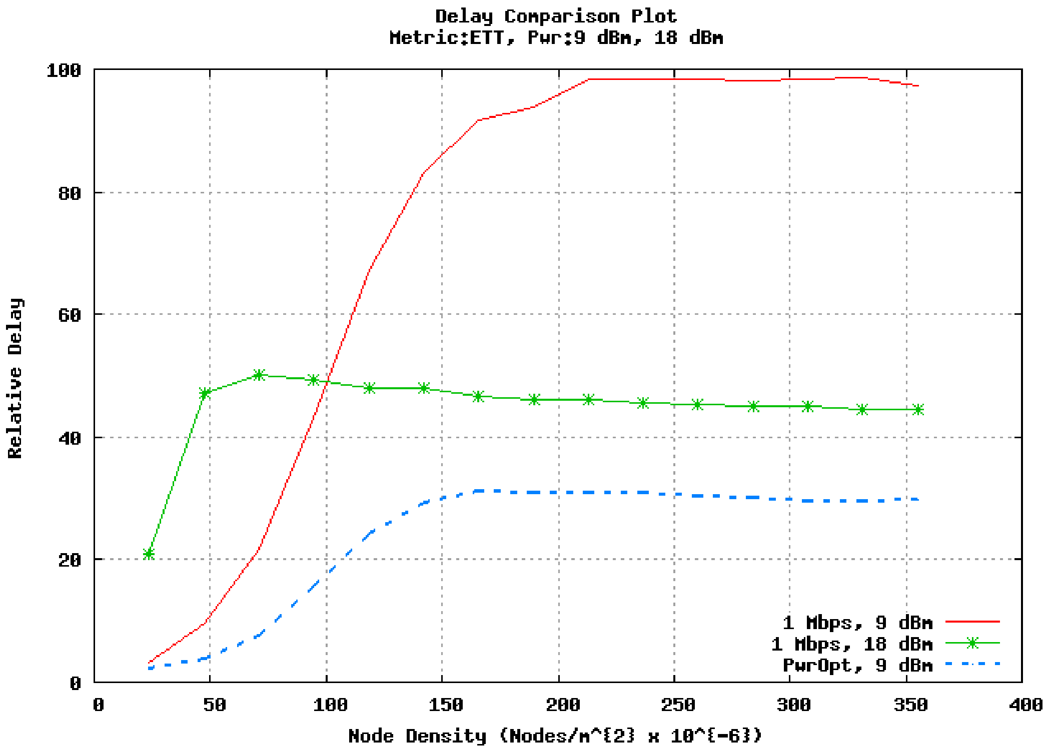


Figure 5-48: Delay comparison with 1 Mbps line rate.

Figure 5-49 represents a plot of the percentage difference in delay between the 1 Mbps fixed line rate cases and the power adaptation method as the node density increases. For a node density of 237 x 10^{-6} Nodes/m^2 (i.e. 100 nodes) there is approximately a 69% decrease in delay over a fixed line rate with a transmit power of 9 dBm and a 33% decrease over a fixed line rate with a transmit power of 18 dBm.

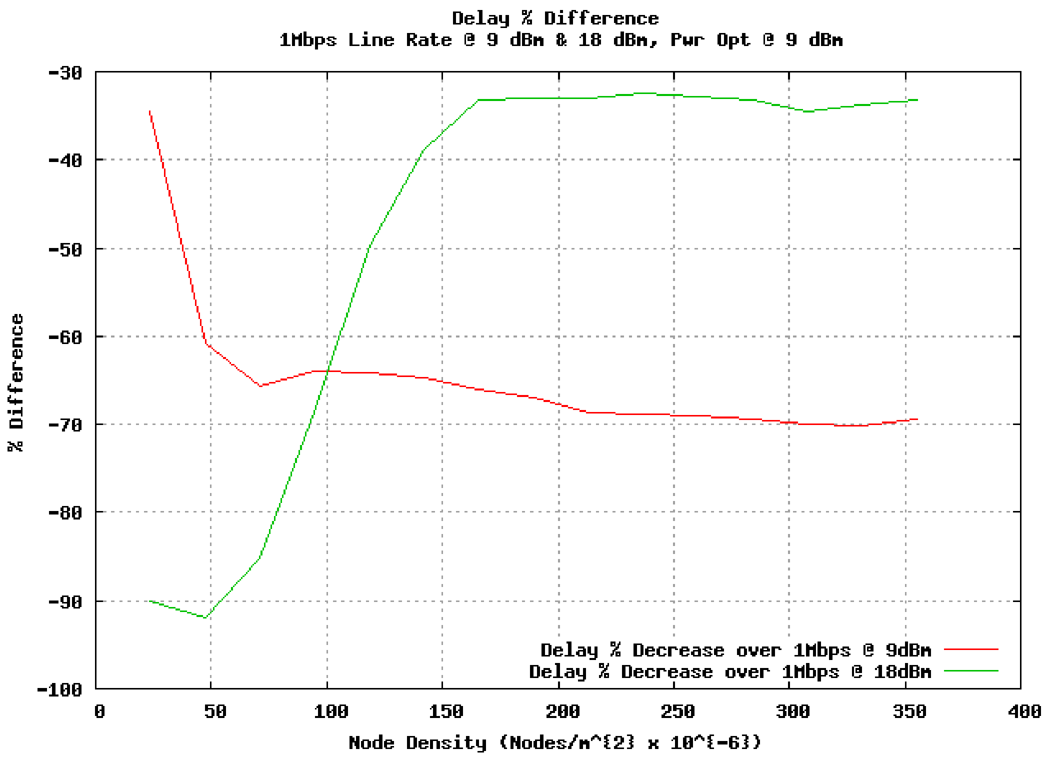


Figure 5-49: Delay % difference with 1 Mbps line rate.

Table 5-6 summarises the difference in throughput (TP) and delay performance when a node density of 237 x 10^{-6} Nodes/m^2 (i.e. 100 nodes) is used. The performance difference for each of the node densities can be found in Appendix D.

Density = 237 x 10^{-6} Nodes/m^2 (*n =100*)	TP, Mbps	Power Opt. % TP Difference	Delay	Power Opt. % Delay Difference
1 Mbps, 9 dBm	0.068	+393 %	98	-69 %
1 Mbps, 18 dBm	0.029	+1043 %	45	-33 %
Power Opt., 9 dBm	0.336		31	

Table 5-6: Comparison of throughput (TP) and delay.

5.5 Practical Implementation - Prototype

In [KBK08a] we have demonstrated that per packet power control can be implemented with a granularity of 0.5 dBm with a latency < 1 ms. We have also demonstrated in [KBK08b] a technique for a conservative transmit power control scheme using the Click Modular Router [KMC00]. In this paper we have demonstrated the relationship between delivery rate and Tx power (as well as RSSI and Tx power). In this paper it was shown that it is possible to decrease the transmit power to maintain an acceptable delivery rate and reduce interference and hence increase throughput.

Our multicast power adaptation algorithm presented in section 4.3 can therefore be adapted to operate in a similar way. In our simulation the algorithm determines if the Tx power should be changed depending on the received power and the receiver sensitivity thresholds. This can be modified to operate based on the delivery rate. The MadWifi [TMP10] driver used in [KBK08a] and [KBK08b] is capable of supporting various bit rate selection algorithms. By using a feedback mechanism to control the Tx rate we can then achieve the following;

- Increase the *Parent* Tx power to increase the delivery rate to the weakest *Child*.

- An increased delivery rate will allow for an increase in the Tx rate.
- Increased Tx rate on a previously weak *Child* will increase the whole multicast rate on a given branch of a multicast tree.

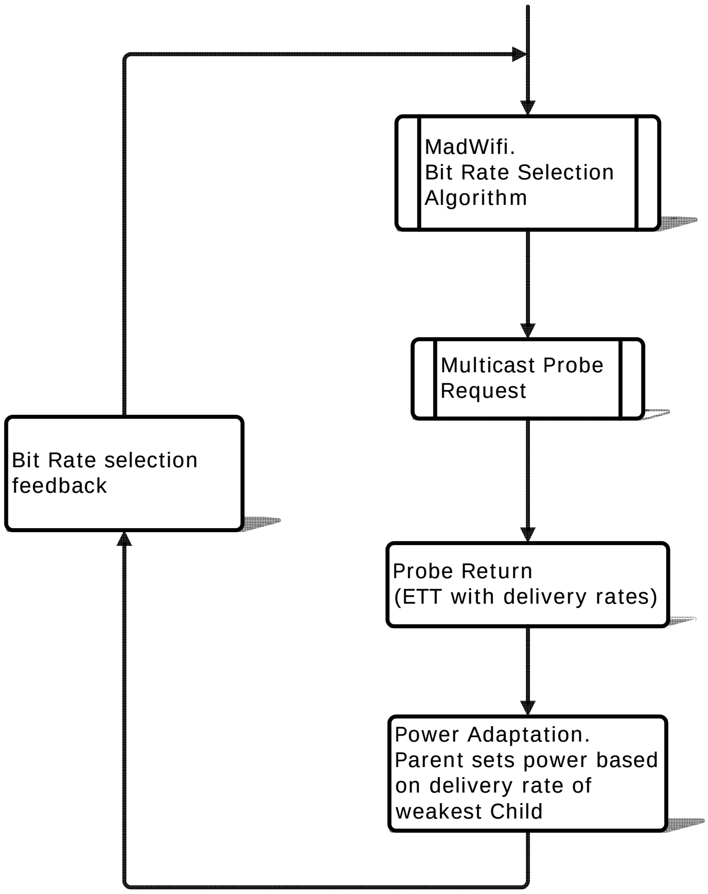


Figure 5-50: Operation of power adaptation algorithm using MadWiFi bit rate selection.

Figure 5-50 above illustrates the basic concept of the implementation. A multicast *probe request* packet is sent to all nodes within the multicast group. The multicast *probe return* contains the delivery rates to each of the nodes in the multicast group. The power adaptation algorithm now operates based on the delivery rate of the weakest *Child*. Feedback (in the form of positive or negative acknowledgement) to the bit rate selection algorithm determines the

Tx rate. The feedback mechanism, sent by all *Child* nodes, allows us to extend existing rate adaptation algorithms (such as Onoe, Amrr [LMT04] or Sample [Bic05] implemented by MadWifi) which are designed for unicast, to now operate in a multicast environment.

Figure 5-50 demonstrates how the power and rate are closely coupled together. This is due to the fact that an increase in the Tx power allows for the use of higher modulations rates (i.e. Tx rates) and therefore more efficient use of the wireless medium. However, at the same time an increase in the Tx power increases interference on the neighbouring branches of the tree. Therefore, power and rate control should not be implemented as separate mechanisms. Only a combined operation can lead to optimal network-wide performance.

5.6 Chapter Summary

In this chapter we present the main findings of our results obtained through simulation. We characterise the network performance using different link cost metrics. This is achieved by using our Basic Model to evaluate and compare the link cost metrics on the performance of mean network throughput, maximum relative delay and node coverage as the transmit power and node density is increased. It was shown that poor performance in throughput and delay occurs when there is a lack of network connectivity. This generally occurs through a combination of the node density being relatively low (below 70 x 10^{-6} Nodes/$m^{2)}$ and when the transmit power is insufficient to ensure reliable connections. It can also be seen that, for results with high throughput and low delay performance, the node density and transmit power are inversely proportional to each other (i.e. high density and low power or vice versa).

Through a comparison of link cost metrics in the Basic Model we found that ETT performs best. Increasing the transmit power will increase the node coverage, however the increased power will also increase the contention for access causing the network performance to degrade. A high network coverage will not guarantee a particular rate; this will be determined by the multicast rate.

By adding relay nodes to the network, using ETT again performs better when compared to *MinHop*. We have shown that by increasing the node density by 14% for ETT a throughput performance gain of 69% can be achieved. We have shown that by increasing the number of nodes further we can provide 11 Mbps multicast rates throughout the entire network. For ETT we can provide a 167% gain for a 90% node density increase. Although the *MinHop* performance gain is higher than ETT the overall network throughput performance is less.

To put these results into perspective Figure 5-51 illustrates the gain in mean network throughput performance for each of our relay node placement methods alongside the predicted gains (red line) for adding relay nodes according to [GuK00] [ABS08].

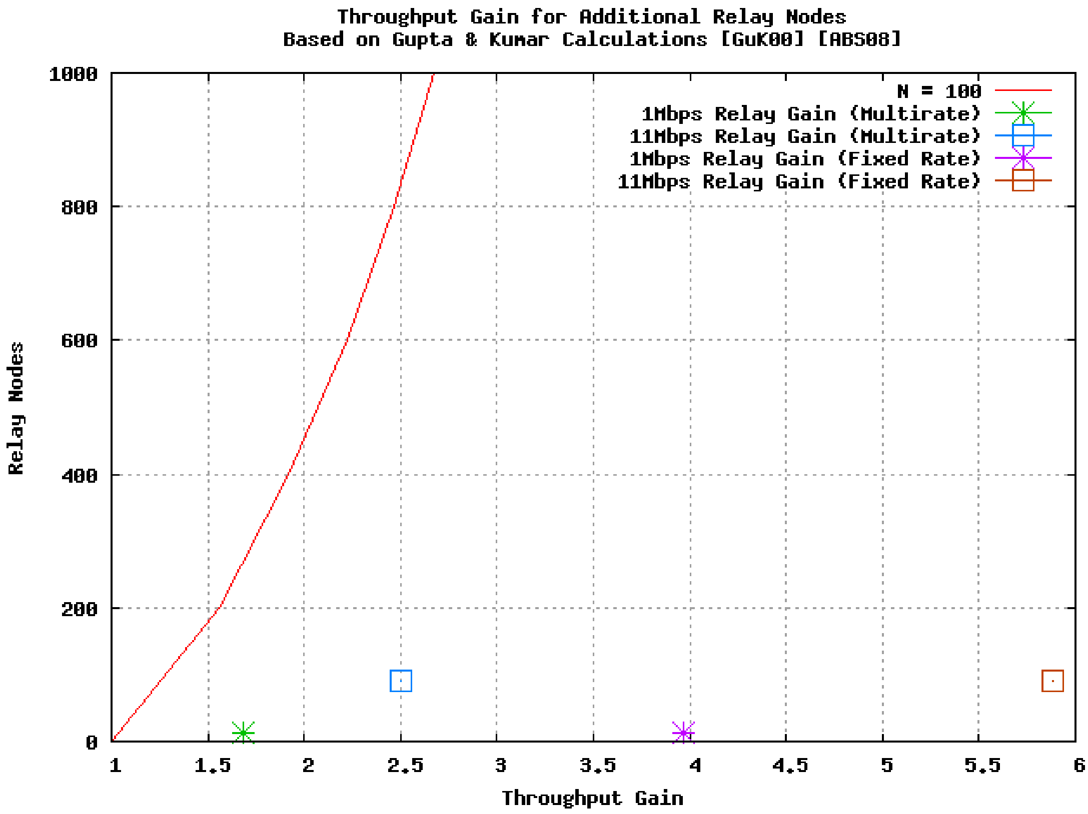


Figure 5-51: Comparison of relay node placement gains. The red line indicates the theoretical values presented by [GuK00] [ABS08].

The plot shows the approximate gains for both of our methods compared to a single fixed rate network and an un-optimised multirate network. The figure clearly shows that by strategically placing relay nodes, the network gain can be improved considerably and will use significantly less relay nodes (less than 100 additional relay nodes for all cases) than that predicted by [GuK00] [ABS08].

We then present results for using our power adaptation algorithm. The algorithm improves the network performance in two ways: by improving the multicast rate (by increasing the *Parent* node transmit power) and by decreasing the *Parent* node transmit power when a higher rate cannot be achieved in order to reduce the path contention. We have shown that mean network throughput gains and a reduction in delay can be achieved across all node densities, with lower densities showing higher gains. Initial starting power should be selected based on the node density. Using our power control algorithm will also improve the node coverage without the overhead of adding additional relay nodes.

To further highlight the performance gain of the power adaptation algorithm we compare it to the performance of a network using a fixed line rate. Two scenarios are presented; a 1 Mbps line rate with using a transmit power of 9 dBm and a 1 Mbps line rate using a transmit power of 18 dBm. In both cases the power adaptation algorithm significantly outperforms these scenarios with regard to the mean network throughput and delay. Figure 5-52 and Figure 5-53 illustrates the PDF and CCDF of the average network throughput calculated for 1000 random topologies for each of the scenarios tested.

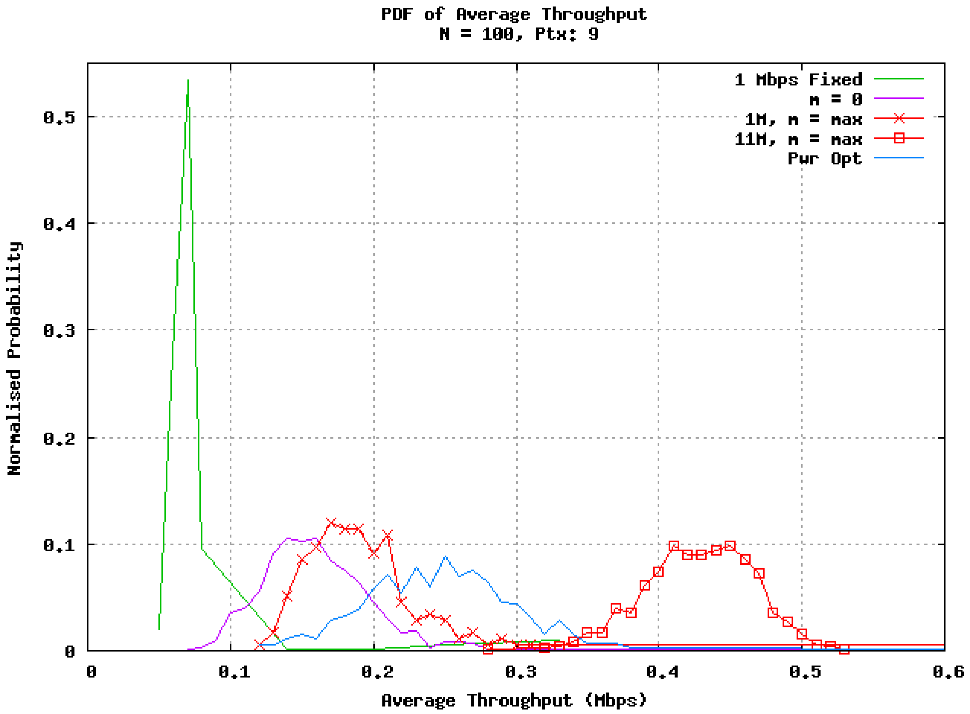


Figure 5-52: Comparison of throughput performance using PDF

Both of these diagrams display graphs for a fixed node density of 237 x 10^{-6} Nodes/m^2 (i.e. 100 nodes) and for a transmit power of 9 dBm for the following simulation results.

- Fixed line rate for all nodes (green line).
- Multirate multicast with no relay nodes and no power optimisation used (purple line).
- Multirate multicast using relay nodes to remove 1 Mbps links (red line, x marker).
- Multirate multicast using relay nodes to guarantee all 11 Mbps links (red line, □ marker).
- Multirate multicast using the power adaptation algorithm (blue line).

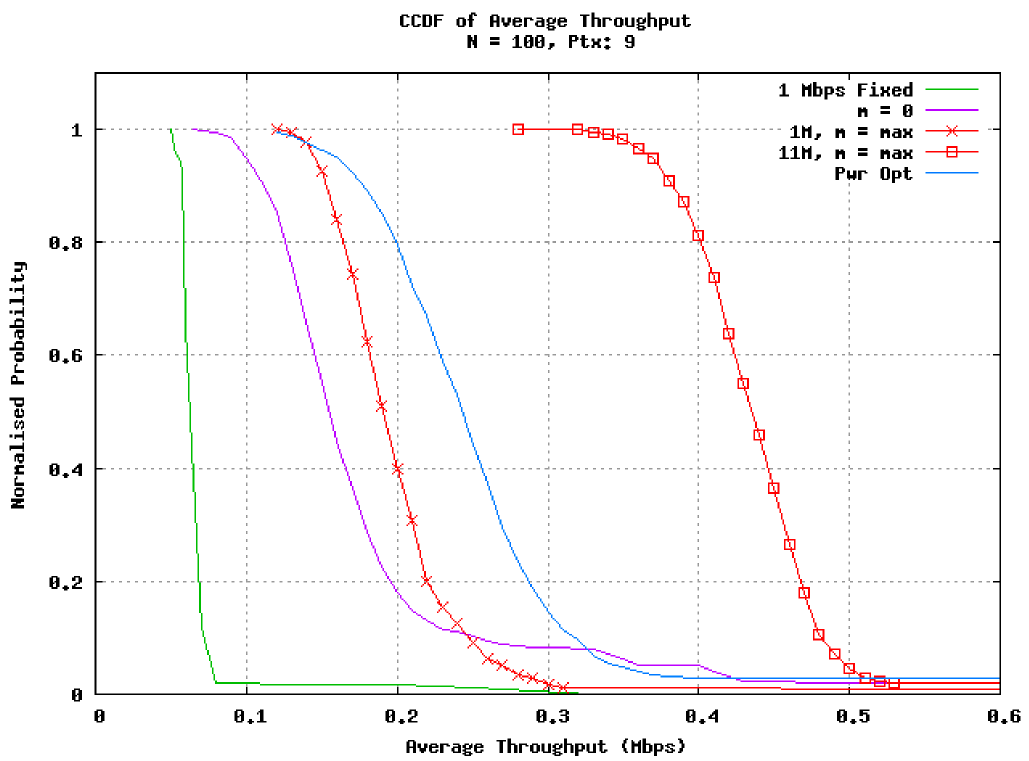


Figure 5-53: Comparison of throughput performance using CCDF

Figure 5-53 shows the level of mean network throughput that can be expected from each of the multicasting methods. For example, if 1 Mbps fixed line rate is used (green), then the maximum network throughput that can be expected is 0.05 Mbps. On the other hand, if an average throughput of at least 0.2 Mbps is required, then the m = 0 case (i.e. a multirate multicast without optimisation, purple) can only deliver this performance for approximately 20% of the topologies. For the case providing relay nodes to remove all 1 Mbps links, m = max (red line, x marker) the same performance can be achieved for 40% of the topologies. The power adaptation algorithm can deliver this performance for approximately 80% of the topologies. As expected, using relay nodes to provide all 11Mbps links, m = max (red line, □ marker) can deliver this performance for 100% of the topologies, but at the cost of a considerable amount of additional nodes.

Finally, we presented a brief outline for a practical implementation of the power control algorithm using the MadWiFi driver. A feedback mechanism, sent by all *Child* nodes, allows us to extend existing rate adaptation algorithms used by the MadWiFi to tune the transmit power of *Parent* nodes.

6 Summary and Conclusions

Multicast is a bandwidth-conserving technology specifically designed to reduce traffic by simultaneously delivering a single stream of information. The most significant benefits of multicasting can be seen in high bandwidth applications such as multimedia transmissions where a single transmission can be used. When employing multicast on wireless networks, the traditional approach of using a single fixed transmission rate for all nodes results in suboptimal performance that limits the capacity and prevents high bandwidth applications from being supported. In this thesis we have proposed two novel approaches for increasing the network throughput in a multirate multicast WMN.

In this thesis we have presented through extensive simulation a comparative study of two multicasting schemes specifically designed for WMNs. We have characterised the operation of multicasting over wireless networks through analysis of a graphical representation of the network topology and through analysing the network performance when using various link cost metrics. We adopted a methodical design approach in the development of a custom simulator which includes all of the detail necessary for conducting wireless multicast multirate simulation. A custom simulation model allows us greater freedom and flexibility in implementing a multicast WMN simulator. By carefully designing a custom simulator it was possible to modify and build upon the simulator in a precise manner (i.e. building in minor modifications and validating each modification through extensive testing) without the need for unnecessary detail. Developing a custom simulator allows us the ability to modify and add new features as required with the flexibility to only include the mechanisms which are of interest to us. This degree of flexibility is not afforded when using commercially available or open source network simulation tools (due to restricted access to modify the necessary modules or lack of support). Furthermore, the use of off-the-shelf simulation tools does not necessarily provide a comparative platform for validation as highlighted in the discussion of simulation tools presented in section 3.4. For these reasons many researchers develop custom simulation tools to develop new protocols (according to [KCC05] over 27% of network simulators are custom builds).

The simulation model used makes basic assumptions regarding the channel model and surrounding environment. The purpose of the simulation is to provide proof of concept. It is likely that the performance gains presented here would be less than those in an experimental hardware test-bed. Network performance is largely dependent on the topology and as such a hardware test-bed would be susceptible to physical limitations of the environment. One of the main reasons for a reduction in performance would be due to assumptions made regarding the channel model. In reality the network would be susceptible to external sources of interference as well propagation losses as discussed in section 2.9. Furthermore, by assuming a circular transmission range the wireless multicast advantage is maximised. In reality the coverage area is not circular (nor is it equal for all radios in the same network) [KNE03] and as such the number of neighbours and hence available paths through the network would be reduced. However, a worst case scenario using the power control algorithm would yield no improvement and a best case which would be less than ideal simulation results given a hostile operating environment.

The performance of various scenarios which included the position of the *Root* node (multicast source) as well as the effects of local clustering was analysed before further development of the Basic Model was carried out. The Basic Model was then modified to allow for enhancements to be made to the network topology. These enhancements are categorised into two groups;

- Simulations using additional relay nodes to improve the multicast branch line rate.
- Simulations using a power adaptation algorithm to improve the multicast branch line rate.

Both techniques aim to improve the mean throughput of the network by allowing higher line rates to be used. Throughout all simulations we found that the ETT link cost metric outperforms all other link cost metrics tested in terms of mean network throughput and delay.

For the method requiring additional relay nodes to be added to the network two approaches were taken. The first approach was to strategically place relay nodes in order to eliminate all 1 Mbps links. The second approach continued to add relay nodes to the network until 11 Mbps links were guaranteed. We show that our method of strategically placing relay nodes in a network can provide significant performance gains in terms of mean network throughput and requires less additional relay nodes than that suggested by [GuK00] [ABS08].

Our next method for enhancing the network throughput performance involved using an algorithm for tuning the transmit power on a per node basis. Each forwarding node (i.e. a *Parent* node) adapts its transmit power in order to increase the multicast branch line rate. If an increase in branch line rate can not be achieved, the wireless node reduces the transmit power in order to decrease interference. We provide statistical results to compare the performance of each of the simulation methods developed a particular *Root* node position. It can be shown that when a node density of 237 x 10^{-6} Nodes/m^2 (i.e. 100 fixed nodes) is used with a transmit power of 9 dBm the power optimisation algorithm can deliver a minimum throughput of at least 0.2 Mbps approximately 80% of the time. The mean network throughput (TP) results compare as follows;

$\mathrm{P}\left[TP > 0.2\,Mbps\right] = 0.2$, using multirate multicast, m = 0.

$\mathrm{P}\left[TP > 0.2\,Mbps\right] = 0.4$, relays on 1 Mbps links, m = max

$\mathrm{P}\left[TP > 0.2\,Mbps\right] = 0.8$ using power adaptation.

From which we can conclude that the power optimisation algorithm is more effective in delivering a network throughput performance improvement without the need for additional hardware.

6.1 Future Work

The results show that there is potential for significant gains to be achieved when using a power optimised multirate multicast network. For this reason we have suggested a method for implementing the power adaptation algorithm using the MadWiFi driver to implement rate control by tuning the transmit power based on a feedback mechanism. Future work should provide an implementation of a hardware test-bed using this method whereby each forwarding node would be responsible for tuning its own transmit power based on its immediate neighbours. Furthermore to ensure repeatability and reliability of testing, the hardware test-bed should aim to minimise external sources of interference. This can be achieved using an RF screened room. However, such a solution does not work well with WMNs due to the restricted size of such rooms. This is a common problem with large scale wireless mesh network experiments and hence the reason why simulation is often used [BBE99]. An alternative solution to this problem would be to connect each of the wireless nodes radio equipment via RF cabling. Line attenuation can be controlled using attenuated couplers to emulate specific network conditions. Furthermore, modifying commonly used network simulators such as NS-2 is worth further investigation for a comparative evaluation. However, such modifications may not be possible or would at least prove to be a non-trivial task. NS-2 is a packet based simulator and would require extensive modification in order to yield comparable results. The methodology employed throughout the design of the simulator meant that each stage was self validating. For the purpose of publication it is acknowledged that there may be a requirement for further validation. Future work would provide validation through the use of a hardware test-bed.

The network simulator can be improved by using a traffic flow model and a more sophisticated channel model in order to provide a more detailed simulation environment. Moreover, the simulation assumes a pure multicast environment without any other network traffic. Future work should consider a mixed traffic environment which would include unicast traffic. However, the main objective was to show that single rate multicast networks are sub-optimal and thus it was felt that an increase in such detail was unnecessary to prove this. It is worth noting that the simulator can easily be adapted to support 801.11a/g line rates. For ease of simulation and to reduce the simulation time the four line rates available under 802.11b were used. The simulation model itself underwent significant profiling analysis in order to improve the efficiency of the code. For further development of the simulator it would be worth porting the code from its current interpreted language, Perl to a compiled language such as C or C++. Using a compiled language would help to improve the execution times. The average execution time for 1000 topologies, over 15 node densities and 19 transmit powers takes approximately 72 hours.

There still remain many open issues regarding multicasting over WMNs. For example, reliable service, efficient membership updates, multi-radio multi-channel networks, and quality of service guarantees [KLS07] are amongst those not covered in our discussion from section 3. These issues and the current lack of support present an ideal opportunity for researchers to develop new techniques without the constraints of standardised guidelines. Furthermore, with the recent advances in network coding a cross discipline design would be possible by using our power adaptation algorithm alongside such network coding schemes. Furthermore, our research considers a single radio solution only. With the emergence of 802.11n and dual radio mesh nodes an even more sophisticated solution would be possible. For instance, by using a multi-channel multi-radio wireless mesh network it would be possible to schedule separate transmissions on separate channels to the multicast group. During multicast sessions, each forwarding node can determine the *Child* node with the worst link in the multicast branch and dedicate a radio and channel to this *Child*. This would then allow the remainder of the multicast branch nodes to fully exploit the multicast advantage without being impeded by the poorest performing node (known as the “cry-baby” scenario).

It is also worth noting that our power adaptation technique, although not designed for the purpose, can easily be adapted to suit sensor networks for energy conservation. Rather than having the network throughput as the optimising objective the transmit power, node coverage or any other relevant performance metric can easily be substituted in its place. Other, more novel approaches, use mobile robotics [MoR10] equipped with mesh nodes and GPS to dynamically transform the network topology [Mil09].

The work presented in this thesis has been separated into two distinct techniques for improving multicast communications, i.e. adding relay nodes and power control adaptation. Two journal papers have been written and submitted to IEEE journals for publication review.

6.2 Conclusions

The main findings from the simulations carried out can be summarised as follows:

- A mean network throughput performance increase of 4-10 times over the single fixed line rate scenario is achieved when using the power adaptation algorithm.
- Significant decrease in delay (33% - 69%) when compared to the single line rate scenario using the power adaptation algorithm.

- The power adaptation algorithm shows a significant improvement in throughput performance when the node density is low.
- The power adaptation algorithm can improve the throughput performance at higher node densities by decreasing the transmit power.
- The performance, in terms of the network throughput and delay, is largely dependent on the network topology, density and transmit power.
- The power adaptation algorithm improves the node coverage by extending the range and increasing the rate to *Leaf* nodes.
- The throughput performance of the power adaptation algorithm is comparable to the throughput performance of the network using relay nodes to guarantee 11 Mbps line rates.
- The power adaptation algorithm does not require the additional hardware resources required using relay nodes.
- The number of relay nodes required to provide a 2.5 times throughput gain (typically less than 100 relay nodes) is significantly less than that suggested by [ABS08] and [GuK08] (greater than 800 relay nodes).

The use of a single rate in multicast WMNs can be shown to be suboptimal. The use of strategically placed relay nodes in the network can provide throughput performance gains. However, the associated cost of equipment and additional hardware deployment makes this solution impractical. By tuning the power in a multirate multicast WMN the throughput can be increased significantly without any additional capital costs.

7 Bibliography.

[ABP04] Atul Adya, Paramvir Bahl, Jitendra Padhye, et al. 2004. A Multi-Radio Unification Protocol for IEEE 802.11Wireless Networks. Proceedings of the First International Conference on Broadband Networks (BROADNETS'04).

[ABS09] Eiman Alotaibi, Prantik Bhattacharyya, and Huan Song. 2009. Advanced Topics in Wireless Communication Networks (ECS 289I). [Online]. Available: http://dsl.cs.ucdavis.edu/~prantik/research.php.

[ACL00] Rudolf Ahlswede, Ning Cai, Shuo-Yen Robert Li et al. 2000. Network Information Flow. IEEE Transaction on Information Theory, Volume 46, No. 4, July 2000.

[AkW05] Ian F. Akyildiz and Xudong Wang. 2005. A Survey on Wireless Mesh Networks. IEEE Radio Communications, September 2005.

[AnY06] T.R. Andel and A. Yasinsac. 2006. On the Credibility of Manet Simulations. IEEE Computer Society, Volume 39, Issue 7.

[ANS05] A. Adams, J. Nicholas, and W. Siadak. 2005. Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised). Network Working Group Request for Comments: 3973.

[BAB05] John Bicket, Daniel Aguayo, Sanjit Biswas, et al. 2005. Architecture and Evaluation of an Unplanned 802.11b Mesh Network. ACM, MobiCom'05, August 28–September 2, 2005, Cologne, Germany.

[BaC05] A. Bahri and S. Chamberland. 2005. Designing WLAN with Minimum Bandwidth Guarantees. Proc. of the IEEE International Conference on Wireless And Mobile Computing, Networking And Communications, WiMob'2005, pp. 86 - 89.

[BaM02] Suman Banerjee and Archan Misra. 2002. Minimum Energy Paths for Reliable Communication in Multi-hop Wireless Networks. ACM, MOBIHOC'02, June 9-11, 2002, EPFL Lausanne, Switzerland.

[BBE99] Sandeep Bajaj, Lee Breslau, Deborah Estrin et al. 1999. Improving Simulation for Network Research. USC Computer Science Department Technical Report 99-702b.

[BEF00] Lee Breslau, Deborah Estrin, Kevin Fall et al. 2000. Advances in Network Simulation. IEEE Computer, Volume 33, No.5, pp. 59-67, May, 2000.

[BeG02] Dimitri Bertsekas and Robert Gallager. 2002. Data Networks, Second Edition. Prentice-Hall, India.

[BeW98] Bell, Witten, and Fellows. 1998. Ice roads - Steiner trees. Computer Science Unplugged. [Online]. Available: http://csunplugged.com/.

[BFC93] Tony Ballardie, Paul Francis and Jon Crowcroft. 1993. Core Based Trees (CBT) ACM, SIGCOMM Computer Communication, Volume 23, Issue 4, Pages: 85 - 95

[Bic05] John C. Bicket. 2005. Bit-rate Selection in Wireless Networks. Master Thesis, MIT. [Online]. Available: http://pdos.csail.mit.edu/papers/jbicket-ms.pdf.

[BMT98] Rajive Bagrodia, Richard Meyer, Mineo Takai, et al. 1998. IEEE Computer Society Press, Volume 31, Issue 10 (October 1998), Pages: 77 – 85.

[Bra00] Arnold W. Bragg. 2000. Which Network Design Tool Is Right for You? IEEE Educational Activities Department, IT Professional, Volume 2, Issue 5 (September 2000), Pages: 23 – 31.

[BRC05] O.C. Branquinho, N. Reggiani, C.E. Correa, et al. 2005. WLAN 802.11 MAC anomaly mitigation using SNR to Control Backoff Contention Window. SBMO/IEEE MTT-S International Conference on Microwave and Optoelectronics, 2005.

[BrD07] Michele Bracuto, Gabriele D'Angelo. 2007. Detailed Simulation of Large-Scale Wireless Networks. 11th IEEE Symposium on Distributed Simulation and Real-Time Applications.

[BTA99] Lokesh Bajaj, Mineo Takai, Rajat Ahuja, et al. 1999. Simulation of Large-scale Heterogeneous Communication Systems. In proceedings of MILCOM, November 1999.

[BWG99] Vanu Bose, David Wether, andl John Guttag. 1999. Next Century Challenges: RadioActive Networks. Mobicom '99 Seattle.

[CaK08] Joseph D. Camp and Edward W. Knightly. 2008. The IEEE 802.11s Extended Service Set Mesh Networking Standard. IEEE Communications Magazine Volume 46, Issue 8, page(s) 120 – 126.

[ChM05] Chun Tung Chou and Archan Misra. 2005. Low Latency Multimedia Broadcast in Multi-Rate Wireless. WiMesh-2005, First

IEEE Workshop on Wireless Mesh Networks, Held in conjunction with SECON-2005 Santa Clara, CA 26th September, 2005.

[ChS06] Kim Bong Chang and Lee Hwang Soo. 2006. Performance Comparison of Route Metrics for Wireless Mesh Networks. IEICE Trans Commun (Inst Electron Inf Commun Eng), Volume E89-B, No.11, Page.3124-3127 (2006).

[CJB03] P. Chevillat, J. Jelitto, A. Noll Barreto, et al. 2003. A Dynamic Link Adaptation Algorithm for IEEE 802.11a Wireless LANs. IEEE International Conference on Communications. ICC '03. Volume 2, page(s): 1141 - 1145 vol.2.

[ClJ03] T. Clausen and P. Jacquet. 2003. Optimized Link State Routing Protocol (OLSR). Network Working Group Request for Comments: 3626.

[CLR02] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2002. Introduction to Algorithms, Second Edition. The MIT Press, McGraw-Hill Book Company.

[CMQ06] Chun Tung Chou, A. Misra, and J. Qadir. 2006. Low-Latency Broadcast in Multirate Wireless Mesh Networks. IEEE Journal on Selected Areas in Communications, Volume: 24 , Issue: 11.

[CNO99] James H. Cowie, David M. Nicol, and Andy T. Ogielski. 1999. Modeling The Global Internet. IEEE Educational Activities Department, Computing in Science and Engineering, Volume 1, Issue 1 (January 1999), Pages: 42-5.

[CNP07] Jaehyuk Choi, Jongkeun Na, Kihong Park, et al. 2007. Adaptive Optimization of Rate Adaptation Algorithms in Multi-Rate

WLANs. IEEE International Conference on Network Protocols, 2007. ICNP 2007.

[Cop04] Ben Coppin. 2004. Artificial Intelligence Illuminated. Jones and Bartlett Publishers, Inc.

[CSS02] David Cavin, Yoav Sasson, and Andr´e Schiper. 2002. On the Accuracy of MANET Simulators. POMC'02, October 30-31, 2002, Toulouse, France.

[CSU05] Computer Science Unplugged. 2005. The Muddy City - Minimal Spanning Trees. Computer Science Unplugged. [Online]. Available: http://csunplugged.com/.

[CWP07] Cisco Press. 2007. Cisco White Paper IP Multicast Technical Overview. [Online]. Available: http://www.cisco.com/

[DAB03] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, et al. 2003. A HighThroughput Path Metric for MultiHop Wireless Routing. MobiCom '03, September 14–19, 2003, San Diego, CA.

[DBA04] Qunfeng Dong, Suman Banerjee, Micah Adler, et al. 2004. Minimum Energy Reliable Paths Using Unreliable Wireless Links. MobiHoc'05, May 25–27, 2005, Urbana-Champaign, Illinois.

[Dee89] S. Deering. 1989. Host Extensions for IP Multicasting. Network Working Group Request for Comments: 1112.

[DPZ04a] Richard Draves, Jitendra Padhye, and Brian Zill. 2004. Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks. MobiCom'04, Sept. 26-Oct. 1, 2004, Philadelphia.

[DPZ04b] Richard Draves, Jitendra Padhye, and Brian Zill. 2004. Comparison of Routing Metrics for Static Multi-Hop Wireless

Networks. SIGCOMM'04, Aug. 30–Sept. 3, 2004, Portland, Oregon.

[DSY90] Alexander Dupuy, Jed Schwartz and Yechiam Yemini. 1990. NEST - A Network Simulation and Prototyping Testbed. Communications of the ACM Volume 33, Issue 10 (October 1990). Special Issue on Simulation, Pages: 63 – 74.

[EFH98] D. Estrin, D. Farinacci, A. Helmy et al. 1998. Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification. Network Working Group Request for Comments: 2362.

[Eri94] Hans Eriksson. 1994. MBONE: The Multicast Backbone. Communications of the ACM. Volume 37, Issue 8.

[FlJ95] Sally Floyd and Van Jacobson. 1995. Link-Sharing and Resource Management Models for Packet Networks. IEEE/ACM Transaction On Networking, Vol. 3, No. 4. August 1995.

[Gas02] Matthew Gast. 2002. 802.11 Wireless Networks: The Definitive Guide. O'Reilly.

[GaS99] J.J. Garcia-Luna-Aceves and Marcelo Spohn. 1999. Source-Tree Routing in Wireless Networks. Seventh International Conference on Network Protocols, ICNP '99. Proceedings page(s): 273 – 282.

[GiK05] A.M. Gibney and M. Klepal. 2005. Tool for automatic design and Optimisation of Large Scale WLAN. Proc. of the 7th IFIP International Conference on Mobile and Wireless Communications Networks, MWCN'05.

[GMR08] Cristiana Gomes, Christelle Molle, and Patricio Reyes. 2008. Optimal Design of Wireless Mesh Networks. 9emes Journees Doctorales en Informatique et Reseaux, JDIR 2008, Villeneuve d'Ascq : France.

[GMS10] GloMoSim. 2010. The Global Mobile Information Systems Simulator. [Online]. Available: http://pcl.cs.ucla.edu/projects/glomosim/.

[Gpt10] Gnuplot. 2010. [Online]. Available: http://www.gnuplot.info/.

[GrH85] R. L. Graham and Pavol Hell. 1985. On the History of the Minimum Spanning Tree Problem. Annals of the History of Computing, Volume 7, Number 1, January 1985, Page 2.

[GrT02] Matthias Grossglauser and David N. C. Tse. 2002. Mobility Increases the Capacity of Ad Hoc Wireless Networks. IEEE/ACM Transactions on Networking, Vol. 10, No. 4, August 2002.

[GuK00] Piyush Gupta and P. R. Kumar. 2000. The Capacity of Wireless Networks. IEEE Transactions on Information Theory, Vol. 46, No. 2, March 2000.

[GuM03] Chao Gui and Prasant Mohapatra. 2003. Efficient Overlay Multicast for Mobile Ad Hoc Networks. Wireless Communications and Networking, WCNC March 2003, Volume 2, page(s) 1118 – 1123.

[GuM04] Chao Gui and Prasant Mohapatra. 2004. Scalable Multicasting in Mobile Ad Hoc Networks. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 7-11 March 2004, Volume 3, page(s): 2119 – 2129.

[HBE01] John Heidemann, Nirupama Bulusu, Jeremy Elson et al. 2001. Effects of Detail in Wireless Network Simulation. In Proceedings of the SCS Multiconference on Distributed Simulation, pp. 3-11.

[HBT06] Anwar Al Hamra, Chadi Barakat and Thierry Turletti. 2005. Network Coding for Wireless Mesh Networks: A Case Study. International Workshop on Wireless Mobile Multimedia, Proceedings of the 2006 International Symposium on World of Wireless, Mobile and Multimedia Networks, Pages: 103 - 114.

[HCG08] Elyes Ben Hamida, Guillaume Chelius, and Jean-Marie Gorce. 2008 On the Complexity of an Accurate and Precise Performance Evaluation of Wireless NetworksUsing Simulations. MSWiM'08, October 27–31, 2008.

[Hed08] Jerry d. Hedden. 2008. Math-Random-MT-Auto-6.14. [Online]. Available: http://search.cpan.org/~jdhedden/Math-Random-MT-Auto-6.14/.

[HRB03] M. Heusse, F. Rousseau, G. Berger-Sabbatel, et al .2003. Performance anomaly of 802.11b. IEEE Societies Twenty-Second Annual Joint Conference of the IEEE Computer and Communications, INFOCOM 2003, Volume 2, Page(s) 836 - 843.

[HVB01] Gavin Holland, Nitin Vaidya, and Paramvir Bahl. 2001. A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks. ACM SigMobile 7/2001 Rome, Italy.

[IEE06] IEEE. Draft Amendment to Standard for Information Technology - Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Part 11: Wireless

Medium Access Control (MAC) and physical layer (PHY) specifications: Amendment: ESS Mesh Networking. IEEE Std p802.11s/D0.2. 2006.

[IEE07] IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE IEEE Std 802.11 - 2007 (Revision of IEEE Std 802.11-1999)

[IEE09] Draft STANDARD for Information Technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 10: Mesh Networking. IEEE P802.11s/D3.0, Mar 2009.

[IET09] IEFT. (2009, April 22). Mobile Ad-hoc Networks (manet). [Online]. Available: http://www.ietf.org/.

[INR09] INRIA Activity Report. Team D-NET, Dynamic Networks. 2009. [Online]. Available: http://ralyx.inria.fr/2009/Raweb/d-net/d-net.pdf.

[JeJ01] Jorjeta G. Jetcheva David B. Jonson. 2001. Adaptive Demand-Driven Multicast Routing in Multi-Hop Wireless Ad Hoc Networks. MobiHOC 2001, Long Beach, CA.

[JiC01] Lusheng Ji and M. Scott Corson. 2001. Differential Destination Multicast–A MANET Multicast Routing Protocol for Small Groups.

Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, Volume: 2, INFCOM.2001, Page(s): 1192 – 1201.

[JiS10] JiST / SWANS. 2010. Java in Simulation Time / Scalable Wireless Ad hoc Network Simulator. [Online]. Available: http://jist.ece.cornell.edu/.

[JoM96] David B. Johnson and David A. Maltz. 1996. Dynamic Source Routing in Ad Hoc Wireless Networks. Mobile Computing, edited by Tomasz Imielinski and Hank Korth. Kluwer Academic Publishers.

[JuS03] Jangeun Jun and Mihail L. Sichitiu. 2003. The Nominal Capacity of Wireless Mesh Networks. IEEE Wireless Communications, October 2003.

[KaM97] A. Kamerman and L. Monteban, 1997. WaveLAN-II: A High-Performance Wireless LAN for the Unlicensed Band. Bell Labs Technical Journal, Vol. 2, 1997.

[Kar75] Richard M. Karp. 1975. Reducibility Among Combinatorial Problems. The Journal of Symbolic Logic, Vol. 40, No. 4 (Dec., 1975), pp.618-619.

[KaU02] M. Kamenetsky and M. Unbehaun. 2002. Coverage Planning for Outdoor Wireless LAN Systems. Proc. of the International Zurich Seminar on Broadband Communications, 2002. pp. 49-1 - 49-6.

[KBK08a] Karol Kowalik, Marek Bykowski, Brian Keegan, et al. 2008. An Evaluation of a Conservative Transmit Power Control Mechanism on an Indoor 802.11 Wireless Mesh Testbed. International

Conference on Wireless Information Networks and Systems (WINSYS'08), Porto, Portugal, July 26 - 29, 2008.

[KBK08b] Karol Kowalik, Marek Bykowski, Brian Keegan, et al. 2008. Practical Issues of Power Control in IEEE 802.11 Wireless Devices. IEEE International Conference on Telecommunications (ICT 2008), St. Petersburg, June 2008.

[KCC05] Stuart Kurkowski, Tracy Camp, and Michael Colagrosso. 2005. MANET Simulation Studies: The Incredibles. SIGMOBILE Mob. Comput. Commun. Rev., Vol. 9, No. 4, pp. 50-61.

[Kes91] Srinivasan Keshav. 1991. A Control-Theoretic Approach to Flow Control. ACM SIGCOMM Computer Communication Review, Volume 21, Issue 4 (September 1991), Pages: 3 – 15.

[KiH04] Byung-Soon Kim and Ki-Jun Han. 2004. Multicast handoff agent mechanism for all-IP mobile network. Mobile Networks and Applications. Volume 9, Issue 3.

[KKD07a] Brian Keegan, Karol Kowalik and Mark Davis. 2007. Experimental Measurement of Overhead Associated with Active Probing of Wireless Mesh Networks. IEEE International Conference on Signal Processing and Communications (ICSPC 2007), Dubai.

[KKD07b] Karol Kowalik, Brian Keegan, and Mark Davis. 2007. RARE - Resource Aware Routing for mEsh Proceedings of IEEE International Conference on Communications, ICC 2007.

[KLS07] Zongwu Ke, Layuan Li, Qiang Sun et al. 2007. A QoS Multicast Routing Algorithm for Wireless Mesh Networks. *Eighth ACIS*

International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing.

[KMC00] Eddie Kohler, Robert Morris, Benjie Chen, John Jannotti, and M. Frans Kaashoek. 2000. The Click modular router. ACM Transactions on Computer Systems, 4(18), November 2000.

[KNE03] David Kotz, Calvin Newport, and Chip Elliott. 2003. The mistaken Axioms of Wireless-Network Research. Dartmouth College Computer Science Technical Report TR2003-467.

[KNG04] David Kotz, Calvin Newport, Robert S. Gray, et al. 2004. Experimental Evaluation of Wireless Simulation Assumptions. ACM, MSWiM October 2004, Venezia.

[KoB06] Can Emre Koksal and Hari Balakrishnan. 2006. Quality-Aware Routing Metrics for Time-Varying Wireless Mesh Networks. IEEE Journal on Selected Areas in Communications, Vol. 24, No. 11, November 2006.

[KRH08] Sachin Katti, Hariharan Rahul and Wenjun Hu. 2008. XORs in the Air: Practical Wireless Network Coding. IEEE/ACM Transactions on Networking, Volume. 16, NO. 3, June 2008.

[KUK05] S. Kouhbor, J. Ugon, A. Kruger, et al. 2005. Branch, Optimal Placement of Access point in WLAN Based on a New Algorithm. Proc. of the International Conference on Mobile Business, ICMB’05, pp. 592 – 598.

[KyV05] Pradeep Kyasanur and Nitin H. Vaidya. 2005. Routing and Interface Assignment in Multi-Channel Multi-Interface Wireless

Networks. Wireless Communications and Networking Conference.. IEEE, Vol. 4 (2005), pp. 2051-2056 Vol. 4.

[KyV06] Pradeep Kyasanura and Nitin H. Vaidya. 2006. Routing and Link-layer Protocols for Multi-Channel Multi-Interface Ad Hoc Wireless Networks. SIGMOBILE Mob. Comput. Commun. Rev., Vol. 10, No. 1. (January 2006), pp. 31-43.

[LiJ07] Jeff Liebermann, 2007, WiFi Receiver Sensitivity. [Online]. Available: http://802.11junk.com/jeffl/rx-sens/receiver%20sensitivity.htm.

[LEH03] Jun Luo, Patrick Th. Eugster, and Jean-Pierre Hubaux. 2003. Route Driven Gossip: Probabilistic Reliable Multicast in Ad Hoc Networks. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies INFOCOM 2003, Volume 3, page(s): 2229 - 2239.

[LFJ03] Gilberto Flores Lucio, Marcos Paredes-Farrera, Emmanuel Jammeh, et al. 2003. OPNET Modeler and Ns-2: Comparing the Accuracy Of Network Simulators for Packet-Level Analysis using a Network Testbed. WSEAS Transactions on Computers 2, 3 (July 2003), 700—707.

[LGC99] Sung-Ju Lee, Mario Gerla, and Ching-Chuan Chiang. 1999. On-Demand Multicast Routing Protocol. Wireless Communications and Networking Conference, 1999. WCNC.

[LGD07] GD Library. 2007. [Online]. Available: http://www.libgd.org/Main_Page.

[LHZ08] Hui Liu, Wei Huang, Xu Zhou, et al. 2008. A Comprehensive Comparison of Routing Metrics for Wireless Mesh Networks. IEEE International Conference on Networking, Sensing and Control, ICNSC 2008.

[LiX09] Xiang-Yang Li. 2009. Multicast Capacity of Wireless Ad Hoc Networks. IEEE/ACM Transactions on Networking, Vol. 17, No. 3, June 2009.

[LKC02] Y. Lee, K. Kim, and Y. Choi. 2002. Optimization of AP Placement and Channel Assignment in Wireless LANs. Proc. of IEEE Conference on Local Computer Networks (LCN'02), 2002, pp. 831 - 836.

[LMT04] Mathieu Lacage, Mohammad Hossein Manshaei, and Thierry Turletti. 2004. IEEE 802.11 Rate Adaptation: A Practical Approach. MSWiM '04: Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems. New York, NY, USA: ACM, 2004, pp. 126-134.

[LSG99] S.-J. Lee, W. Su and M. Gerla, 1999. On-Demand Multicast Routing Protocol for ad-hoc networks, Internet Draft, draft-ietf-manet-odmrp-01 (June 1999).

[LZL06] Weiming Lin, Chongqing Zhang, Minglu Li, et al. 2006. An Interference Graph Based MAC Protocol for Ad Hoc Wireless Networks. Proceedings of The Sixth IEEE International Conference on Computer and Information Technology, CIT'06.

[MaC06] Joseph P. Macker and Ian D. Chakeres. 2006. Mobile Ad Hoc Networking and the IETF. ACM SIGMOBILE Mobile Computing and Communications Review. Volume 10, Issue 1.

[MBP06] Juha-pekka Makela, Timo Braysy and Kaveh Pahlavan. 2006. Analysis of Mobility in Adaptive Data Rate Wireless Networks. MILCOM 2006 pp.1-6.

[Mil09] Jonathan B. Miller. 2009. CERMUSA's Mobile Access Point (MAP) Robotics. [Online]. Available: http://www.jmillville.com/mobile_robotics.html

[MLR01] G.R. Mateus, A.A.F. Loureiro, R.C. Rodrigues. 2001. Optimal Network Design for Wireless Local Area Network. Annals of Operations Research, vol. 106, Sept. 2001, pp. 331 – 345.

[MoR10] Mobile Robots. 2010. Pioneer P3-AT. [Online]. Available: http://www.mobilerobots.com/ResearchRobots/ResearchRobots/P3AT.aspx.

[Moy94] J. Moy. 1994. Multicast Extensions to OSPF. Network Working Group Request for Comments: 1584.

[MuG96] Shree Murthy and J.J.Garcia-Luna-Aceves. 1996. An Efficient Routing Protocol for Wireless Networks. Mobile Networks and Applications 1 (1996) 183-197.

[NaG03] Valeri Naoumov and Thomas Gross. 2003. Simulation of Large Ad Hoc Networks. MSWiM’03, September 19, 2003.

[Ngu08] Uyen Trang Nguyen. 2008. On Multicast Routing in Wireless Mesh Networks. Science Direct, Computer Communications 31 (2008) 1385–1399.

[NS210] The Network Simulator - ns-2. 2010. [Online]. Available: http://nsnam.isi.edu/nsnam.

[NTC99] Sze Y. Ni, Yu C. Tseng, Yuh S. Chen, et al. 1999. The Broadcast Storm Problem in a Mobile Ad hoc Network. Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, pp. 151-162.

[NWa10] NetGear. 2010. WAG511 ProSafe Dual Band Wireless PC Card, Technical Data Sheet. [Online]. Available: http://kbserver.netgear.com/datasheets/WAG511_ds_r4Sep04.pdf.

[OPM10] OPNET Technologies, Inc. 2010. OPNET Modeler. [Online]. Available:http://www.opnet.com/.

[PeB94] Charles E. Perkins, Pravin Bhagwat. 1994. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. In SIGCOMM '94: Proceedings of the conference on Communications architectures, protocols and applications, Vol. 24, No. 4, pp. 234-244.

[PeR99] Charles Perkins and Elizabeth Royer. 1999. Ad-hoc on Demand Distance Vector Routing. Second IEEE Workshop on Mobile Computing Systems and Applications. Proceedings, WMCSA '99.

[PKT02] C. Prommak, J. Kabara, D. Tipper, et al. 2002. Next Generation Wireless LAN System Design. Proc. of the Military Communications Conference (MILCOM’02), 2002, pp. 473 – 477.

[PKT04] C. Prommak, J. Kabara, D. Tipper. 2004. Demand-based Network Planning for Large Scale WLANs. Proc. of the 1st

International Conference on Broadband Networks, BROADNETS'04.

[PNY03] Luiz Felipe Perrone, David M. Nicol, and Yougu Yuan. 2003. Modeling and Simulation Best Practices for Wireless Ad Hoc Networks. Proceedings of the 2003 Winter Simulation Conference.

[PPL10] Perl. 2010. The Perl Programming Language. [Online]. Available: http://www.perl.org/.

[QiC05] Daji Qiao and Sunghyun Choi. 2005. Fast-Responsive Link Adaptation for IEEE 802.11 WLANs. IEEE International Conference on Communications, ICC 2005. Volume 5.

[RGJ06] Pedro M. Ruiz, Francisco J. Galera, Christophe Jelger, et al. 2006. Efficient Multicast Routing in Wireless Mesh Networks Connected to Internet. InterSense '06. Proceedings of the First International Conference on Integrated Internet Ad hoc and Sensor Networks.

[RKD06] Sabyasachi Roy, Dimitrios Koutsonikolas, Saumitra Das, et al. 2006. High Throughput Multicast Routing Metrics in Wireless Mesh Networks. ICDCS. Proceedings of the 26th IEEE International Conference on Distributed Computing Systems, Page: 48.

[RMC08] Roberto Riggio, Daniele Miorandi, Imrich Chlamtac et al. 2008. Hardware and Software Solutions for Wireless Mesh Network Testbeds. IEEE Communications Magazine , June 2008.

[RMM06] Nilson Reyes, Jens Mahnke and Ilka Miloucheva. 2006. Multicast Retransmission Strategies for content delivery in heterogeneous mobile IP environment. International Conference on Advances in the Internet, Processing Systems and Interdisciplinary Research (VIPSI), Belgrade, Serbija August 28-31, 2006.

[RoE09] Beiyu Rong and Anthony Ephremides. 2009. Cooperation Above the Physical Layer: the Case of a Simple Network. IEEE International Symposium on Information Theory, ISIT Seoul, Korea, July 2009.

[RoP99] Elizabeth M. Royer and Charles E. Perkins. 1999. Multicast Operation of the Ad-hoc On-demand Distance Vector Routing Protocol. Mobicom ‘99 Seattle Washington USA.

[RuG05] Pedro M. Ruiz and Antonio F. Gomez-Skarmeta. 2005. Approximating Optimal Multicast Trees in Wireless Multihop Networks. Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC 2005).

[RUS08] Joshua Robinson, Mustafa Uysal, Ram Swaminathan, et al. 2008. Adding Capacity Points to a Wireless Mesh Network Using Local Search. IEEE INFOCOM 2008. The 27th Conference on Computer Communications, p1247-1255.

[SKS02] B. Sadeghi, V. Kanodia, A. Sabharwal, et al. 2002. Opportunistic Media Access for Multirate Ad Hoc Networks. MOBICOM’02, September, 2002, Atlanta, Georgia.

[SMB06] Anand Prabhu Subramanian, Milind M. Buddhikot, and Scott Miller. 2006. Interference Aware Routing in Multi-Radio Wireless

Mesh Networks. Wireless Mesh Networks, 2006. 2nd IEEE Workshop on In Wireless Mesh Networks, WiMesh 2006. 2nd IEEE Workshop on (2006), pp. 55-63.

[SSB99] P. Sinha, R. Sivakumar, and V. Bharghavan. 1999. MCEDAR: Multicast Core-Extraction Distributed Ad hoc Routing. In Proc. of IEEE WCNC, September 1999.

[Sto08] Ivan Stojmenovic. 2008. Simulations in Wireless Sensor and Ad Hoc Networks: Matching and Advancing Models, Metrics, and Solutions. IEEE Communications Magazine, December 2008.

[TBL99] Mineo Takai, Rajive Bagrodia, Addison Lee, et al. 1999. Impact of Channel Models on Simulation of Large Scale Wireless Networks. Proceedings of the 2nd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM '99, pp. 7-14.

[TMB01] Mineo Takai, Jay Martin, Rajive Bagrodia. 2001. Effects of Wireless Physical Layer Modeling in Mobile Ad hoc Networks. Proceedings of the 2nd ACM International Symposium on Mobile Ad hoc Networking & Computing, MobiHoc '01, pp. 87-94.

[TMP10] The MadWifi Project. 2010. [Online]. Available: http://madwifi-project.org/

[Uni05] Uninett, The Norwegian Research Network. 2005. [Online]. Available: http://forskningsnett.uninett.no/wlan/throughput.html.

[UPC10] UCLA Parallel Computing Laboratory. 2010. The Parallel Simulation Environment for Complex Systems, Parsec. [Online]. Available: http://pcl.cs.ucla.edu/.

[Var02] Upkar Varshney. 2002. Multicast over Wireless Networks. Communications of the ACM. December 2002/Vol. 45, No. 12.

[VaV97] Upkar Varshney and Ron Vetter. 2002. Mobile Commerce: Framework, Applications and Networking Support. Mobile Networks and Applications 7, 185–198.

[VCO07] J. Villalon, P. Cuenca, L. Orozco-Barbosa, et al. 2007. ARSM: A Cross-Layer Auto Rate Selection Multicast Mechanism for Multi-Rate Wireless LANs. IET Communications, Volume: 1 , Issue: 5.

[VHH07] T. Vanhatupa, M. Hannikainen, and T.D. Hamalainen. 2007. Genetic Algorithm to Optimize Node Placement and Configuration for WLAN Planning. 4th International Symposium on Wireless Communication Systems, 2007. ISWCS 2007.

[VRH05] Simulation of Large-Scale Sensor Networks Using GTSNetS. 2005. El Moustapha Ould-Ahmed-Vall, George F. Riley, Bonnie S. Heck, et al. Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS’05).

[VST06] J. Villalon, Y. Seok, T. Turletti, et al. 2006. ARSM: Auto Rate Selection Multicast Mechanism for Multi-rate Wireless LANs. Lecture Notes in Computer Science, pages 239-250.

[Wal07] Renaud Waldura. 2007. Dijkstra's Shortest Path Algorithm in Java. [Online]. Available: http://renaud.waldura.com/.

[WFA09] WiFi Alliance. (2009, October 14). Wi-Fi Alliance announces groundbreaking specification to support direct Wi-Fi connections between devices. [Online]. Available: http://www.wi-fi.org/.

[WMA08] WiMedia Alliance. (2008). How It Works? UWB, WPAN and WiMedia Radio Space. [Online]. Available: http://www.wimedia.org/.

[WMB06] Bernhard H. Walke, Stefan Mangold, and Lars Berlemann. 2007. IEEE 802 Wireless Systems. Protocols Multi-hop Mesh Relaying Performance and Spectrum Coexistence. John Wiley & Sons Ltd. England.

[WNE00] Jeffrey E. Wieselthier, Gam D. Nguyen, and Anthony Ephremides. 2000. On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks. In INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, Vol. 2 (2000), pp. 585-594 vol.2.

[WPD88] D. Waitzman, C. Partridge, and S. Deering. 1988. Distance Vector Multicast Routing Protocol. Network Working Group Request For Comments: 1075.

[WPD88] D. Waitzman, C. Partridge, and S. Deering. 1988. Distance Vector Multicast Routing Protocol. Network working Group Request for Comments: RFC1075.

[WSL04] P. Wertz, M. Sauter, F.A. Landstorfer, et al. 2004. Automatic Optimization Algorithms for the Planning of Wireless Local Area Networks. Proc. of the IEEE Vehicular Technology Conference (VTC'04-Fall), 2004, pp. 3010 - 3014.

[XGB03] K. Xu, M. Gerla and S. Bae, 2003. Effectiveness of RTS/CTS Handshake in IEEE 802.11 Based Ad Hoc Networks. Ad Hoc Networks Journal 1, July 2003, 107-123.

[XKW06] Yong Xi, Byung-Seo Kim, Ji-bo Wei, et al. 2006. Adaptive Multirate Auto Rate Fallback Protocol for IEEE 802.11 WLANS. IEEE Military Communications Conference, 2006. MILCOM 2006.

[XTM02] Jason Xie, Rajesh R. Talpade and Anthony McAuley. 2002. AMRoute: Ad Hoc Multicast Routing Protocol. Mobile Networks and Applications 7, 429–439.

[XyP97] George Xylomenos and George C. Polyzos. 1997. IP Multicast for Mobile Hosts. IEEE Communications Magazine, January 1997.

[YLL09] Zhenyu Yang, Ming Li and Wenjing Lou. 2009. R-Code: Network Coding based Reliable Broadcast in Wireless Mesh Networks with Unreliable Links. IEEE Global Telecommunications Conference (GLOBECOM 2009), Honolulu, Hawaii, USA, November 30 - December 4, 2009.

[YWK05a] Yaling Yang, Jun Wang, and Robin Kravets. 2005. Interference-aware Load Balancing for Multihop Wireless Networks. University of Illinois at Urbana-Champaign, Tech. Rep., 2005. [Online]. Available: http://www.cs.uiuc.edu/.

[YWK05b] Yaling Yang, Jun Wang, and Robin Kravets. 2005. Designing Routing Metrics for Mesh Networks. Proceedings of the IEEE Workshop on Wireless Mesh Networks (WiMesh). IEEE Press.

[ZWE05] Rui Zhao, Bernhard Walke, Michael Einhaus et al. 2005. A Primary Adaptation of the W-CHAMB Protocol for Gigabit WPANs. IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications, 2005.

8 Related Work

[ABB04] Daniel Aguayo John Bicket Sanjit Biswas et al. 2004. Link-level Measurements from an 802.11b Mesh Network. *ACM SIGCOMM'04, Aug. 30–Sept. 3, 2004, Portland, Oregon, USA*.

[ATL06] Robert Akl and Dinesh Tummala and Xinrong Li. 2006. Indoor Propagation Modelling at 2.4 GHz for IEEE 802.11 Networks. The IASTED International Conference on Wireless Networks and Emerging Technologies (WNET) 2006.

[AWM08] Andres Arjona, Cedric Westphal, Cedric Westphal et al. 2008. Can the current generation of wireless mesh networks compete with cellular voice? *Computer Communications, Volume 31, Issue 8 (May 2008)*.

[AWW05] Ian F. Akyildiz, Xudong Wang and Weilin Wang. 2005. Wireless Mesh Networks - A Survey. *Computer Networks 47 (2005) 445–487*.

[BaK09] Sung-Jun Bae and Young-Bae Ko. 2009. An Efficient Proactive Tree Building Scheme for IEEE 802.11s based Wireless Mesh Networks. *IEEE VTS Asia Pacific Wireless Communications Symposium (IEEE VTS APWSC 2009)*.

[BeT06] M. Benveniste and Z. Tao. 2006. Performance Evaluation of a Medium Access Control. Protocol for IEEE 802.11s Mesh Networks. Proceedings of IEEE Sarnoff Symposium, 2006.

[BNL06] R. Beuran, L. T. Nguyen, K. T. Latt et al. 2006. Wireless LAN Emulation. *Research Report, IS-RR-2006-015, Japan Advanced*

Institute of Science and Technology (JAIST), Ishikawa, Japan, October 2006.

[BQJ08] Saad Biaz, Bing Qi and Yiming Ji. 2008. Improving Expected Transmission Time Metric in Multi-rate Multi-hop Networks. *Proceedings of IEEE CCNC 2008*.

[Bri06] Patrick Bristow. 2006. Unicast vs Multicast over Wireless a Cross Disciplinary Mindshare. *Proceedings of ITiCSE'06, June 26–28, 2006, Bologna, Italy.*

[CBM05] I. Cubic,. D. Begusic and T. Mandi. 2005. Client based wireless LAN indoor positioning. *Proceedings of the 8th International Conference on Telecommunications, 2005. ConTEL 2005*.

[Cha95] Peter E. Chadwick. 1995. Sensitivity and Range in WLAN Receivers. *IEE Colloquium on Radio LANs and MANs, 1995*.

[DDC97] Christophe Diot, Walid Dabbous, and Jon Crowcroft. 1997. Multipoint Communication: A Survey of Protocols, Functions, and Mechanisms. IEEE Journal on Selected Areas in Communications, Volume. 15, No. 3, April 1997.

[DME07] V.Dasarathan, M.Muthukumar, K.N.Elankumaran et al. 2007. Outdoor Channel Measurement, Pathloss Modelling and System Simulation of2.4 GHz WLAN IEEE 802.11g. *Proceedings of Asia-Pacific Microwave Conference 2007*.

[ECM08] Esposito, P. M., Campista, M. E. M., Moraes et al. 2008. Implementing the Expected Transmission Time Metric for OLSR Wireless Mesh Networks. *IFIP Wireless Days Conference, Dubai, United Arab Emirates, November 2008.*

[EES05] Farshad Eshghi, Ahmed K. Elhakeem, and Yousef R. Shayan. 2005. Performance Evaluation of Ad-hoc Multi Hop. *IEEE Communications Magazine, March 2005*.

[FAC07] Ferre, P. Agrafiotis, D. Chiew, T.K. 2007. Multimedia Transmission over IEEE 802.11g WLANs: Practical Issues and Considerations. *International Conference on Consumer Electronics, 2007. ICCE 2007. Digest of Technical Papers*.

[GMC08] P. Pablo Garrido, Manuel P. Malumbres and Carlos T. Calafate. 2008. ns-2 vs. OPNET - a comparative study of the IEEE 802.11e Technology on MANET Environments. *Proceedings of SIMUTools , March 03 – 07, 2008, Marseille, France*.

[GNA04] Hrishikesh Gossain, Nagesh Nandiraju, Kumar Anand et al. 2004. Supporting MAC Layer Multicast in IEEE 802.11 based MANETs: Issues and Solutions. *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN'04)*.

[HMZ07] G.R Hiertz, S. Max, Rui Zhao et al. 2007. Principles of IEEE 802.11s. *Proceedings of 16th International Conference on Computer Communications and Networks, ICCCN 2007*.

[HoL99] M Hope and N Linge. 1999. Determining the Propagation Range of IEEE 802.11 Radio LANs for Outdoor Applications. *Proceedings of the Conference on Local Computer Networks, 1999, LCN' 99*.

[Jan10] Philipp K. Janert. 2010. *Gnuplot in Action. Understanding Data with Graphs*. Manning Publications.

[JMB00] D. B. Johnson, D. A. Maltz, and J. Broch. 2000. DSR: the dynamic source routing protocol for multihop wireless ad hoc networks. *Addison-Wesley Longman Publishing Co, Inc., December 2000, pp. 139-172*.

[JoH94] David B. Johnson. 1994. Routing in Ad Hoc Networks of Mobile Hosts. *First Workshop on Mobile Computing Systems and Applications, 1994. WMCSA 1994*.

[JZT06] Zhengrong Ji, Junlan Zhou, Mineo Takai et al. 2006. Improving scalability of wireless network simulation with bounded inaccuracies. *ACM Transactions on Modeling and Computer Simulation, Vol. 16, No. 4, October 2006, Pages 329–356.*

[Kap06] Martin Kappes. 2006. An experimental performance analysis of MAC multicast in 802.11b networks for VoIP traffic. *Computer Communications, Volume 29, Issue 8 (May 2006)*.

[KDR07] Dimitrios Koutsonikolas, Saumitra M. Das, Sabyasachi Roy et al. 2007. Towards High-Throughput and Fair Multicast in Wireless Mesh Networks. *In IEEE INFOCOM 2007, Anchorage, Alaska, May 9th, 2007*.

[KNG05] Gautam Kulkarni, Alok Nandan, Mario Gerla et al. 2005. A Radio Aware Routing Protocol for Wireless Mesh Networks. [Online]. Available:http://211.75.223.172/qualnet0092873/papers/ARadio %20AwareRoutingProtocolforWirelessMeshNetworks.pdf.

[KoD06] Karol Kowalik and Mark Davis. 2006. Why Are There So Many Routing Protocols for Wireless Mesh Networks? *Irish Signal and Systems Conference, Dublin, June 28–30, 2006*.

[KuC01] Thomas Kunz and Ed Cheng. 2001. Multicasting in Ad-Hoc Networks: Comparing MAODV and ODMRP. *Proceedings of the Workshop on Ad hoc Communications, 2001*.

[LeR05] Young J. Lee and George F. Riley. 2005. Efficient Simulation of Wireless Networks using Lazy MAC State Update. *Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05)*.

[LeW04] Ann Lee and Paul A.S. Ward. 2004. A Study of Routing Algorithms in Wireless Mesh Networks. *Australian Telecommunication Networks and Applications Conference. December, 2004*.

[LiN01] J¨org Liebeherr and Michael Nahas. 2001. Application-layer multicast with Delaunay triangulations. Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE, Vol. 3 (2001), pp. 1651-1655 vol.3.

[LSW09] Guikai Liu, Chunli Shan, Gang Wei et al. 2009. Subarea Tree Routing (STR) in Multi-hop Wireless Ad hoc Networks. *SciRes Wireless Sensor Network, 2009, 1, 1-60*.

[LWM06] Stephanie Liese, Daniel Wu and Prasant Mohapatra. 2006. Experimental characterization of an 802.11b wireless mesh network. Proceedings of the 2006 International Conference On Communications And Mobile Computing.

[MoA06] Amr Mohamed Hussein Alnuweiri. 2006. Cross-Layer Optimization Framework for Rate Allocation in Wireless

Multicast. *IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS), 2006.*

[MWT09] Allam Maalla, Chen Wei and Haitham J. Taha. 2009. Optimal Power Multicast Problem in Wireless Mesh Networks by using a Hybrid Particle Swarm Optimization. *American Journal of Applied Sciences 6 (9): 1758-1762, 2009*.

[Nic02] David Nicol. 2002. Comparison of network simulators revisited. [Online].Available:http://www.ssfnet.org/Exchange/gallery/dumbbell/dumbbell-performance-May02.pdf.

[Per04] Jimmy Perrson. 2004. Methods of SNR Estimation from Asynchronously Sampled Data in an 802.11b system. Master Thesis, Lulea University of Technology. [Online]. Available: http://epubl.luth.se/1402-1617/2004/002/LTU-EX-04002-SE.pdf.

[REP07] Bernard F. Rolfe, Samitha W. Ekanayake, Pubudu N. Pathirana et al. 2007. Localization with Orientation Using RSSI Measurements: RF Map based approach. *Proceedings of ISSNIP 2007*.

[RoK07] Christof Röhrig and Frank Künemund. 2007. Estimation of Position and Orientation of Mobile Systems in a Wireless LAN. *Proceedings of the 46th IEEE Conference on Decision and Control*.

[RPD05] Joshua Robinson, Konstantina Papagiannaki, Christophe Diot et al. 2005. Experimenting with a Multi-Radio Mesh Networking Testbed. *In Proceedings of the First Workshop on Wireless Network Measurements (WiNMee 2005) (April 2005)*.

[TaG00] K. Tang and M. Gerla. 2000. MAC layer broadcast support in 802.11 wireless networks. *Proceedings of MILCOM 2000. 21st Century Military Communications Conference*.

[TsC05] Tzu-Jane Tsai and Ju-Wei Chen. 2005. IEEE 802.11 MAC Protocol over Wireless Mesh Networks_Problems and Perspectives. *Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA'05)*.

[VOT06] Kumar Viswanath, Katia Obraczka and Gene Tsudik. 2006. Exploring Mesh and Tree-Based Multicast Routing Protocols for MANETs. *IEEE Transactions on Mobile Computing, Vol. 5, No. 1, January 2006.*

[WLW09] Elias Weingärtner, Hendrik vom Lehn, and Klaus Wehrle. 2009. A performance comparison of recent network simulators. *ICC 2009: IEEE International Conference on Communications*.

[WYZ07] Kun Wang, Fan Yang, Member, Qian Zhang et al. 2007. Modelling Path Capacity in Multihop IEEE 802.11 Networks for QoS Services. IEEE Transactions on Wireless Communications, Vol. 6, No. 2, February 2007.

[Xiu04] Xiuchao Wu. 2004. Simulate 802.11b Channel within NS2. Technical Report, 2004. [Online]. Available: http://www.comp.nus.edu.sg/~wuxiucha/research/reactive/publication/Simulate80211ChannelWithNS2.pdf

[Zha07] Rui Zhao. 2007. Mesh Distributed Coordination Function for Efficient Wireless Mesh Networks Supporting QoS. Master

Thesis. [Online]. Available: www.comnets.rwth-aachen.de/publications/dissertations.

[ZLH07] Yan Zhang (Editor), Jijun Luo (Editor) and Honglin Hu (Editor). 2007. *Wireless Mesh Networking: Architectures, Protocols and Standards (Wireless Networks and Mobile Communications)*. Auerbach Publications.

9 Publications

Marek Bykowski, Karol Kowalik, Brian Keegan and Mark Davis. 2008. Throughput Enhancement through Combined Fragmentation and Rate Method in IEEE 802.11b WLANs. *1st Workshop on Wireless Broadband Access for Communities and Rural Developing Regions - WIRELESS4D'08, Karlstad, Sweden 2008*.

Brian Keegan, Karol Kowalik and Mark Davis. 2008. Optimisation of Multicast Routing in Wireless Mesh Networks. *Information Technology and Telecommunications Conference 2008 (ITT 2008), Galway Mayo Institute of Technology, October 2008*.

Karol Kowalik, Marek Bykowski, Brian Keegan and Mark Davis. 2008. An Evaluation of a Conservative Transmit Power Control Mechanism on an Indoor 802.11 Wireless Mesh Testbed. *International Conference on Wireless Information Networks and Systems (WINSYS'08), Porto, Portugal, July 26 - 29, 2008.*

Karol Kowalik, Marek Bykowski, Brian Keegan and Mark Davis. 2008. Practical Issues of Power Control in IEEE 802.11 Wireless Devices. *IEEE International Conference on Telecommunications (ICT 2008), St. Petersburg, June 2008*.

Brian Keegan, Karol Kowalik and Mark Davis. 2007. Experimental Measurement of Overhead Associated with Active Probing of Wireless Mesh Networks. *IEEE International Conference on Signal Processing and Communications (ICSPC 2007), Dubai, November 2007*.

Karol Kowalik, Brian Keegan and Mark Davis. 2007. Making OLSR Aware of Resources. *International Conference on Wireless Communications, Networking and Mobile Computing, WiCom 2007, Pages 1488 - 1493, 21-25 September 2007*.

Karol Kowalik, Brian Keegan and Mark Davis. 2007. RARE - Resource Aware Routing for mEsh. *Proceedings of IEEE International Conference on Communications (ICC 2007), 24-28 June 2007, Glasgow UK*.

Brian Keegan, Karol Kowalik and Mark Davis. 2006. Experimental Analysis of an 802.11b Wireless Mesh Network. *Information Technology and Telecommunications Conference 2006 (ITT 2006), Carlow, Ireland, October 2006*.

Brian Keegan and Mark Davis. 2006. An Experimental Analysis of the Call Capacity of IEEE 802.11b Wireless Local Area Networks for VoIP Telephony. *IEE Irish Signals and Systems Conference 2006 (ISSC 2006), Dublin, June 2006*.

Brian Keegan, 2005. An Experimental Analysis of the Call Capacity of IEEE 802.11b Wireless Local Area Networks for VoIP Telephony. *Master of Philosophy Thesis, Dublin Institute of Technology, August 2005*.

Appendix A

PDF plots for grid positions 1 – 25. Folder: “\Appendix A\Grid Position 1 - 25 Plots\”

Basic Model results data and plot files. Folder: “\Appendix A\Basic Model Results\”

Appendix B

Obsolete (Section moved to main text).

Appendix C

Files containing PDF data (avgTPPDF-N-m.txt) for relay nodes m_{min} to m_{max} are located in, Folder: "\Appendix C\Appendix C1 – C4\";

C1: ETT 1 Mbps midpoint optimised.

C2: *MinHop* 1 Mbps midpoint optimised.

C3: ETT 11 Mbps midpoint optimised.

C4: *MinHop* 11 Mbps midpoint optimised.

Perl script "processFreqDataForPDF.pl" can be used to generate PDF data for all node densities in files "TPFreq-<NODE DENSITY>.txt"

% Gain is calculated as ((TP m_{max} / TP m_{min}) x 100) – 100

ETT 1 Mbps			
	m	TP Mbps	Peak Probability
m_{min}	0	0.16	0.106
m_{max}	14	0.27	0.98
% Gain		68.75	
Hop 1 Mbps			
	m	TP Mbps	Peak Probability
m_{min}	0	0.07	0.44
m_{max}	63	0.09	0.5
% Gain		28.57	
ETT 11 Mbps			
	m	TP Mbps	Peak Probability
m_{min}	0	0.15	0.116
m_{max}	90	0.4	0.98
% Gain		166.67	
Hop 11 Mbps			
	m	TP Mbps	Peak Probability
m_{min}	0	0.07	0.446
m_{max}	219	0.25	0.90
% Gain		257.14	

Figure C-1: Summary of peak probability values.

Appendix D

Complete set of plot files for transmit (Tx) power settings 0 dBm to 18 dBm are located in;

Folder: “\Appendix D\Power Plots\”

Folder: “\Appendix D\3D Plots \”

Folder: “\Appendix D\Contention\”

Folder: “\Appendix D\Coverage Gain Plots\”

Folder: “\Appendix D\Throughput and Delay\”

Appendix E

All source code and scripts for processing data are located in “\Appendix E\”.

Source code for Basic Model simulations located in Folder: “\Appendix E1 - Basic Model\”. The Basic Model contains all of the core functionality of the simulator but does not incorporate any of the optimisation techniques. Parameters are hard coded as global variables. Log and plot files are generated as “*.txt” and “*.plt” respectively. See file header comments for further details.

Source code for Midpoint Optimised simulations located in Folder: “\Appendix E2 - Relay Nodes\”. This program operates using the basic model with the midpoint optimisation. Simulations will optimise the network for a specified “optRate” set by the user. Parameters are hard coded as global variables. Log and plot files are generated as “*.txt” and “*.plt” respectively. See file header comments for further details.

Source code for Power Optimised simulations located in Folder: “\Appendix E3 - Power Adaptation\”.Simulations will optimise a network by tuning the power for each node. Parameters are hard coded as global variables. Log and plot files are generated as “*.txt” and “*.plt” respectively. See file header comments for further details.

Appendix F

Frequency distribution plots of random node placement using Mersenne Twister PRBS (MT). The working plane of 650m x 650m was divided into a 5 x 5 grid. Plots for distributions for each grid position are given below.

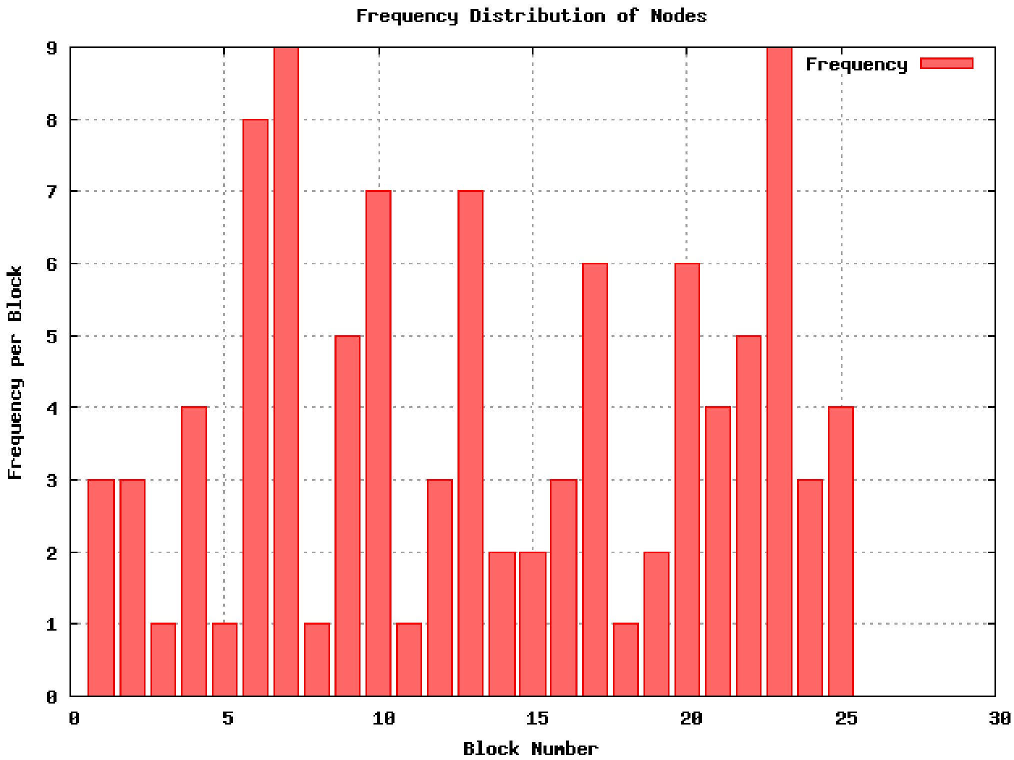


Figure F-1: 10 iterations of MT using 1000 nodes.

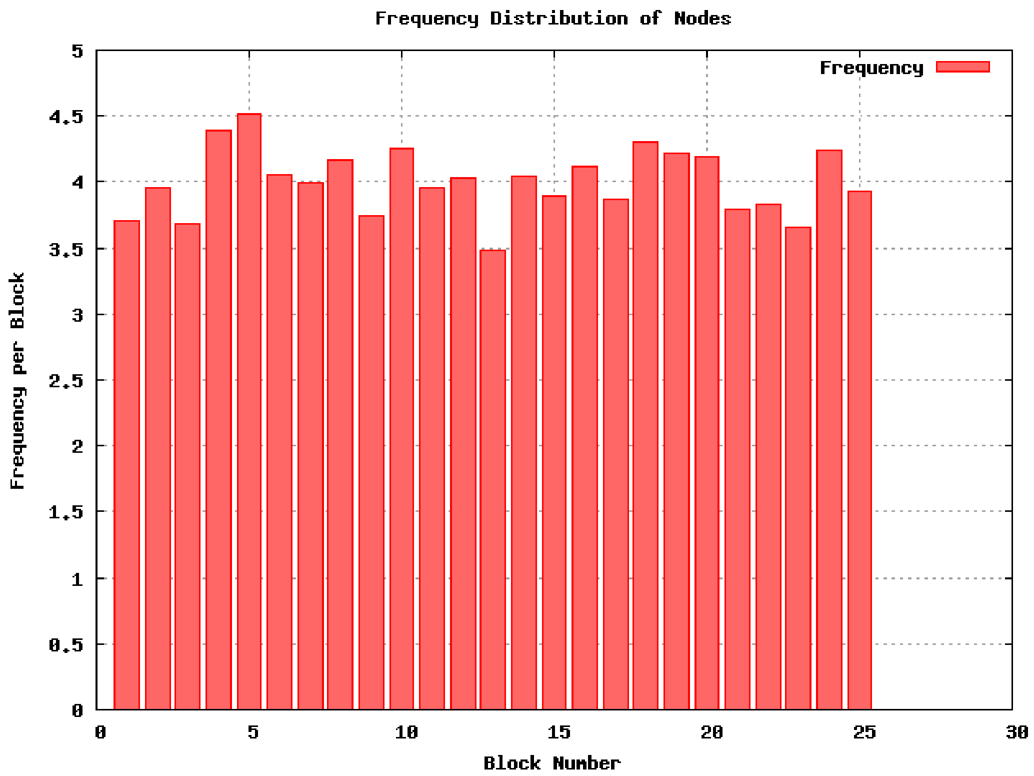


Figure F-2: 100 iterations of MT using 1000 nodes.

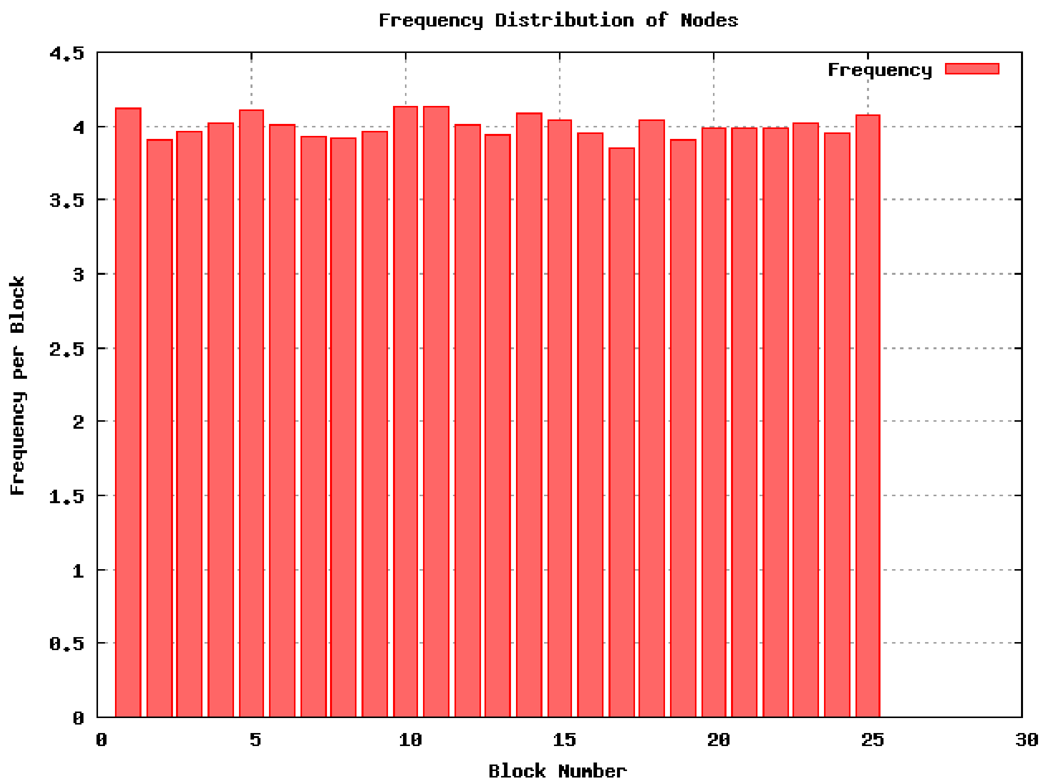


Figure F-3: 1000 iterations of MT using 1000 nodes.

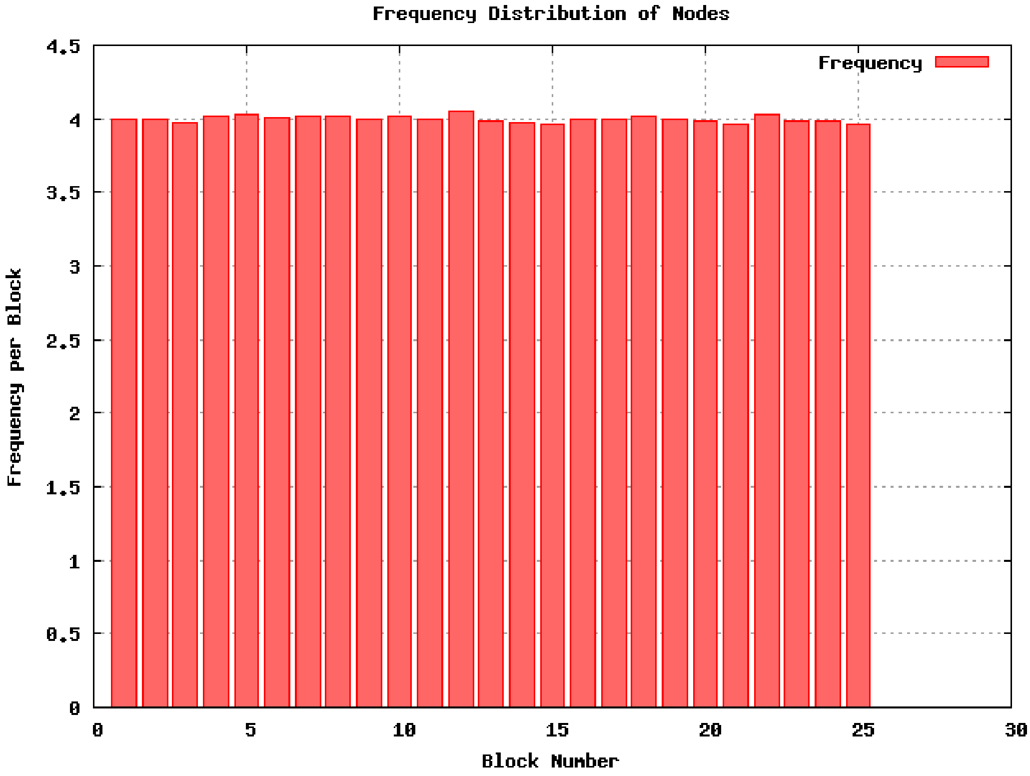


Figure F-4: 10,000 iterations of MT using 1000 nodes.

Appendix G

Diagrams and plots for each chapter are located in;

Folder: “\Appendix G1\Chapter 1\”

Folder: “\Appendix G2\Chapter 2\”

Folder: “\Appendix G3\Chapter 3\”

Folder: “\Appendix G4\Chapter 4\”

Folder: “\Appendix G5\Chapter 5\”

Folder: “\Appendix G6\Chapter 6\”